Computer Networks 57 (2013) 1577–1596

Contents lists available at SciVerse ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

Adaptive Delay-Aware Energy Efficient TDM-PON

S.H. Shah Newaz [a,*], Ángel Cuevas [b], Gyu Myoung Lee [c,*], Noël Crespi [c], Jun Kyun Choi [a]

[a] Department of Electrical Engineering, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea
[b] Department of Telematic Engineering, University Carlos III of Madrid, Spain
[c] Department of Wireless Networks and Multimedia Services, Institut Telecom, Telecom SudParis, France

ARTICLE INFO

Article history:
Received 25 September 2012
Received in revised form 19 January 2013
Accepted 2 February 2013
Available online 14 February 2013

Keywords:
Energy saving
TDM-PONs
Sleep mode
Operator delay requirement
TCP throughput

ABSTRACT

Passive Optical Networks (PONs) are widely adopted as the last-mile technology due to the large bandwidth capacity it provides to end users. In addition, PON is viewed as a green access technology since it reduces energy consumption compared to other access technologies (e.g. Fiber to the Node). However, there is still room for enhancing the energy efficiency of PON further, and we can find many attempts along those lines in academia and industry. A widely used approach to save energy in Time Division Multiplexing (TDM)-PON is to keep the Optical Network Units (ONUs) in sleep mode when they do not have anything to receive or transmit. However, sleep intervals have a direct negative impact on increasing traffic delay. Therefore, energy efficiency in a TDM-PON presents a clear trade-off: the longer an ONU sleeps, the less energy it consumes, but the higher the delay experienced by the downlink traffic, and vice versa. In this paper, we propose an Adaptive Delay-Aware Energy Efficient (ADAEE) TDM-PON solution. The ADAEE aims at saving as much energy as possible while meeting the PON access delay restrictions imposed by the operator. We evaluate our solution in terms of energy consumption and delay performance using real traffic traces. The results demonstrate that the proposed solution can meet delay requirements while being more energy efficient solution compared to the existing solutions.

1. Introduction

There is an urgent need for energy efficient Information Communication Technology (ICT) solutions due to the rise of energy costs. Novel solutions are sought for due to the increasing awareness of the ecological impact of ICT equipment in the environment. For instance, the authors in [1] show that communication networks produced 64 megatons of CO_2 in 2002. Due to the huge demand of Internet services with applications consuming high data rate resources, network capacity has grown very quickly in the past few years. The production of green-house gases by communication networks has expanded very rapidly too. It is interesting to observe that network access equipment is responsible for close to 70% of the overall telecom network energy demand, even though its utilization represents less than 15% [2]. These numbers suggest that research and industry efforts should focus on proposing energy efficient access networks while providing increased access capacity to anticipate future demand.

Time Division Multiplexing (TDM)-Passive Optical Network (PON) (e.g. Ethernet PON (EPON) and Gigabit-capable PON (GPON)) appears to be a promising access technology meeting both requirements: (i) it increases the transmission rate of traditional access technologies up to the order of Gbps and (ii) it consumes much less energy than other access technologies like WiMAX, Point-to-Point Optical Access Network, Fiber-to-the-Node (FTTN) [3].

* Corresponding authors. Addresses: Room T240, Truth hall, KAIST-ICC, Munji-ro 193, Yuseong-gu, Daejeon 305-732, South Korea. Tel.: +82 423506282; mobile: +82 1096855060; fax: +82 428666226 (S.H. Shah Newaz), Office - D05, 9 rue Charles Fourier, 91011 Evry Cedex, France. Tel.: +33160764119; Mobile: +33670793926; Fax: +33 160764578 (Gyu Myoung Lee).

E-mail addresses: newaz@kaist.ac.kr (S.H.S. Newaz), acrumin@it.uc3m.es (Á. Cuevas), gm.lee@it-sudparis.eu (G.M. Lee), noel.crespi@it-sudparis.eu (N. Crespi), jkchoi59@kaist.ac.kr (J.K. Choi).

1389-1286/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.comnet.2013.02.001

Although a TDM-PON is regarded as an energy efficient network compared to other access networks, there is still room for reducing their energy consumption. To improve energy efficiency of TDM-PONs, the effort is concentrated on the Media Access Control (MAC) and the physical layers. In the physical layer, the effort is focused on developing optical transceivers and electronic circuits with lower power consumption and response time. In the MAC layer, most studies propose to apply sleep control and energy-aware scheduling schemes. TDM-PONs can save energy by moving the Optical Network Unit (ONU) to sleep (reducing consumption by turning off power hungry electronic components such as the optical transceiver) when it does not have any incoming or outgoing traffic.

The sleep mode mechanism appears as a good option to save energy since the access traffic is usually bursty [3]. However, the sleep mode approach is challenging in practice because once an ONU is in sleep mode, the Optical Line Terminal (OLT), which is the centralized intelligence in a TDM-PON, cannot communicate with it until that ONU leaves the sleep mode. Therefore, the OLT needs to buffer all the packets while an ONU is in sleep mode. The duration of the sleep interval directly affects the delay experienced by those packets stored in the OLT, since longer sleep interval leads to longer OLT frame storage and thus longer frame delay in the PON-based access network. Therefore, energy saving in PON presents a clear trade-off between energy saving and delay performance.

When an ONU sleeps it loses its connection to the OLT clock and synchronization. After completion of sleep, it takes about 5.125 ms [4] for an ONU to recover its connection to both the OLT clock and synchronization. Therefore, an ONU's sleep interval (usually assigned by the OLT) needs to be long enough for transiting from active mode to sleep mode, then from sleep to active mode, and for recovering connections to the OLT clock and synchronizing with it. Consequently, when the downlink or uplink traffic arrival rate is high and has a strict[1] delay requirement, an ONU cannot move to sleep mode due to the clock recovery and synchronization time [4]. Reducing clock and synchronization time would thus make it possible to efficiently minimize energy consumption in ONUs [4,5].

We found several studies [3,6,7] proposing mechanisms in the MAC layer for managing sleep mode in PON to save energy. In addition, research results have also improved the physical layer of ONUs by defining a new ONU architecture, in particular, for recovering the OLT clock and finishing synchronization quicker, as proposed in [4]. Furthermore, there is ongoing research work on developing optical transceivers and circuitry with low power consumption and low response time [5].

In fact, the actual role of an ONU should be to reduce energy consumption as much as possible while always meeting the access delay requirements imposed by the operator. All previous studies have focused on one or the other of these aspects, but none have dealt with both issues at the same time. In this paper, we propose Adaptive Delay-Aware Energy Efficient (ADAEE) solution that aims to: (i) meet access delay requirements and, (ii) at the same time reduce the energy consumption as much as possible. ADAEE presents two main contributions: (i) a novel ONU architecture and (ii) a novel algorithm to compute the maximum (T_{max}) and minimum (T_{min}) sleep interval values. Both contributions together allow fulfilling operator delay requirements while maximizing energy savings.

We use 24 h traces of real time traffic to evaluate the ADAEE proposal and compare it with two previously proposed solutions. We also evaluate performance of the ADAEE using Constant Bit Rate (CBR) and Variable Bit Rate (VBR) traffic. The results demonstrate that in different scenarios (strict delay and relaxed delay) ADAEE solution is the most energy efficient solution compared to the existing ones while it meets the access delay requirement. Therefore, to the best of our knowledge, the ADAEE is the first proposal that addresses energy efficiency linked to the obligation of fulfilling the required delay for the PON traffic. In addition, we examine how sleep mode affects TCP throughput. The results exhibit that in compare to existing ones (proposals that consider sleep mode to improve energy efficiency of PON) ADAEE shows a best throughput performance.

The remainder of this paper is structured as follows: Section 2 introduces the TDM-PON background and related work. Section 3 describes our proposal, the ADAEE approach, in detail. We present the performance evaluation of our proposal in Section 4 together with a comparison to two earlier proposals from the literature. Finally, Section 5 provides conclusions.

