TCP Performance over Gigabit-Capable Passive Optical Networks

Julio Orozco[1] and David Ros[2]

[1] Orange Labs, 2 avenue Pierre Marzin, 22307 Lannion Cedex, France
julio.orozco@orange-ftgroup.com
[2] Institut TELECOM / TELECOM Bretagne, Rue de la Châtaigneraie, CS 17607, 35576 Cesson Sévigné cedex, France
David.Ros@telecom-bretagne.eu

Abstract. The deployment of optical access networks is considered by many as the sole solution able to cope with the ever-increasing bandwidth needs of data and media applications. Gigabit-capable Passive Optical Networks (GPON) are being adopted by many operators worldwide as their preferred fiber-to-the-home network architecture. In such systems, the Medium Access Control (MAC) layer is a key aspect of their operation and performance.

TCP is the transport protocol of choice of most popular applications. However, TCP performance is known to be sensitive to the behavior of MAC-layer mechanisms. Thus, it is important to assess the impact that the GPON MAC layer may have on TCP. Motivated by this, in this paper we present a preliminary study of TCP performance issues that may arise in GPON networks. Based on a simple system model, the interaction of some GPON MAC features with TCP is explored both analytically and by simulation.

Keywords: TCP, network asymmetry, Passive Optical Networks, GPON.

1 Introduction

A Passive Optical Network (PON) is a type of shared Fiber-To-The-Home (FTTH) network architecture, in which a single fiber is used to connect several users by means of passive splitters. As depicted in Fig. 1, a PON consists of an OLT (Optical Line Terminal) located at the provider's central office and multiple ONTs (Optical Network Terminations) installed inside the customers' premises. The OLT is connected to the ONTs by means of the passive Optical Distribution Network (ODN) composed of fibers and splitters, forming a point-to-multipoint configuration. Current technologies use two wavelengths for data transmission in the same fiber, one for the downstream and the other for upstream. The downstream operates in a broadcast-and-select TDM manner, whereas TDMA is used in the upstream.

Currently, two major PON standards are being deployed, ITU-T's Gigabit-capable PON (GPON) [1], and the IEEE 802.3ah Ethernet PON (EPON). In this paper we focus on the GPON, currently in deployment by major carriers in

C. Wang (Ed.): AccessNets, LNICST 6, pp. 264–279, 2009.

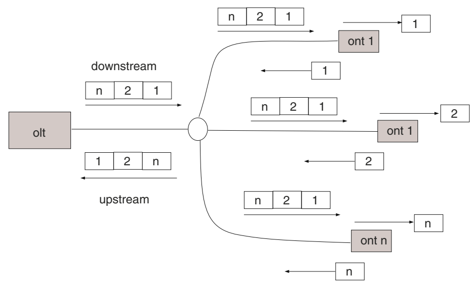


Fig. 1. Passive Optical Network

the US and Europe due to its advantages in terms of bit rates and transport of legacy services [2]. A typical GPON configuration defines an asymmetric capacity of 2.5 Gbit/s in the downstream, 1.25 Gbit/s on the upstream and a split ratio of 32 (i.e., 32 users/ONTs).

The Medium Access Control (MAC) is a key aspect in GPON operation and performance. In what follows, we briefly review the main features of this MAC, specially the Dynamic Bandwidth Allocation (DBA) feature. Recall that ONTs use a single wavelength for upstream transmission. Thus, individual transmissions must be time-scheduled in order to avoid contention. Due to the fact that a simple, fixed allocation is efficient only if all users are active, the standard includes mechanisms for dynamic bandwidth allocation, so that a better resource utilization can be achieved under different, dynamic usage and traffic patterns. Bandwidth allocation is controlled by the OLT, which performs scheduling and communicates the resulting allocation to ONTs periodically.

In GPON, bandwidth can be allocated with a very fine granularity thanks to an abstraction called *Traffic Container* (T-CONT). Indeed, the OLT allocates bandwidth not to ONTs or classes of service, but to individual T-CONTs. The system can ideally handle up to 4096 T-CONTs, each belonging to one of five bandwidth-allocation types and identifed by a number called *alloc id*.

The GPON operates in a cyclic fashion so as to carry TDM traffic. Downstream, a constant 125 μs frame is used. Each frame includes (among other control information) an allocation map which informs on the slots granted to each *alloc id*. Upstream, a reference frame of 125 μs is used. However, this is not an absolute value, since a round of allocations can span through multiple upstream frames. GPON uses the Generic Encapsulation Method (GEM), which allows for the transport, segmentation and reassembly of Ethernet frames and legacy traffic (ATM or TDM).

1.1 TCP and Asymmetry Issues

TCP is the transport protocol used by most Internet applications, including download-based video and rich data applications. The closed-loop nature of TCP becomes an issue with some networking technologies. The TCP source uses the acknowledgment feedback to regulate the transfer rate. When the flow of acknowledgments in the reverse direction is somehow imperfect or variable, the performance in the forward direction can be significantly degraded. This imperfection or variability is thoroughly discussed in RFC 3449 [3], where it is generically called *asymmetry*.

Several access technologies present asymmetric characteristics of different types. RFC 3449 distinguishes two direct types of asymmetry. The first one is plain bandwidth asymmetry, in which the capacities in the forward and reverse directions are different (like in ADSL). The other type of direct asymmetry is due to MAC layers in shared, hub-and-spoke access networks (packet radio, cable, satellite) where upstream transmission experiences higher overhead and latency. Additional asymmetry effects come from bidirectional traffic and the differences in loss rates between the forward and reverse directions. Usually, a GPON system has an inherent *bandwidth asymmetry*. However, due to the characteristics of the link layer there may also exist a *MAC-layer asymmetry*.