2. Related work

In this section we first briefly describe operation of a TDM-PON, and then we present those related works that influence our proposal classified into two categories: MAC layer and physical layer approaches.

2.1. TDM-PON background

A TDM-PON utilizes a broadcast-and-select mechanism for downlink transmission (from OLT to ONU). In a TDM-PON, downlink frames are transmitted through a fiber and then broadcasted using a passive component called splitter. Usually the splitting ratio is *1:n*, where *n* can be 16, 32, 64 or 128 ONUs. Fig. 1 depicts a generic TDM-PON architecture. In order to identify the destination ONU, the OLT uses a unique identifier for each of them (i.e. 'Alloc ID' in GPON [8] and unique Logical Link Identifier (LLID) in EPON [9]). Furthermore, ONUs do not know when the downlink traffic arrives at the OLT and the time when the OLT schedules the traffic transmission for them. Therefore, IEEE standard 802.3ah [9] specifies that ONUs should keep its optical receiver continuously on in order to capture all downlink frames and check whether they are destined for it or not. Clearly this solution is far from being energy efficient since there are many periods of time in which an ONU does not have any incoming traffic, and thus

[1] In all this paper we use the term strict delay requirement to refer those cases in which the delay in the access network needs to be very low (e.g. 4 ms), while we use the term relaxed delay requirement for longer PON access delay (e.g. 30 ms)).

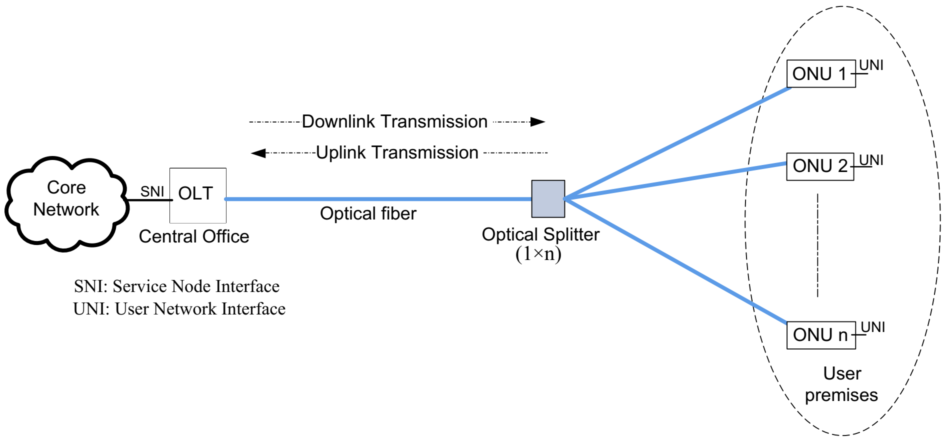


Fig. 1. Generic TDM-PON architecture.

could save energy by turning off the optical receiver and other components. From now on in the paper we will refer to this ONU as *existing solution* since ONUs that implement this standard (e.g. IEEE 802.3ah [9]) are available in the market.

In a TDM-PON system, for uplink transmission, a dedicated time slot is assigned to each of the ONUs by the OLT during an uplink transmission cycle. An ONU that has traffic to transmit in the uplink direction needs to send a request to the OLT indicating the required bandwidth. Then, the OLT collects all request messages and decides what transmission slot for each of the ONUs during the next uplink transmission cycle after executing the so-called bandwidth allocation algorithm (i.e. dynamic or fixed bandwidth allocation). Next, the OLT notifies each of the requester ONUs.

At this point we should notice that most of research proposals (both in the MAC and physical layers proposals) are mainly focusing on the PON downlink part in sleep mode management, as we will see in the next subsections. The main reason is that in a TDM-PON, managing the downlink traffic along with sleep mode management in ONUs is more challenging because an ONU knows when it has something to transmit, but it rarely knows when it will receive the next downlink frame from the OLT. Therefore, although we also consider how PON uplink works in our solution, we devote more effort to define the downlink part.

2.2. MAC layer approaches

To reduce energy consumption, key ideas in the literature are to switch off some components (e.g. Photo Detector (PD), Limiting Amplifier (LA), Continuous Mode Clock and Data Recovery (CM-CDR)) when there is no available uplink or downlink traffic to/from that ONU [3,4]. This state is known as sleep mode. When an ONU moves to sleep mode, the OLT-ONU communication is not feasible until the ONU activates all the components entering in active mode again. Therefore, to check whether any incoming packet is available at the OLT or not, each ONU periodically wakes-up, following a schedule provided by the OLT and listens to the optical medium. If there are any buffered frames in the OLT destined for an ONU, the OLT sends them. If no frames are available, then the OLT schedules the next sleep interval for the ONU, that is, the OLT indicates the ONU when it should enter in active mode again to check downlink traffic availability. For uplink transmissions the OLT assigns a fixed and dedicated time slot to ONUs willing to transmit traffic, so that an ONU can wake up and transmit in the assigned time slot staying in sleep mode until that moment.

To reduce PONs energy consumption, the utilization of the sleep mode has been widely adopted in the literature. However, deciding on sleep intervals for an ONU is indeed a challenging issue since there is a clear trade-off between energy efficiency and Quality of Service (QoS) in terms of delay. A long sleep interval might provide a large energy saving at the price of increasing the PON delay performance. Moreover, the bursty nature of access networks traffic makes difficult to accurately predict the right sleep interval [3]. For instance, after a silent period in an ONU downlink traffic, the OLT could define a long sleep interval to that ONU, and suddenly receive a traffic burst that will experience a higher delay due to wrong prediction of the OLT.

In [6,10], authors propose a three way handshake approach. When an ONU does not have any traffic to receive at the OLT, the OLT computes the sleep interval and sends it to the ONU. Finally the ONU acknowledges the reception of the new sleep interval and then switches into sleep mode. The three way handshake mechanism imposes an extra roundtrip delay [3]. Authors in [10] refer to the limitation of this three way message exchanging mechanism in a TDM-PON having large number of ONUs. When the number of ONUs is large, the goodput from both uplink and downlink is reduced [6]. It must be noted that these proposals require both optical transmitter and receiver to remain on in the ONU during the active period due to the three way handshake. This implies higher energy consumption in the ONU side. As we will see later, other proposals allow turning off the transmitter when the ONU is receiving traffic and does not need to transmit.

For lightly loaded networks, authors in [7] propose to assign fixed bandwidth for each of the ONUs. To enable that, they assign a fixed duration downlink time slot to each ONU (similar approach is followed for the uplink).

However, the limitation of this solution comes from the fact that assignment of fixed slots sometimes causes under-allocation or over-allocation of bandwidth to an ONU, as traffic arrival of an ONU can change dynamically.

Authors in [11] propose a novel algorithm to calculate the sleep interval leading to a more suitable response time in the ONU. This algorithm runs in the OLT side, which computes a new sleep interval every time an ONU wakes up while there is no downlink traffic. It must be noted that ONUs need to acknowledge the reception of the new sleep interval, thus forcing the transmitter to be active every time they wake up. In this mechanism, the initial sleep interval starts with a minimum sleep interval T_{min}. Then, if there is no downlink traffic arrival after each sleep interval ends, the ONU's sleep interval increases exponentially (as described later in Eq. (3)) until it reaches a maximum sleep interval T_{max}.

Authors in [3] introduce a novel mechanism where a common algorithm running on both, ONUs and OLT, calculates the sleep interval. This allows the OLT to be synchronized with the ONUs to know when each of them is available (i.e. in active mode) without requiring any message exchange. When an ONU wakes up, it waits a predefined time to check whether some downlink traffic is available. If it does not receive any frame, it computes the new sleep interval and moves to sleep mode again. The OLT follows the same operation and knows when the ONU will be in active mode again, so that the OLT can store and then forward frames if it receives some traffic directed for that ONU. In particular, authors in [3] mention that an ONU decides to go into sleep mode whenever it does not receive anything from the OLT within one Dynamic Bandwidth Allocation (DBA) cycle, which last 3 ms for each one. Therefore, in this mechanism an ONU spends 3 ms before moving to sleep mode, which can cause significant energy consumption. Contrary to [11], the sleep interval increases much faster in this case. This could lead downlink traffic to experience more delay than in the solution presented in [11]. However, unlike the solution presented in [11], authors in [3] propose to turn off the transmitter during reception periods if there is no uplink traffic available that allows reducing the energy consumption.

ITU-T Recommendation G.sup 45 [12] specifies four types of power saving modes for GPON ONUs. These are: cyclic sleep, power shedding, deep sleep and doze mode. In power shedding, unused User Network Interfaces (UNIs) are turned off, while other three modes are used for reducing energy consumption in the analog circuitry part of an ONU [6]. In cyclic sleep mode, the transmitter and receiver of an ONU are turned off and turned on periodically. Both the transmitter and receiver are turned off in the deep sleep mode. Finally, doze mode proposes to only turn off the transmitter and is thought to save energy in cases in which the ONU is receiving downlink traffic but does not have any outgoing traffic.

2.3. Physical layer approaches

The limitation of the existing ONU architecture is considered in [4]. The existing ONU takes a significant amount of time to finish synchronization and clock recovery (both are required in order to receive and/or transmit traffic) when it wakes-up. This time amounts to 2–5 ms for recovering the clock and 125 μs for the synchronization with the PON network. Two novel ONU architectures for EPON and GPON are presented in [4,13]. These architectures shorten the overhead time (i.e. time required for recovering OLT clock and synchronization) to 125 μs when an ONU moves from sleep to active mode.

A novel ONU architecture is proposed in [4]. This architecture turns off fewer components when initiating sleep mode, presenting a better response time but allowing more power to be consumed in sleep mode (1.28 W compared to the 0.75 W consumed by the existing ONU architecture). In [4], when an ONU goes to sleep mode, all the components except the de-serializer (DMUX), which consumes much more power than any other component, are kept on in the frontend analog circuit part. With this, an ONU can maintain the OLT clock. Since the overhead time is reduced using this mechanism, now an ONU sleep interval can be reduced to a very small amount of time (i.e. less than 5.125 ms), which is not possible in the existing ONU architecture. However, we again face the trade-off between energy and QoS performance because in this novel architecture the ONU consumes 1.28 W in the sleep mode that represents almost the double power consumption than that of the existing ONU architecture. It must be noted that in active mode both architectures consume the same amount of power (i.e. 4.69 W) [14].

Therefore, to the best of our knowledge, no previous work in the literature aims to fit operator delay requirements while being energy efficient at the same time. Therefore, in this paper we propose ADAEE solution that relies on a novel ONU architecture with several modes (including two sleep modes) and an algorithm for finding suitable sleep interval. Hence, our solution utilizes good characteristics from physical and MAC layer solutions together. ADAEE centers its effort on providing a quick ONU response time when the delay requirement is strict, while under 'relaxed delay' requirement it increments the energy saving.

3. Adaptive Delay-Aware Energy Efficient (ADAEE)

ADAEE is a solution that presents two main contributions: (i) a novel ONU architecture that is capable of selecting suitable sleep mode to maximize energy saving and performing all other functionalities of a standard ONU, and (ii) a novel algorithm, which runs at the OLT, to compute the minimum (T_{min}) and maximum (T_{max}) sleep interval values for ONUs.

The main objectives of ADAEE are: (i) satisfying QoS in terms of operator delay requirement, and (ii) maximizing energy savings. As we have already stated there exists a trade-off between energy consumption and delay performance. It is clear that between the two goals, satisfying traffic delay requirements prevails over saving energy, since a PON operator cannot sacrifice users' QoS to reduce energy consumption. However, there might be scenarios in which, due to the traffic nature (e.g. web traffic), the PON access delay could be relaxed and permit longer delays (e.g. 30 ms).

3.1. ADAEE system model

In our solution an ONU can have several modes as related in Section 3.1.2 (e.g. active, sleep, doze, etc.). For managing those different modes we introduce a new element named Mode Controller Logic (MCL). The MCL unit goes through different decision steps and manages all the remaining ONU components for satisfying a specific requirement (e.g. put the ONU in active mode for transmission and reception, turning off some of the components for saving energy in absence of uplink or/and downlink traffic, etc.). ADAEE ONU has two sleep modes. The most important task of the MCL is to decide between two different sleep modes: (i) when an ADAEE ONU loses the OLT clock and works as the existing solution, the power consumption is 0.75 W and the response (or overhead) time is 5.125 ms. We name this mode as Deep Sleep Mode (DSM), and refer to this ONU operational mode as Long Overhead ONU Architecture (LOOA). (ii) We use the ONU architecture as defined in [4]. That means power consumption increases up to 1.28 W in sleep mode. However, response time is reduced down to 125 μs. We refer to this sleep mode as Light Sleep Mode (LSM), and the ONU operational mode is named as Short Overhead ONU Architecture (SOOA).

Next, we present the functional block diagram of the OLT and an ONU in our solution. Then, we briefly explain the different modes that an ONU could adopt in ADAEE. Finally, we describe the decision criteria for the MCL in order to select some of the different modes.

3.1.1. Functional block diagram of OLT and ONU in ADAEE

3.1.1.1. OLT functional block diagram. ADAEE OLT, as previously proposed in [3,6,11], is in charge of computing ONUs sleep interval. Therefore, we presume that the OLT is composed of a Transceiver Unit, an Uplink-Downlink Bandwidth Management Unit, and a Sleep Interval Measurement and Control (SIMC) Unit. Fig. 2a presents an OLT diagram. Next, we describe the main functional units in ADAEE OLT:

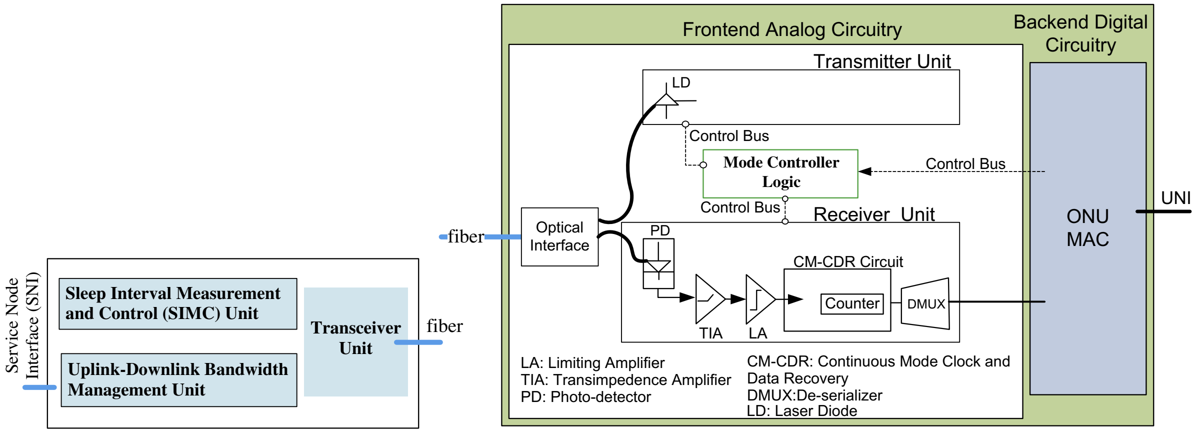
- **Transceiver Unit**: Transceiver unit consists of all the transmission and reception related circuitry, for example laser, avalanche photodiode, Burst Mode CDR (BM-CDR) [5].
- **Sleep Interval Measurement and Control (SIMC) Unit**: The task of SIMC unit is to obtain information of the downlink traffic received through the Service Node Interface (SNI) (see Fig. 1). If the traffic is directed to an ONU that is in sleep mode, then the SIMC unit buffers the traffic in the OLT. Whenever that ONU enters in active mode, the SIMC unit invokes the Uplink-Downlink Bandwidth Management Unit to transmit the buffered traffic. In addition, ADAEE proposes two more functionalities within the SIMC unit: (i) When there is no downlink traffic for an ONU, the SIMC unit decides the values of T_{min} and T_{max} based on the operator policy for access delay requirement and average arrival rate by using a novel algorithm (see Algorithm 1 in Section 3.3.3), which tries to satisfy the established delay requirement. T_{min} and T_{max} are sent to the ONU, and it acknowledges receipt. (ii) The SIMC unit computes the average arrival rate, which is required in the algorithm (i.e. Algorithm 1) to calculate T_{min} and T_{max}, by using a predefined Sample Time Window (STW).
- **Uplink-Downlink Bandwidth Management Unit**: Similarly to [3,6,10,12], it is assumed that the OLT assigns consistently a minimum amount of uplink slot to each sleeping ONU. We define this minimum amount of uplink slot as mini-slot in this paper. Therefore, all sleeping ONUs have the option (all of them have uplink time slots) to request bandwidth allocation for the next DBA cycle to the OLT. Therefore, when a sleeping ONU receives uplink traffic from user premises through UNI, the sleeping ONU can wake up and send bandwidth request to the OLT during the dedicated mini-slot period without waiting for a sleep period to be expired.