1.2 Related Work

To the best of our knowledge, TCP performance over PON networks has been the subject of few papers [4,5,6], and these have all focused on IEEE EPON systems. TCP has also been studied in access networks whose MAC bears some relation with GPON (like the use of TDMA, or the presence of bandwidth asymmetry). See e.g. [7] for an evaluation of TCP over the DOCSIS cable MAC, and [8] for the case of satellite networks. The general asymmetry issues treated in [3] (like the effect of delayed ACKs) are valid to some extent in each context, but the specific TCP behavior and performance issues depend on each type of network.

Prior GPON MAC and DBA performance studies have focused on the local PON system in the upstream direction using open-loop traffic sources [2,9]. However, many key applications are based on TCP, and their quality of service depends on the end-to-end, closed-loop performance. This subject needs to be studied now that large GPON deployments are under way, and both system vendors and operators tend to focus primarily on the optical-layer transmission capacity.

1.3 Goal and Structure of the Paper

This paper presents a preliminary study of the impact of GPON on TCP performance. By means of both a simple mathematical model and simulation, we assess the impact of the GPON MAC layer on TCP. The main focus is on the following aspects: (a) the TDMA operation of the uplink and the fragmentation of Ethernet frames; (b) Dynamic Bandwidth Allocation over the uplink; (c) Bidirectional traffic. The scenarios under study, though very simple, allow us to highlight some interesting effects of the workings of the MAC-level mechanisms.

The rest of the paper is organized as follows. A simple mathematical model of the effect of TDMA and fragmentation on TCP is described in Section 2. Simulation results are shown in Section 3. Lastly, Section 4 concludes the paper.

2 Model

Consider a simple scenario as follows. Without loss of generality, we will focus on a single ONT, connected to a GPON system as depicted in Fig. 1. For the optical link between the ONT and the OLT, let C_u and C_d denote the "raw" bandwidth (i.e., including GEM overhead) in the uplink and downlink, respectively. This link has a (symmetrical) one-way delay of d_o seconds.

Let r_d be the allocated bandwidth in the downstream direction, and r_u the rate allocation in the upstream direction. Both r_d and r_u include the GEM framing overhead. GEM encapsulation adds a header of constant size h. For the uplink, we will consider fixed-length TDMA frames of duration τ_u. Regarding the downlink, for the sake of simplicity we shall assume that there is *no* TDM framing; a GEM packet (i.e., an Ethernet frame with the added GEM header) will be sent to the ONT as soon as it is ready for transmission at the OLT, without waiting for a time slot, and without fragmentation. Note that it is straightforward to extend the model below to explicitly consider the downlink TDM framing.

Recall that GPON supports fragmentation of the transported Ethernet frames. The number of fragments needed to carry a given Ethernet frame will depend on the maximum *net* amount b_u of data that a single TDMA frame can transport, for a given ONT; such amount is given by:

$$b_u = r_u \tau_u - h \, . \tag{1}$$

Of course, allocated bandwidths and frame durations must be such that: $r_u \tau_u - h > 0$. Given our previous assumption on the downlink, there will be no fragmentation of downstream packets.

Let s_d be the (average) size of Ethernet frames sent in the downstream direction. Likewise, let s_u denote the (average) size of upstream Ethernet frames. Then, the minimum number of fragments N_u required for sending a frame of size s_u is given by:

$$N_u = \left\lceil \frac{s_u}{b_u} \right\rceil \geq 1 \, . \tag{2}$$

The number of *full-sized* fragments (i.e., fragments of size b_u) corresponding to an upstream packet of size s_u is:

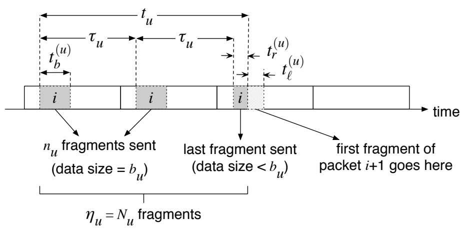
$$n_u = \left\lfloor \frac{s_u}{b_u} \right\rfloor \geq 0 \, . \tag{3}$$

In other words, an Ethernet frame of size s_u could completely fill n_u TDMA bursts (each carrying $r_u \tau_u$ of GEM-level data), and a minimum of $N_u \geq n_u$ bursts would be needed to carry such a frame.

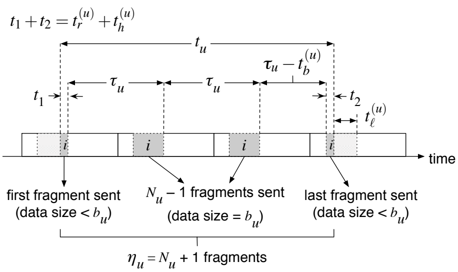
2.1 Packet Transmission Times

In GPON, the TDM/TDMA framing of data packets (i.e., Ethernet frames), together with the eventual fragmentation and "packing" of several GEM packets in a single burst, may have an impact on the time it takes to send a packet over the optical link, as well as on the spacing of packets over time. Packets of the same size may experience different sending times, depending on their transmission start and end times with respect to the framing. Moreover, in the case there is a backlog of packets ready to be sent, the transmission of a packet may not start immediately after the previous one, but it may be delayed until the next TDM/TDMA frame.