3.1.1.2. ONU functional block diagram. Fig. 2b presents ADAEE ONU diagram including all the basic components that can be found in an ONU [3,4,13]. In the analog circuit part there are two main units: Transmitter and Receiver unit. When ADAEE ONU moves to sleep mode, some part of the digital circuitry can remain on, as proposed by [4,13]. For supporting sleep modes an additional component is required, this is a counter [4]. In addition, ADAEE ONU includes a MCL unit that decides when an ONU transits from one mode to another by turning on and off the suitable components. Next subsection presents all possible modes in an ADAEE ONU.

3.1.2. ADAEE ONU mode diagram

Next we briefly explain all the possible modes for an ADAEE ONU. We consider 5 different modes: active mode, transmission mode, doze mode, LSM and DSM. The mode transition diagram of an ADAEE ONU is presented in Fig. 3.

- **Active Mode**: In this mode an ONU can transmit and receive frames. This case is the most power hungry one (all components are on), and consumes 4.69 W [14]. MCL unit selects this mode when an ONU needs to transmit and receive at the same time.
- **Doze Mode**: Only the receiver unit is on in an ONU [3,12,14]. Hence, only reception is possible [12,14]. An ONU enters in this mode when there is no traffic for uplink transmission but it needs to listen to the downlink channel to receive traffic from the OLT. In this case the ADAEE ONU consumes 1.7 W. MCL unit selects this mode when only downlink reception exists. For instance, in the proposed ADAEE, when an ONU enters in sleep mode it needs to listen whether there is downlink traffic available or not after each sleep interval expires. Therefore, MCL unit moves an ONU in doze mode for checking the downlink availability and returns to the sleep mode again (e.g. LSM or DSM) if there is nor downlink neither uplink traffic. While in case of downlink traffic availability the MCL unit keeps the ONU in doze mode. The MCL unit could move the ONU from doze to active mode if, while receiving downlink frames, there is some uplink traffic ready and the time reaches the assigned uplink time slot (see Fig. 3).



(a) ADAEE OLT. (b) ADAEE ONU.

Fig. 2. Functional block diagram of an OLT and ONU in ADAEE.

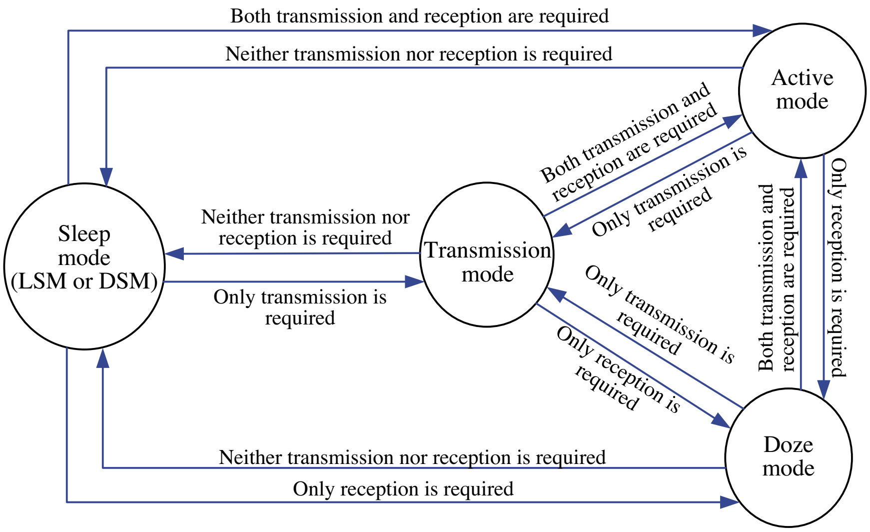


Fig. 3. ADAEE ONU mode transition diagram.

- **Transmission Mode**: An ONU enters in this mode when it has uplink frames (including signaling messages) available and it is within the assigned time slot, keeping the receiver off [3].
- **Sleep Mode (LSM or DSM)**: This mode is selected in the absence of downlink and uplink traffic. When the OLT detects that there is no downlink traffic for an ONU, OLT's SIMC unit computes T_{min} and T_{max} and it sends both values to that ONU. In reply, an acknowledgment message is sent (see Fig. 7) from the ONU to the OLT. At this point MCL unit of the ONU needs to calculate jth sleep interval (T_j) using T_{min} and T_{max} (T_j calculation is described later in Eq. (3)). Note that the OLT also uses Eq. (3) to know when the ONU will be available to receive downlink traffic. This avoids the need to exchange messages between the ONU and the OLT when there is no downlink traffic, thus allowing ONUs to decouple the activation of the transmitter from the activation of the receiver (e.g. doze mode).

In our solution, before an ONU enters into sleep mode it needs to decide what kind of sleep mode (i.e. LSM or DSM) is suitable for this sleep interval based on a threshold value. Note that MCL unit should consider the trigger time for moving from sleep mode to active/doze/transmission mode considering the overhead time required for each level of sleep mode (i.e. LSM takes 125 µs, DSM takes 5.125 ms) (see Fig. 4). Section 3.1.3 explains how this

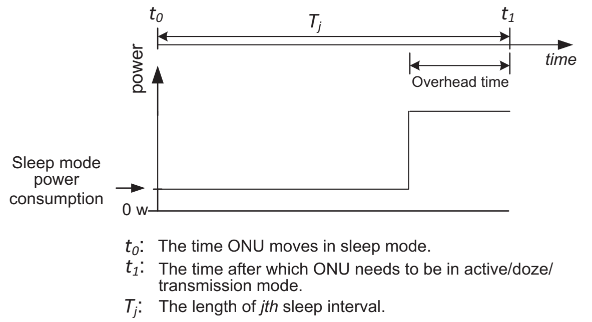


Fig. 4. Sleep mode to active/doze mode triggering time consideration in the MCL unit of an ADAEE ONU.

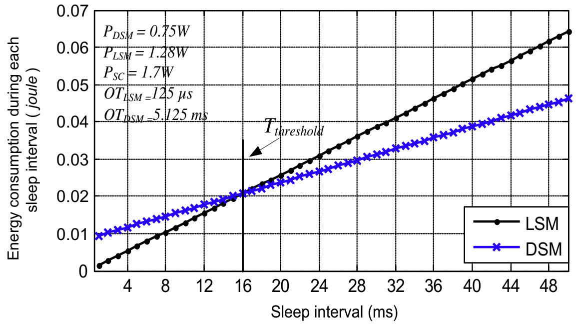


Fig. 5. Finding the $T_{threshold}$ for ADAEE.

threshold value can be found and how the MCL makes decision for choosing either LSM or DSM for sleep mode. Next, we describe LSM and DSM:

- **Light Sleep Mode (LSM)**: When an ONU enters in this mode, only the DMUX is turned off. While CM-CDR, APD photo diode, TIA and LA are left on so that the ONU does not lose the OLT clock [4]. In this mode the ONU consumes 1.28 W and requires only 125 μs to be able to receive traffic from the OLT. The improvement in the response time happens because the ONU never loses the OLT clock, and thus when it enters into active/doze/transmission mode it only needs to synchronize with the OLT.
- **Deep Sleep Mode (DSM)**: In this case all components DMUX, CM-CDR, APD photo diode, TIA, LA and counter are turned off during the sleep mode. In this mode the ONU consumes only 0.7 W. When entering in DSM, the ONU loses OLT clock. Consequently, when it returns to active/doze/transmission mode it first needs to get the OLT clock and later complete synchronization, which requires 5.125 ms.