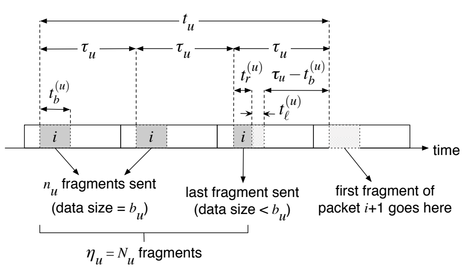
To better visualize the effect of the GPON link layer, consider the three different cases that may arise, illustrated in Fig. 2. Assume a packet i is ready to



(a) Best case; $t_\ell^{(u)} \geq (h+1)/C_u$.



(b) Extra burst needed; $t_\ell^{(u)} \geq (h+1)/C_u$.



(c) Next packet delayed until the next burst; $t_\ell^{(u)} < (h+1)/C_u$.

Fig. 2. Effective transmission time for sending a packet upstream

be sent at the ONT, and there is a backlog of packets $i+1, i+2\ldots$ to be transmitted upstream. Assume further that there is no actual bandwidth contention between ONTs, in the sense that every ONT can send a burst of size $r_u\tau_u$ in *every* TDMA frame.

We will denote by t_u the time interval between the transmission start times of two consecutive, backlogged upstream packets. Note that t_u can be regarded as an *effective* packet transmission time, in the sense that t_u includes any eventual delay due to the TDMA framing[1].

Let us examine first the case of Fig. 2a, in which the number of packet-i fragments sent (denoted as η_u) is *exactly* N_u. The first fragment of i completely fills an upstream burst of duration[2]:

$$t_b^{(u)} = r_u\tau_u/C_u \,. \tag{4}$$

Each fragment (including its GEM encapsulation) is sent at the uplink rate of C_u. Since $N_u \geq 1$, several TDMA frames may be needed to send a packet of size s_u. If $N_u = n_u$, all N_u frames carry a fragment of maximum size b_u. On the other hand, if $N_u = n_u + 1$ the last frame will carry the remaining $s_u - n_u b_u$ units of data, plus the GEM header of length h. This last GEM packet takes $t_r^{(u)}$ time units to be transmitted, where:

$$t_r^{(u)} = \frac{s_u - (N_u - 1)\cdot b_u + h}{C_u} \,. \tag{5}$$

Finally, the time $t_\ell^{(u)}$ that remains at the end of the last burst is long enough to hold a fragment, carrying at least one byte of data, of the next queued packet $(i+1)$. Hence, transmission of $i+1$ may start in the same burst as the last fragment of i. That is, assuming units of bytes and bytes/s: $t_\ell^{(u)} \geq (h+1)/C_u$.

Putting all this together, we can express the time t_u in the best case as:

$$t_u = n_u\tau_u + (N_u - n_u)\cdot t_r^{(u)} \,. \tag{6}$$

Note this formula covers both $N_u = n_u$ and $N_u = n_u + 1$ cases. It also includes the particular case in which one Ethernet frame is sent in a single GEM packet (i.e., a frame of size $< b_u$, so that $n_u = 0$ and $N_u = 1$), and the transmission of the next packet starts in the same burst.

In a worse case, it may happen that an extra upstream burst is required to send the whole packet—i.e., the number of transmitted fragments is $\eta_u = N_u + 1$ because the "excess" amount of data $s_u - n_u b_u$ (which is $< b_u$) ends up being sent in two GEM packets instead of a single one. This is illustrated in Fig. 2b, in which the first burst can only hold a "small" fragment of packet i (i.e., carrying less than b_u of data); hence, an extra GEM packet has to be sent in the end,

[1] In the case where there is no backlog, packets may of course be more spaced over time; t_u is thus a lower bound on the inter-arrival times of packets at the uplink.

[2] Without loss of generality, we do not consider here the physical- and MAC-layer overhead in GTC frames, which is assumed to be accounted for in the value of r_u.

incurring in an extra overhead of h. This adds a time length of $\tau_u - t_b^{(u)} + t_h^{(u)}$ to the actual transmission time, so that, in this case:

$$t_u = N_u \tau_u + t_r^{(u)} + t_h^{(u)} - t_b^{(u)} , \tag{7}$$

with $t_h^{(u)} = h/C_u$, and $t_r^{(u)}$, $t_b^{(u)}$ given by (5) and (4), respectively. Since $t_\ell^{(u)} \geq (h+1)/C_u$, transmission of packet $i+1$ can start in the same burst.

Finally, consider the case shown in Fig. 2c. The number of sent fragments is $\eta_u = N_u$, but the time $t_\ell^{(u)}$ available at the end of the last burst is too small to hold a fragment of the next packet; that is, $t_\ell^{(u)} < (h+1)/C_u$. In this case, an interval of $\tau_u - t_b^{(u)} + t_\ell^{(u)}$ time units is added to the actual transmission time of packet i, so finally we have:

$$t_u = (n_u + 1) \cdot \tau_u + (N_u - n_u) \cdot t_r^{(u)} + t_\ell^{(u)} - t_b^{(u)} . \tag{8}$$

2.2 Special Case: Small Packets

When Ethernet frames are "small enough", we may have: $N_u = 1$, $n_u = 0$ for a wide range of system parameter values. Generally speaking, this happens whenever $s_u < b_u$; in particular, this may often be the case when frames carry TCP pure ACK segments—i.e., segments carrying no data, only ACK information. In such a case, expressions (6)-(8) reduce respectively to:

$$t_u = \begin{cases} t_r^{(u)}, & \text{if } \eta_u = 1 \text{ and } t_\ell^{(u)} \geq \frac{h+1}{C_u} \\ \tau_u + t_r^{(u)} + t_h^{(u)} - t_b^{(u)}, & \text{if } \eta_u = 2 \\ \tau_u + t_r^{(u)} + t_\ell^{(u)} - t_b^{(u)}, & \text{if } \eta_u = 1 \text{ and } t_\ell^{(u)} < \frac{h+1}{C_u} \end{cases}$$

As we will see, given the common numerical values of C_u and other system parameters, it is safe to assume that $t_\ell^{(u)} \approx t_h^{(u)} \ll \tau_u$ in the latter case. Therefore, for the sake of simplicity we may treat the last two cases as a single one, so:

$$t_u = \begin{cases} t_r^{(u)}, & \text{if } \eta_u = 1 \text{ and } t_\ell^{(u)} \geq \frac{h+1}{C_u} \quad (9) \\ \tau_u + t_r^{(u)} + t_h^{(u)} - t_b^{(u)}, & \text{otherwise} \quad (10) \end{cases}$$

From (9) and (10) we see that, for a given packet size s_u, the absolute difference in effective transmission times of two consecutive small packets can be as large as $\tau_u + t_h^{(u)} - t_b^{(u)}$. With typical parameter values, this difference is $\approx \tau_u$, which may be much greater than the best-case time $t_r^{(u)}$. Fig. 3 illustrates this phenomenon, for the following settings: $C_u = 1.24$ Gbit/s, $C_d = 2.48$ Gbit/s, $r_u = 2$ Mbit/s, $r_d = 100$ Mbit/s, $h = 5$ bytes, $s_u = 78$ bytes.

As we can see in the figure, when $N_u = 1$, $n_u = 0$ (i.e., to the right of the dashed vertical line), the difference in effective transmission times between the worst and best cases is large. It is easy to see from (6)-(8) that, as soon as $N_u > 1$, the difference in the values of t_u for consecutive packets is still on the order of τ_u. Nevertheless, this difference is no longer large relative to the best-case time given by (6).

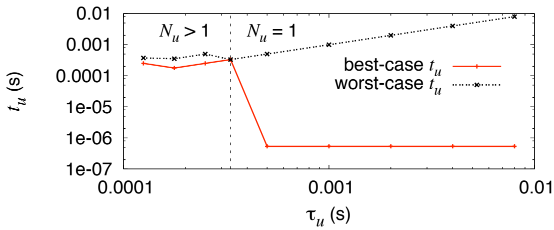


Fig. 3. Effective transmission times: best and worst cases

2.3 ACK Compression

TCP ACK compression effects [3] are characteristic of bandwidth-asymmetric networks. In the case of GPON, bandwidth asymmetry may of course arise depending on the allocated uplink and downlink rates r_u and r_d. Nonetheless, the potentially large "jitter" in the values of t_u may induce strong ACK compression *even in the absence of (normalized) bandwidth asymmetry and of reverse traffic.*

To illustrate this, consider a download-only scenario in which there is a single long-lived TCP flow, with data flowing downstream. For the sake of simplicity, we shall take fixed-size data packets, of length s_{data}. TCP pure ACK packets are of (fixed) size $s_{\text{ack}} \ll s_{\text{data}}$. We will consider that the receiver uses standard delayed ACKs, so that $\delta = 2$ data packets have to be received before an ACK is sent. Let us assume that network parameters are such that there is *no* bandwidth asymmetry, that is, the *normalized bandwidth ratio* κ [3]:

$$\kappa = \frac{1}{\delta} \cdot \frac{s_{\text{ack}} + h}{r_u} \cdot \frac{r_d}{s_{\text{data}} + h} \tag{11}$$

is such that: $\kappa < 1$; besides, $N_u = 1$ and $n_u = 0$, i.e., ACKs are "small" with respect to the burst size, so (9)-(10) hold. Further, the TCP sender's window is supposed large enough to ensure a steady flow of ACKs from the TCP receiver. That is, the sender is in principle able to "fill the pipe" by sending at a rate r_d.

Since the sender is not slowed down by bandwidth asymmetry effects, ACKs may arrive at the ONT at a rate $\lambda_{\text{ack}} = r_d/(\delta \cdot (s_{\text{data}} + h))$ ACKs per unit time. If, for some packet i (a TCP ACK), the effective transmission time t_u corresponds to the worst case (10), a total of B_{ack} ACKs will arrive at the ONT queue during this interval t_u, with:

$$B_{\text{ack}} = \lambda_{\text{ack}} t_u = \lambda_{\text{ack}} \cdot \left(\tau_u \cdot (1 - r_u/C_u) + t_r^{(u)} + t_h^{(u)} \right) . \tag{12}$$

In order for this backlog to be cleared, during the next burst (lasting $t_b^{(u)}$ time units) the ONT has to send at least B_{ack} Ethernet frames of size s_{ack}, that is: $B_{\text{ack}} < t_b^{(u)} C_u/(s_{\text{ack}} + h)$. From (11) and (12), we see this condition is equivalent to: $\kappa < \tau_u / \left(\tau_u + t_r^{(u)} + t_h^{(u)} - t_b^{(u)} \right) \approx 1$. Hence, the ONT queue will oscillate between empty periods and periods where a (potentially large) number of ACKs get queued. As we will see in Section 3.1, the release of a burst of ACKs in a single TDMA frame may result in increased burstiness at the TCP sender.