3.1.3. Decision criteria for selecting DSM or LSM in ADAEE ONU

We define a threshold value for sleep mode selection ($T_{threshold}$) to select either LSM or DSM for sleep interval time. If the jth sleep interval $T_j \leqslant T_{threshold}$, MCL unit puts the ONU in LSM. Otherwise, it moves the ONU to DSM. $T_{threshold}$ is decided as follows. Let us consider OT_{LSM}, OT_{DSM}, P_{LSM}, P_{DSM} and P_{SC} to be the overhead time in LSM, overhead time in DSM, power consumed in LSM, power consumed in DSM and power consumed when an ONU tries to gain OLT clock or complete PON synchronization, respectively. It must be noted here that when an ONU needs to recover OLT clock and PON synchronization it can extract them from the downlink traffic by using its receiver unit [4,17,18]. Therefore, during this clock recovery and PON synchronization time an ONU turns on receiver unit only and spends as much power as it spends in doze mode (P_{SC} = 1.7 W). Then, the energy consumed during T_j sleep interval if LSM is selected is given by:

$$E^{LSM}_{T_j} = (T_j - OT_{LSM}) * P_{LSM} + OT_{LSM} * P_{SC}. \quad (1)$$

Energy consumed during T_j sleep interval when MCL unit selects DSM is expressed as follows:

$$E^{DSM}_{T_j} = (T_j - OT_{DSM}) * P_{DSM} + OT_{DSM} * P_{SC}. \quad (2)$$

Then $T_{threshold}$ is computed as follows:

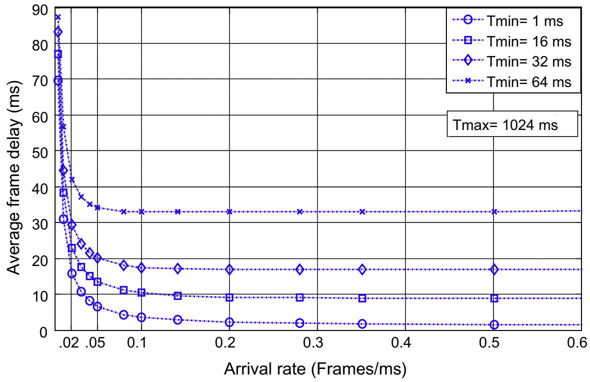
$$T_{threshold} = T_j, if \frac{E^{LSM}_{T_j}}{E^{DSM}_{T_j}} = 1, \text{ where } j = 1, 2, \ldots, n.$$

Fig. 5 compares the energy consumed during a sleep interval for DSM and LSM under different values of the sleep interval. This allows us to find the $T_{threshold}$ that discriminates from where we should use between DSM and LSM. Therefore, if $T_j \leqslant T_{threshold}$, MCL unit selects LSM that leads to higher energy saving. In contrast, if $T_j > T_{threshold}$, ADAEE ONU will enter into DSM, i.e. this case is the most efficient in terms of energy. It must be noted that selecting LSM for any sleep interval $T_j \leqslant T_{threshold}$ is a wise decision. The reason behind this is that an ADAEE ONU will end up spending less energy than in the case of selected DSM when $T_j \leqslant T_{threshold}$. Besides, ADAEE ONU can move to sleep mode for any small sleep interval (e.g. 1 ms) since the overhead time in case of LSM is very small (i.e. 125 μs).

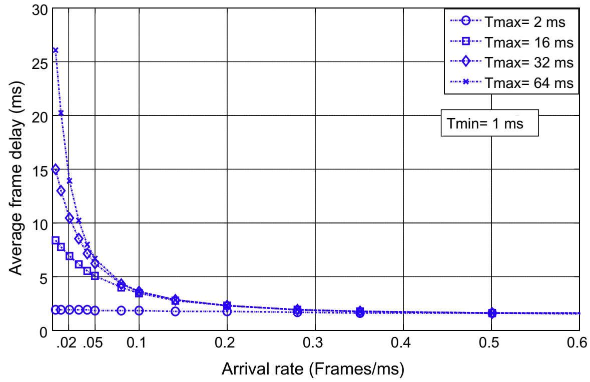
3.2. Algorithm to compute ONUs sleep interval

ADAEE algorithm to compute the sleep interval is inspired by the algorithm[2] used in [11]. This algorithm runs in both the OLT and ONU. Therefore the OLT decides the values for the maximum (T_{max}) and minimum (T_{min}) allowed sleep intervals (as described later in Section 3.3) and informs the ONU. The ONU acknowledges receipt of these two values. Following the exponential function shown in Eq. (3), the ONU and OLT can easily measure what would be the sleep interval after each listening interval, as long as there is no downlink traffic arrivals (in case there is a new arrival new T_{min} and T_{max} values are computed in the OLT and sent to the ONU) using the same T_{min} and T_{max} values. Therefore, contrary to [11], in our solution ONU and OLT do not need to exchange any messages until some downlink and/or uplink traffic is

[2] Previously the concept of incrementing sleep interval from T_{min} to T_{max} exponentially has been widely used for measuring sleep interval for mobile terminals in wireless domain [15].



(a) Influences of T_{min} on downlink frame delay.



(b) Influences of T_{max} on downlink frame delay.

Fig. 6. T_{min} and T_{max} influence on PON downlink frame delay.

available, since both of them know already the sleep pattern of the ONU. Next, Eq. (3) presents the algorithm:

$$T_j = \begin{cases} 2^{j-1}T_{min}, & \text{if } 2^{j-1}T_{min} < T_{max} \\ T_{max}, & \text{Otherwise}. \end{cases} \tag{3}$$

where T_j denotes the length of the jth sleep interval. That is for example with T_{min} = 1 ms, the first sleep interval (j = 1) will be $T_1 = 1 \cdot 2^{1-1} = 1$ ms, and the second sleep interval (j = 2) will be $T_2 = 1 \cdot 2^{2-1} = 2$ ms. The sleep interval gradually increases up to T_{max}.

3.3. Novel algorithm for finding T_{min} and T_{max} in ADAEE

In this section we present a novel algorithm in which the ADAEE OLT computes T_{min} and T_{max} taking into account the ONU average downlink arrival rate (λ) and the delay requirement (D_T) specified by the PON operator. After that the OLT sends T_{min} and T_{max} to the ONU that computes the next sleep interval using Eq. (3). Authors in [11] also use Eq. (3) to compute the sleep interval, but they always use fixed T_{min} and T_{max} values. As it has been stated above, the decision of T_{min} and T_{max} is essential in ADAEE since it is used to compute the sleep interval, which in turns is used by the ONU MCL unit to select which sleep mode will be used by the ONU. However, in order to fully understand our algorithm we first need to understand how T_{min} and T_{max} influence PON delay, and then present a model for the PON average frame delay.