Note from (12) that, for a fixed r_u, B_{ack} increases with increasing τ_u; on the other hand, for the values of r_u/C_u we will consider, the impact of the allocated uplink rate on B_{ack} should be negligible.

When there is bandwidth asymmetry (i.e., when $\kappa > 1$), a "permanent" backlog may form at the ONT due to the asymmetry, but such backlog may also oscillate due to the fluctuations in t_u. Such oscillations will be salient when $n_u = 0$ (so $N_u = 1$), since in that case several ACKs may be released in a single uplink burst of duration $t_b^{(u)}$ at the line rate C_u. The amount of ACKs ΔB_{ack} that can be sent during such a burst is approximately:

$$\Delta B_{\mathrm{ack}} = \frac{t_b^{(u)} C_u}{s_{\mathrm{ack}} + h} = \frac{r_u \tau_u}{s_{\mathrm{ack}} + h}\,, \tag{13}$$

hence, longer TDMA frames should result in larger queue oscillations.

3 Simulation Results

In this section we will present some simulation results that support the analysis in Section 2. A simulation model of GPON was implemented in OPNET Modeler. The model allows to simulate with a fair level of detail the MAC layer of the optical access network, in particular: GEM encapsulation, fragmentation of Ethernet frames, and Time Division Multiple Access. Our model enables to explore different aspects related to Dynamic Bandwidth Allocation in a generic fashion, since the specific algorithms are vendor-proprietary. We consider a single explicit traffic container per ONT. The rate allocations can be fixed or variable (which allows for the representation of different types of T-CONTs like best-effort or assured), and can span through a configurable number of TDMA frames.

Fig. 4 shows the general simulation scenario and topology used. Though simplistic, such scenario allows to verify the occurrence of some phenomena predicted by the model in Section 2. The one-way delay between the router R2 and the end host H0 is 100 ms; all other propagation delays are negligible with respect to this value. Unless stated otherwise, buffer sizes in both routers, as well as in the OLT, are "infinite", so that no packet loss occurs in them; ONT buffers can contain up to 1000 maximum-sized Ethernet frames.

3.1 Download-Only Case: Fixed Bandwidth Allocation

In this scenario, host H0 acts as a server, and host H1 downloads a large file (of size 100 MB) from H0. The TCP connection is opened by H1 at time $t = 60$ s.

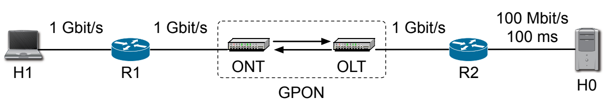
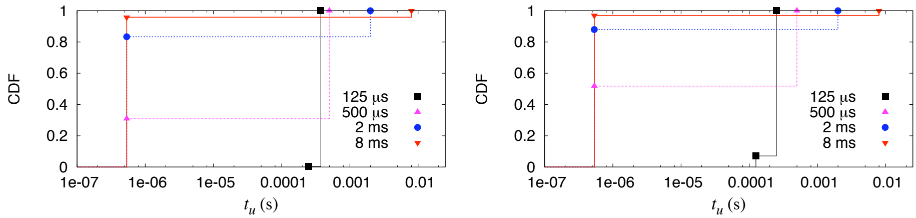


Fig. 4. Simulation scenario and topology



(a) $r_u = 2$ Mbit/s. **(b)** $r_u = 10$ Mbit/s.

Fig. 5. CDF of the effective upstream transmission times t_u

The TCP sender at the server advertises a constant receiver window *rwnd* of 2.5 MB; such a large value was chosen so as to ensure that the sender would in principle be able to fill its downstream pipe of bandwidth r_d (the bandwidth-delay product of the path between H0 to H1 is ≈ 2.5 MB).

The allocated upstream rate r_u remains constant during the simulation. We considered the common, realistic case of $r_u = 10$ Mbit/s, and also three lower rates: 1 Mbit/s, 2 Mbit/s and 5 Mbit/s. Remark that κ, as given by (11), is > 1 for $r_u = 1$ and 2 Mbit/s, but < 1 for $r_u = 5$ and 10 Mbit/s. In order to assess the impact of TDMA frame duration, we considered a wide range of values of τ_u, including a few extreme cases (i.e., very long frames) which are not necessarily realistic, but which allow to "push" the system in order to better highlight some potential issues. Values of τ_u are multiples of the basic frame length of 125 μs.

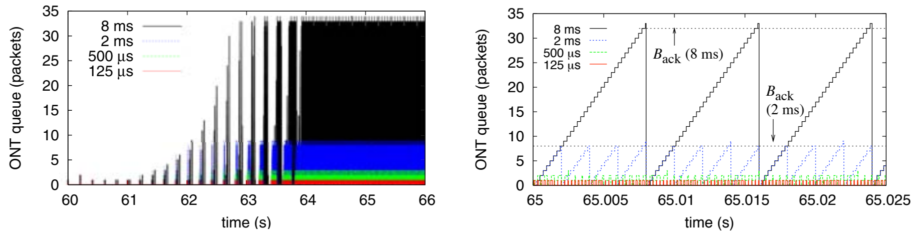
Values for other network parameters were fixed as follows: $C_u = 1.24$ Gbit/s, $C_d = 2.48$ Gbit/s, $r_d = 100$ Mbit/s, $h = 5$ bytes, $s_u = 78$ bytes, $s_d = 1518$ bytes.

Fig. 5 shows the distribution of the effective upstream transmission times t_u, for two different values of r_u and four different frame lengths τ_u (125 μs, 500 μs, 2 ms and 8 ms). As seen from the CDFs, the proportion of ACKs for which $t_u \approx \tau_u$ (i.e., "long" effective transmission times, during which the ONT queue fills up) gets lower with increasing τ_u; however, according to (12) the backlog B_{ack} of ACKs that builds up during such long transmission times should increase with increasing τ_u, leading to larger queue oscillations. This can be verified in Fig. 6a, which corresponds to the bandwidth-symmetric case ($\kappa < 1$). The value of B_{ack} predicted by (12) is shown for the two longest frame durations.

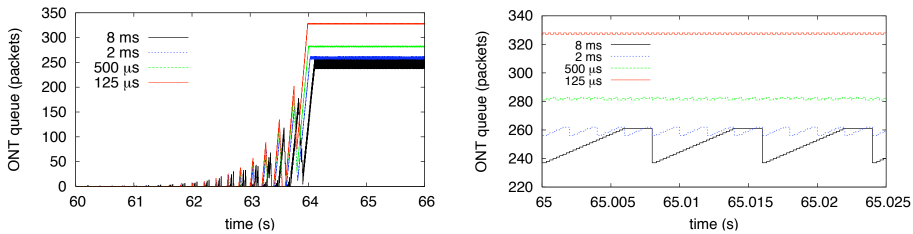
In the bandwidth-asymmetric case ($r_u = 2$ Mbit/s), shown in Fig. 6b, we can observe: (a) the persistent ACK backlog in the ONT queue; (b) the oscillations in this queue due to the large fluctuations in t_u, for long TDMA frames. The amplitude of the queue fluctuations for the 2-ms and 8-ms frames fits well with the value of ΔB_{ack} given by (13), as expected (≈ 6 and 24 packets, respectively).

ACK compression due purely to MAC-layer effects can be seen in Fig. 7, for a bandwidth-symmetric setting. Remark that, with respect to the $\tau_u = 500$ μs case, the burstiness of the sender increases sharply with a frame of 2 ms: bursts of $B_{\text{ack}} \approx 8$ ACKs arrive at H0 every 2 ms, so the sender generates bursts of $\approx \delta B_{\text{ack}} = 16$ data packets every 2 ms.

Download times (i.e., the time it takes to receive the full 100 MB file at H1) are shown in Fig. 8, for a wide range of TDMA frame durations. As expected,

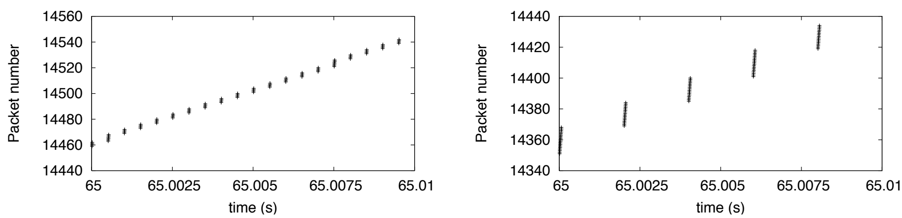


(a) $r_u = 10$ Mbit/s.



(b) $r_u = 2$ Mbit/s.

Fig. 6. Uplink ONT queue



(a) $\tau_u = 500\ \mu$s. **(b)** $\tau_u = 2$ ms.

Fig. 7. Sequence numbers of data packets; $r_u = 10$ Mbits/s

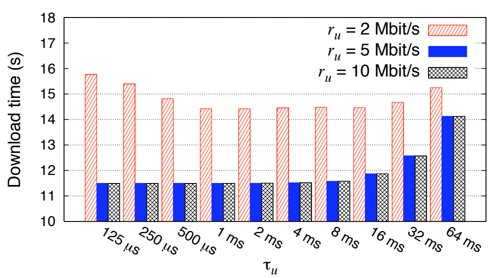


Fig. 8. Total download times as a function of TDMA frame duration

in the two bandwidth-symmetric cases (i.e., when $\kappa < 1$) the uplink rate has no sensible effect on the transfer time, whereas such time increases noticeably when the return path is "slow". Note that, in the $\kappa > 1$ case, download times increase when τ_u gets smaller. This is due to several factors. First, for $\tau_u = 125$

and 250 μs, we have $N_u > 1$, so the GEM overhead for every ACK gets larger in proportion. Second, as seen in Fig. 6b, the backlog at the ONT increases with decreasing τ_u, and so the mean RTT experienced by the flow (not shown for space reasons) grows; hence, the mean throughput gets lower and thus the transfer time increases. In all cases, when τ_u gets very large, its contribution to delay becomes non-negligible, so the download performance degrades.

3.2 Impact of DBA

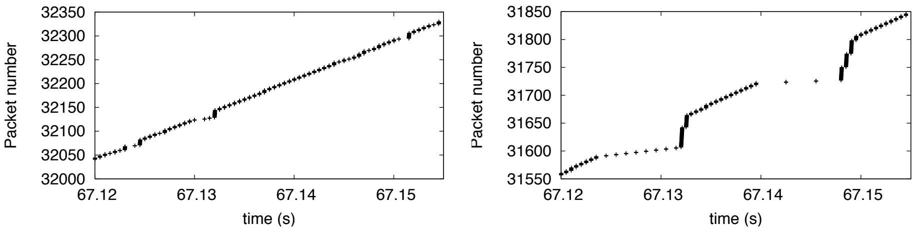
In order to assess the effect of DBA, we performed a similar evaluation as that in Section 3.1 (i.e., a download-only scenario), but using a model of allocated bandwidth as follows[3]. The OLT grants the ONT a new rate allocation in a periodic fashion, every T_{DBA} seconds, with T_{DBA} an integer multiple of τ_u. The allocation values (in bytes) are drawn from an exponential distribution of mean a; outcomes from the distribution are truncated at a maximum value $a_{\max}$. An allocation of β bytes, valid for $N_f = T_{\mathrm{DBA}}/\tau_u$ frames, corresponds to the amount of data that can be sent in *one* TDMA frame lasting τ_u time units; hence, the "short-term" allocated rate (i.e., during an interval of T_{DBA} time units) would be $8\beta/\tau_u$ bit/s, and the mean allocated upstream rate $E(r_u)$ is $8a/\tau_u$ bit/s.