3.3.1. TDM-PON downlink frame delay

We assume that an ONU's listening interval (L) has fixed length for checking downlink traffic availability. Let the length of the jth sleep interval be T_j. Then, $T_j + L$ is the sum of one sleep interval and the next listening interval for a particular ONU. Let e_j denotes an event indicating that one or more frames have arrived at the OLT during the monitoring period j with an arrival rate λ. Then from [15], we get the following equations.

$$\begin{aligned} &Pr(e_j = true) = 1 - e^{-\lambda(T_j+L)} \\ &Pr(n = 1) = Pr(e_1 = true) = 1 - e^{-\lambda(T_1+L)} \end{aligned} \tag{4}$$

$$\begin{aligned} Pr(n = j) &= Pr(e_1 = false; \ldots; e_{j-1} = false; e_j = true) \\ &= \prod_{i=1}^{j-1} Pr(e_i = false) Pr(e_j = true) \\ &= e^{-\lambda \sum_{i=1}^{j-1} T_i + L} \{1 - e^{-\lambda(T_j+L)}\}. \end{aligned} \tag{5}$$

Let us consider that the notation $E[\cdot]$ represents average value. Then for a given set of values T_{min}, T_{max} and λ, the average delay experienced by the downlink frames can be computed as follows [15]:

$$E[Frame\ delay] = \sum_{j=1}^{\infty} Pr(n = j)(T_j + L)/2$$

$$E[Frame\ delay] = 1/2 \left\{ \sum_{j=1}^{\infty} e^{-\lambda \sum_{i=1}^{j-1} T_i + L}(T_j + L) - \sum_{j=1}^{\infty} e^{-\lambda \sum_{i=1}^{j} T_i + L}(T_j + L) \right\}. \tag{6}$$

where T_j is obtained from Eq. (3), which relies on T_{min} and T_{max} values to compute T_j.

3.3.2. Discussing T_{min} and T_{max} influence on downlink frame delay

Fig. 6 plots Eq. (6), which defines TDM-PON downlink frame delay, for λ going for 0.001 frames/ms to 0.6 frames/ms. In particular, in Fig. 6a we compare the delay for different T_{min} values and keep T_{max} = 1024 ms. In Fig. 6b we keep static T_{min} = 1 ms and show results for different T_{max} values.

We can notice from Fig. 6a that T_{min} does not influence the delay for low λ values (below 0.02 frame/ms). That is, for low λ (below 0.02 frame/ms), frame delays are similar, independently of the T_{min} values. However, for larger λ values, the higher T_{min} is, the more is the delay experienced by PON frames. Therefore, T_{min} has special importance in those scenarios with medium and high data rates, which are especially interesting for optical network technologies.

T_{max} has an opposite behavior (see Fig. 6b). For low λ values (e.g. below 0.1 frames/ms) the higher T_{max} is, the higher the frame delay. However, the results for high λ values are similar, independently of the T_{max} values. Therefore, tuning T_{max} has special importance in cases presenting low downlink traffic.

3.3.3. ADAEE algorithm for T_{min} and T_{max}

From previous discussion on how T_{min} and T_{max} influence the delay, we conclude that we should deal with both of them but not at the same time. When the OLT tunes T_{min}, it keeps T_{max} fixed, and vice versa. The decision to control T_{max} or T_{min} comes from a threshold λ_T on the frames arrival rate. Then when the arrival rate is above λ_T we play with T_{min} otherwise we tune T_{max} to satisfy the delay requirement while minimizing energy consumption.

As mentioned, this algorithm needs to estimate the arrival rate. For that, the ADAEE OLT SIMC unit counts the number of arrivals during a defined STW for an ONU (e.g. ONU-k), and computes the ONU-k's average downlink frame arrival rate (λ) that will be used as estimation of future arrival rates. Therefore, the proposed algorithm measures T_{min} and T_{max} for the present time ($t_{present}$) based on the arrival rate estimation. Next, we define some parameters that are used in our algorithm:

D_T: The delay requirement defined by the operator for downlink traffic (e.g. 4 ms, 10 ms, 30 ms).

λ_T: The arrival rate decision threshold.

$T_{max\_Threshold}$: The maximum amount of time that can be assigned to an ONU for sleep mode considering physical layer and PON protocol constrains.[3]

$T_{min\_Threshold}$: The minimum amount of time that can be assigned to an ONU for sleep mode considering physical layer and PON protocol constrains.[4]

$R\,\{T_{max\ 1}, T_{max\ 2}, \ldots, T_{max\ n}\}$: A set containing all the values of T_{max}, where $T_{max\ i} < T_{max\ i+1}$ and $T_{max\ n} = T_{max\_Threshold}$.

$Q\,\{T_{min\ 1}, T_{min\ 2}, \ldots, T_{min\ n}\}$: A set containing all the values of T_{min}, where $T_{min\ i} < T_{min\ i+1}$, $T_{min\ 1} = T_{min\_Threshold}$ and $T_{min\ n} = T_{max\_Threshold}$.

$T_{Last\ arrival,\ ONU\text{-}k}$: The time when the last packet arrived for an ONU (e.g. ONU-k) at the OLT.

$SS_{Threshold}$: The time between the last packet arrival and the moment when the OLT decides to invoke an ONU to enter into sleep mode.

The following is the algorithm that is used in ADAEE to compute T_{min} and T_{max}.

Algorithm 1: **T_{min} and T_{max} measurement**

/* Input parameters*/
λ_T, D_T, $T_{min\_Threshold}$, $T_{max\_Threshold}$, λ /* λ is a global variable*/
/* Output parameters*/
T_{min}, T_{max}

while $t_{present}$ - $T_{Last\ arrival,\ ONU\text{-}k} \geq SS_{Threshold}$
  If $D_T \leq Strict\_delay\_requirement$
    $T_{min} = T_{min\_Threshold}$;
    T_{max} = *get_Tmax ();* /*Invoke *Get T_{max} function**/
  end if
  If $D_T > Strict\_delay\_requirement$
    If $\lambda > \lambda_T$
      $T_{max} = T_{max\_Threshold}$;
      T_{min} = *get_Tmin ();* /*Invoke *Get T_{min} function**/
    end if
    If $\lambda \leq \lambda_T$
      $T_{min} = T_{min\_Threshold}$;
      T_{max} = *get_Tmax ();* /*Invoke *Get T_{max} function**/
    end if
  end if
end while

Get T_{min} function: **Sub-algorithm for finding suitable T_{min}**

/* Input parameters*/
$T_{max\_Threshold}$, Q
While $Dt_{Required} \neq Dt_{caculate}$
  $Dt_{calculate} = f1\,(T_{min\ i}, T_{max\_Threshold})$; /*Invoke *f1 function**/
  $i=i+1$;
end while
return $T_{min\ i}/2$; /*Return this value to *Algorithm 1**/

Get T_{max} function: **Sub-algorithm for finding suitable T_{max}**

/* Input parameters*/
$T_{min\_Threshold}$, R
While $Dt_{Required} \neq Dt_{caculate}$
  $Dt_{calculate} = f1\,(T_{min\_Threshold}, T_{max\ i})$; /*Invoke *f1 function**/
  $i=i+1$;
end while
return $t_{max\ i}$; /*Return this value to *Algorithm 1**/

f1 function: **Sub-algorithm for calculating delay**

/*Input parameters*/
L, $T_{min\ f1}$, $T_{max\ f1}$

/*The calling function passes two parameters. Then, those parameters are used here for assigning value of $T_{min\ f1}$ and $T_{max\ f1}$. */

/*Use Eq. (6) to calculate delay*/

$$Delay = \frac{1}{2}\left\{\sum_{j=1}^{\infty} e^{-\lambda \sum_{i=1}^{j-1} T_i + L}(T_j + L) - \sum_{j=1}^{\infty} e^{-\lambda \sum_{i=1}^{j} T_i + L}(T_j + L)\right\}.$$

/* where T_j is calculated using following expression (i.e. Eq. (3))*/
/*$T_j = min\,(2^{j-1} \times T_{min\ f1}, T_{max\ f1})$;*/

/* where T_i is calculated using following expression (i.e. Eq. (3))*/
/*$T_i = min\,(2^{i-1} \times T_{min\ f1}, T_{max\ f1})$;*/

return Delay ; /*Return this value to the calling function*/

[3] An ONU should wake up, even if there is no uplink or downlink traffic available, in order to send a report message to the OLT in EPON [3]. According to the 1G EPON and 10G EPON, the inter report sending time should not be more than 50 ms [9,16]. This prevents OLT's watchdog timer from expiring. Therefore, although there is no data traffic for transmission or reception, an ONU cannot stay in sleep mode more than 50 ms.