The TDMA frame duration was fixed at $\tau_u = 500$ μs. We used three different values of T_{DBA}: 500 μs, 2 ms and 8 ms, i.e., allocations may span $N_f = 1$, 4 or 16 TDMA frames. Values of a and $a_{\max}$ were chosen as follows: $a_{\max} = 6250$ bytes (so the short-term upstream rate may go up to a maximum of 100 Mbit/s); $a = 125$ and 625 bytes, corresponding respectively to $E(r_u) = 2$ and 10 Mbit/s.

Fig. 10a shows the ONT queue for the $E(r_u) = 10$ Mbits/s average rate allocation, for the three values of N_f. Remark that the fluctuations in allocated rate may induce strong ACK queue fluctuations—hence, a bursty behavior at the TCP sender. This is an expected result: since the allocations may fall to very low values, a fairly large backlog may build up at a given time, then be quickly released when a large allocation is granted—compare this with the fixed $r_u = 10$ Mbit/s case in Fig. 6a, for the $\tau_u = 500$ μs frame. Note that, the higher the value of N_f (i.e., the longer the allocations last), the higher both the "peaks" *and* the mean queue tend to be; thus, the TCP sender may be much more bursty, as seen in Fig. 9.

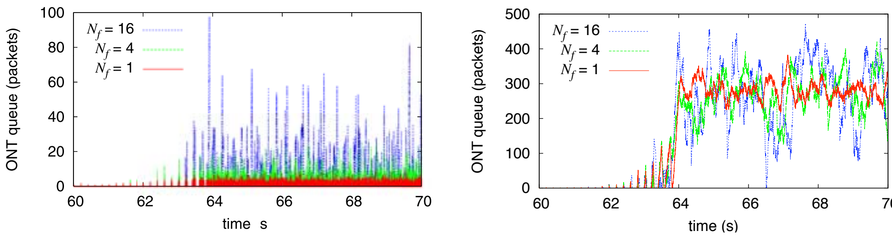
In the $E(r_u) = 2$ Mbit/s case, depicted in Fig. 10b, we see that the ONT queue tends to oscillate around a mean value of ≈ 260 packets, irrespective of the allocation cycle N_f. Again, the queue fluctuations are due to the varying allocations. Longer allocation cycles result in larger oscillations. However, the *mean* queue size, which depends essentially on the bandwidth asymmetry, remains stable.

[3] We do *not* claim that this model would correspond to actual bandwidth allocations among competing flows and ONTs, since we are not trying to describe explicitly a particular polling and scheduling mechanism, nor a specific traffic load. Rather, such a model is used simply to capture, in a straightforward manner, the fact that allocations may fluctuate widely over time, around a pre-defined mean value. In practice, this could be implemented as e.g. a Hybrid-type T-CONT.



(a) $N_f = 1$. **(b)** $N_f = 16$.

Fig. 9. Sequence numbers of data packets; $E(r_u) = 10$ Mbit/s



(a) $E(r_u) = 10$ Mbits/s **(b)** $E(r_u) = 2$ Mbits/s

Fig. 10. Uplink ONT queue, when DBA is used

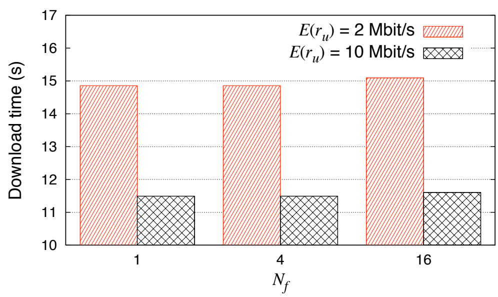


Fig. 11. Total download times as a function of the allocation cycle

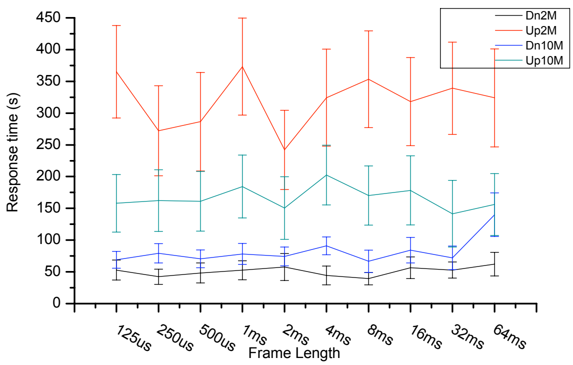
Given that the kind of allocation mechanism modeled ensures a long-term, average rate, mean download times should remain close to those observed in the case of Section 3.1. This can be verified by comparing the values for $\tau_u = 500\ \mu$s in Fig. 8 with the average values in Fig. 11, computed from 30 independent simulation runs (95%-confidence intervals, not shown in the plot, are all < 0.25 s).

3.3 Simultaneous Downloading and Uploading

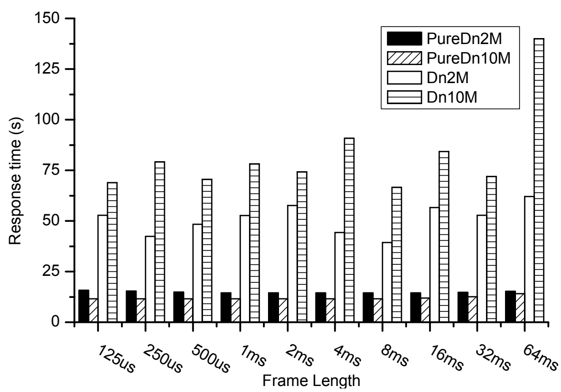
In the previous sections we dealt with unidirectional content transfer in the downstream, so that the sole traffic in the upstream was the ACK flow. In the case of bidirectional, long-lived data transfers, traffic in each direction is a mix of data and ACK packets. On the average, the proportion of data and ACK packets can be approximated as $\frac{\delta}{1+\delta}$ and $\frac{1}{1+\delta}$, respectively. Therefore, the mean packet size in both directions will be: $s_u = s_d = \frac{s_{\text{ack}}}{1+\delta} + \frac{\delta \cdot s_{\text{data}}}{1+\delta}$, in which case the normalized bandwidth ratio becomes: $\kappa' = r_d/r_u > \kappa$; usually, this will mean $\kappa' > 1$, so there will be bandwidth asymmetry.

Here, we explore the case of bidirectional user traffic, adding a file transfer from H1 to H0 in the simulation model. A 25 MB upload starts after the download with a time offset uniformly distributed between 0 and 5 s. The same fixed allocations as in Section 3.1 were used. For each scenario, 30 independent simulation runs were done, and we measured the transfer times in both directions.

The plot in Fig. 12a shows the average transfer times with 95% confidence intervals, for $r_u = 2$ and 10 Mbit/s. In Fig. 12b we compare the download times obtained in the pure download case (shown previously in Fig. 8) to the mean download times in the bidirectional scenario. There are two salient aspects in these results. First, the download performance is heavily affected by the data



(a) Download and upload duration, for $r_u = 2$ and $r_u = 10$ Mbit/s.



(b) Download times: pure download vs. bidirectional data traffic.

Fig. 12. Transfer times

traffic in the upstream direction, as expected. Second, remark the large variability in transfer times, as shown by the confidence intervals.

As seen in Fig. 12b, download times increase by an average factor of 3.4 for $r_u = 2$ Mbit/s and of 6.8 for $r_u = 10$ Mbit/s. This suggests that download performance degrades with *increasing* upstream rate. A possible explanation is that the higher allocated capacity allows the upstream TCP source to increase its transmission rate faster. Hence, the queue is filled by packets from the upstream data flow at a faster rate, leaving less "transmission opportunities per unit time" for sending ACK packets, and in general impairing the upstream ACK flow.

4 Conclusions and Perspectives

This paper has presented a preliminary assessment of the impact that the MAC layer of GPON systems may have on the performance of the TCP protocol. In particular, we have studied the effect of TDMA operation and fragmentation on quality metrics like transfer times. Also, we have looked at performance issues like TCP sender burstiness that may have an influence on packet loss, when more realistic settings (congestion along the end-to-end path, traffic from competing users, etc.) are considered. These preliminary findings require further analysis, and are of interest because of the potential service provisioning issues.

So far, we have focused on a simple scenario with a single ONT, abstracting the effect of competing traffic and contention between ONTs. As future work, we wish to consider multiple, competing ONTs, in order to evaluate aspects like the fairness among ONTs and the efficiency of resource utilization of the GPON as a system. Besides, more realistic network conditions, like more complex traffic patterns and congestion in different points of the end-to-end path, should be considered. Another subject of interest is the study of the case in which $r_u = r_d$, i.e., when any asymmetry effects would be due solely to the MAC layer. Finally, the behavior of high-speed TCP versions (which compete more aggressively for bandwidth) in the context of GPON is also worth investigating. Also, it would be interesting to compare the performance of TCP flows over GPON and EPON, under similar conditions.

References

1. ITU-T: Recommendation G.984.3: Gigabit-Capable Passive Optical Networks (G-PON): Transmission Convergence Layer Specification (2003)
2. Angelopoulos, J., Leligou, H.C., Argyriou, T., Zontos, S., Ringoot, E., Van Caenegem, T.: Efficient transport of packets with QoS in an FSAN-aligned GPON. IEEE Communications Magazine 42, 92–98 (2004)
3. Balakrishnan, H., Padmanabhan, V.N., Fairhurst, G., Sooriyabandara, M.: TCP performance implications of network path asymmetry. Best Current Practice RFC 3449, IETF (2002)
4. 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)

5. 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)
6. 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)
7. Martin, J.: The impact of the DOCSIS 1.1/2.0 MAC protocol on TCP. In: Proceedings of IEEE CCNC, Las Vegas, pp. 302–306 (2005)
8. Papadimitriou, P., Tsaoussidis, V.: On TCP performance over asymmetric satellite links with real-time constraints. Computer Communications 30, 1451–1465 (2007)
9. Payoux, F., Niger, P., Vu-Brugier, G.: Modeling and simulation of Dynamic Bandwidth Allocation function for GPON and Next Generation PON. In: Proceedings of OPNETWORK 2007, Washington D.C (2007)