[4] In the case of the current ONU architecture, T_{min} cannot be less than 5.125 ms.

First of all, the SIMC unit checks whether $SS_{Threshold}$ has been exceeded or not for a particular ONU-k. If $SS_{Threshold}$ has been exceeded (i.e. there are no arrivals for the ONU-k during that time) our algorithm calculates T_{min} and T_{max} for that ONU. The Algorithm 1 checks whether the current delay requirement is a strict delay requirement or not. If the current delay requirement is a strict one, the algorithm plays with T_{max} while T_{min} is assigned the value of $T_{min\_Threshold}$. Then, Algorithm 1 invokes *Get* T_{max} *function* to find the value of T_{max}. The *Get* T_{max} *function* evaluates all the values of T_{max} contained in set R by using function *f1* that implements Eq. (6) with parameters $T_{min\_Threshold}$, $T_{max\ i}$. The process continues until a suitable T_{max} is found. Finally, the *Get* T_{max} *function* returns this suitable T_{max} to Algorithm 1. That means the T_{max} that saves more energy and at the same time makes frame delay to be below the delay requirement.

If the current delay requirement is not a strict delay requirement, Algorithm 1 tries to find suitable T_{min} and T_{max} values (but not at the same time). Considering the influence of T_{min} and T_{max} on the downlink frame delay performance, if the computed arrival rate λ is higher than λ_T, Algorithm 1 tries to find a suitable T_{min} from the set Q, using the *Get* T_{min} *function*. In this case, the value of T_{max} is set to $T_{max\_Threshold}$, then, the *Get* T_{min} *function* is invoked. This function uses function *f1*, implementing Eq. (6) to calculate frame delay for each element of Q. Finally, it returns suitable value of T_{min} that can save more energy and ensure that most of traffic will be lower or equal to the delay requirement defined by a PON operator.

If current delay requirement is not a strict delay requirement and the current arrival rate is less than or equal to λ_T, the proposed Algorithm 1 first sets $T_{min} = T_{min\_Threshold}$ and then invokes the *Get* T_{max} *function* to find suitable T_{max} from the set R.

3.3.4. Signaling messages

In order to convey T_{min} and T_{max} information from the OLT to the ONU and send an acknowledgment message back, it is needed to consider signaling messages. Both EPON and GPON have their own signaling message used for purposes like bandwidth allocation. Similar to [11], we assume that for EPON Multipoint Point Control Protocol and for GPON Physical Layer Operation, Administration and Maintenance messages can be utilized in our proposed solution.

3.4. ADAEE system operation

In this section we put all the different contributions together to explain the full functionalities of our proposal. For a better explanation of the procedure, we assume a starting point in which an ONU-k (e.g. k_{th} ONU of a TDM-PON) is receiving downlink traffic from the OLT.

1. The OLT continuously monitors ONU-k's arrival rate by averaging ONU-k traffic under a pre-established STW.
2. ***A***. If the SIMC unit in the OLT finds that there is no downlink traffic for an ONU-k, it waits until it exceeds $SS_{Threshold}$ amount of time as shown in Fig. 7. Then, SIMC unit invokes ONU-k to move to sleep mode indicating the values of T_{min} and T_{max} (Algorithm 1 is utilized to compute T_{min} and T_{max}). After receiving these two values, ONU-k sends an acknowledgment message. At this stage, both OLT and ONU run the same algorithm (see Eq. (3)) to calculate the sleep interval. So both of them know when ONU-k will be available again to receive traffic.
   a. If there is no uplink traffic in the ONU, the MCL unit decides to put the ONU into sleep mode. At this point, MCL unit decides what kind of sleep mode (LSM or DSM) the ONU should take. After this decision is made, MCL unit takes the ONU in sleep mode.
   b. Otherwise, if uplink transmission is required, MCL unit puts ONU-k in transmission mode.

   B. If there is any downlink traffic in OLT's buffer, OLT transmits the buffered traffic for ONU-k. After the stored traffic is transmitted, OLT follows step 2A.
3. After ONU-k enters in sleep mode, the counter component notifies the MCL unit when the sleep interval ends, similarly to [4]. Then, MCL unit moves ONU-k to doze mode for listening whether downlink traffic is available or not. If downlink traffic is available, MCL unit keeps the ONU receiving downlink traffic. Otherwise, the MCL unit follows Eq. (3) to calculate the new sleep interval for ONU-k (without exchanging any messages with the OLT). Note that when the ONU requires listening to OLT's downlink frames (if any), the MCL unit decides whether the ONU will stay in doze mode (transmitter off) or fully active mode. Finally, it must be highlighted that as long as any new downlink traffic is available, the ONU and OLT use Eq. (3) to calculate the sleep interval without updating T_{min} and T_{max}.

4. Performance evaluation

In this section we present a comprehensive evaluation of ADAEE and compare our solution with the proposal in [11]. In contrast to ADAEE, the solution in [11] relies on fixed T_{min} and T_{max} values for computing the sleep interval for an ONU without considering at all the operator delay requirement. However, both ADAEE and [11] follow the same exponential sleep interval increment policy (see Eq. (3)). Therefore, we refer the solution presented in [11] as Fixed T_{min} and T_{max} Scheme (FTS). Moreover, FTS could be employed on top of two different physical ONU architectures. The first one would be the existing ONU architecture that has been named as Long Overhead ONU Architecture (LOOA), therefore, we refer to this solution as FTS-LOOA. The second architecture is that one proposed in [4] that presents a much smaller overhead time at the price of consuming more power in sleep mode. We previously named this architecture as Short Overhead ONU Architecture (SOOA), so, in the performance evaluation this second option is referred as FTS-SOOA. It must be noted that the FTS-LOOA uses a Deep Sleep Mode but has a longer response time, thus favouring energy savings over QoS. By contrast, the FTS-SOOA provides good response time by using a Light Sleep Mode. Both these solutions are very suitable to evaluate ADAEE, since one of them seems a good alternative to achieve a high energy saving, while

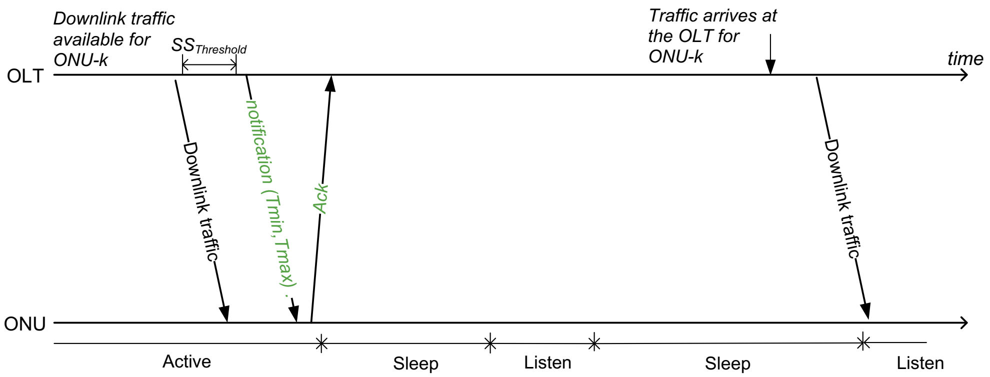


Fig. 7. Sleep mode related signaling between the OLT and an ONU in ADAEE.

the other one will provide a good QoS in terms of low delay. Note that ADAEE has been designed to reach both requirements at the same time: meeting PON operator delay requirements and maximizing energy savings.

We evaluate delay and energy consumption for these three approaches. We compute the delay experienced by downlink frames under strict and relaxed delay requirements. In addition, we measure the energy consumption as the portion of energy that each solution expends compared to the original solution that keeps ONUs always on (e.g. IEEE 802.3ah). We name this solution as Always On Solution (AOS).

We compare the three solutions under three different types of traffic: CBR traffic, VBR traffic, and real traffic traces. This last type of traffic is especially important since, to the best of our knowledge, this is the first work that evaluates a TDM-PON energy efficient proposal with real traffic traces. In order to record real traffic, we captured both uplink and downlink traffic of eight home users and eight office users during 24 h (it must be noted that there can be up to 16 users served by each ONU [19]). We used NS-2 simulator to obtain delay and energy results when CBR and VBR traffic is considered for evaluating performance of these three solutions. In addition, using this simulator we have also evaluated how the sleep mode used in these energy efficient proposals impacts the throughput of TCP flows. However, for evaluating performance of these three solutions under real traffic traces, we used an ad hoc C++ discrete event simulator.[5]

4.1. Delay performance and energy consumption evaluation

In this section, we first compare performance of ADAEE against FTS-SOOA and FTS-LOOA for CBR and VBR traffic, and then compare them under real network traffic traces. In our simulation, we use standard parameters defined for 1G EPON. Table 1 summarizes the values of the parameters utilized in our simulations.

4.1.1. Performance evaluation using CBR traffic

To emulate CBR traffic we used voice data flows using G.711 (an international standard for encoding audio) that generates 1 packet of 160 bytes every 20 ms (i.e. 50 packets/s) [20]. We conducted our simulation during 350 s. We started the simulation with 3 flows and we kept adding voice conversations until we reached 30 flows in the last 50 s. Fig. 8 shows the number of flows that were active over the simulation. Finally, it must be noted that real-time audio traffic delay requirement is considered to be 4 ms in a TDM-PON [21]. Hence, we evaluated the ADAEE, FTS-LOOA and FTS-SOOA under this very strict delay requirement (i.e. $D_T = 4$ ms).

Fig. 9 shows the maximum and average frame delay when there exists 3 CBR flows and 30 CBR flows for FTS-SOOA (Fig. 9a and b, respectively), FTS-LOOA (Fig. 9c and d, respectively) and ADAEE (Fig. 9e and f respectively). When the traffic load is light (only 3 CBR flows), FTS-SOOA and FTS-LOOA yield a higher frame delay than the delay generated by ADAEE. It can be observed from Fig. 9e that the maximum delay values for all 3 flows in ADAEE always less than the imposed delay requirement value (i.e. all of the frames satisfy delay requirement in ADAEE). Whereas, it can be noticed from Fig. 9a and c that in FTS-SOOA and FTS-LOOA some of the frames violate delay requirement (e.g. the maximum frame delay experienced by the frames of flow 1 in FTS-LOOA is around 11 ms). This is because FTS-SOOA and FTS-LOOA keep increasing sleep interval during the silent periods (when there is no CBR traffic) without considering delay requirement ($D_T = 4$ ms). Therefore, when a new frame arrives at the OLT, it incurs a delay that exceeds the operator delay requirement. By contrast, proposed ADAEE takes into consideration the PON operator delay requirement when deciding on the sleep interval. ADAEE limits the sleep interval increment by tuning the value of T_{max} in such a way, so every single frame falls under the delay requirement. However, we can notice from Fig. 9a that FTS-SOOA performs better than FTS-LOOA for both maximum and average delay of each flow. The reason is that FTS-SOOA selects small T_{min} (i.e. 1 ms), while FTS-LOOA selects larger T_{min} (i.e. 6 ms) due to having long overhead time.

The performance of these three solutions is presented in Fig. 9b, d and f when there are 30 CBR flows. In this case, we present average and maximum delay results of a subset of those 30 CBR flows (we select results of 8 CBR flows randomly). In this case, in all of the three solutions, a destination ONU will be busy receiving CBR traffic from the OLT and rarely has a chance to have longer sleep interval. How-

[5] The C++ discrete event simulator, which is developed in our work for TDM-PONs, gives more scalability to deal with real traffic traces than NS-2.

Table 1
Simulation parameters.

Description	Values
Power consumption in active mode	4.69 W [14]
Power consumption in doze mode	1.7 W [14]
Power consumption in DSM	0.75 W [4]
Power consumption in LSM	1.28 W [4]
Overhead time in LSM	125 μs [4]
Overhead time in DSM	5.125 ms [4]
Value of T_{min} in FTS-SOOA	1 ms
Value of T_{min} in FTS-LOOA	6 ms
Value of T_{max} in FTS-SOOA	50 ms
Value of T_{max} in FTS-LOOA	50 ms
$T_{threshold}$	16 ms
λ_T	0.05 frames/ms
$T_{max\_Threshold}$ (for proposed ADAEE)	50 ms
$T_{min\_Threshold}$ (for proposed ADAEE)	1 ms
Downlink and uplink data rate	1 Gbps
Transmission delay between an ONU and the OLT	0.2 ms [6]
STW	10 s
Uplink DBA cycle length	2 ms

ever, several factors influence their performances: (1) T_{min} and (2) three way handshaking. Note that, during high arrival situation in all three solutions, there will be a frame to receive after each sleep interval. Therefore, the length of T_{min} plays a role in frame delay again. In FTS-SOOA, T_{min} = 1 ms, while in FTS-LOOA uses T_{min} = 6 ms. Therefore, FTS-SOOA always performs better than FTS-LOOA in terms of delay. Again in this scenario, for both maximum and average delay of each flow ADAEE appears as the best solution (see Fig. 9f), even though it uses the same T_{min} than FTS-SOOA. Then, the extra delay shown by FTS-SOOA in comparison with ADAEE comes from the required three way handshake message exchange (1.6 ms) between the OLT and the ONU whenever an ONU wakes up from sleep mode. By contrast in ADAEE, an ONU can receive traffic after returning from sleep mode (if there is any). To understand the aggregate results, Fig. 10a shows the frame delay Cumulative Distributed Function (CDF) for each solution during those 350 s. When ADAEE is in place, 100% of the frames meet the delay requirement. This percentage is reduced to 95% for FTS-SOOA, and in the case of FTS-LOOA, only 43.5% of the frames satisfy the delay requirement. This evaluation rejects FTS-LOOA as a practical solution since roughly half voice calls (when there are 30 in parallel) would experience bad QoS.

In Fig. 10b we show the portion of energy that each solution expends compared to the worst case (in terms of energy) in which ONUs remain always active (i.e. AOS). We evaluate the energy consumption for different number of flows. ADAEE greatly outperforms the other solutions. The energy consumption for our solution ranges between 30% to 35% of the maximum energy expenditure, while FTS-SOOA always starts at 75% and increases until it reaches 7 flows, where its energy consumption is saturated at 85%. Finally, FTS-LOOA appears as the most energy consuming solution that in all the cases consumes 90% of the energy compared to the AOS.

In summary, under CBR traffic, ADAEE is the solution that offers better QoS and the most energy efficiency at the same time.

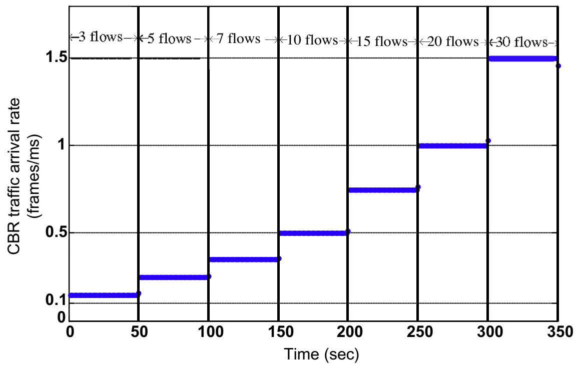


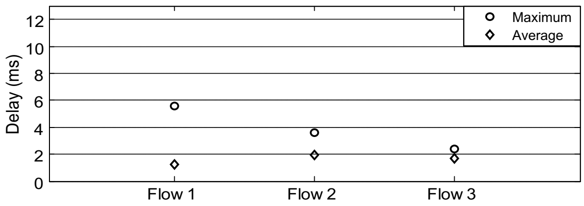
Fig. 8. Arrival rate of CBR traffic during 350 s.

4.1.2. Performance evaluation using VBR traffic

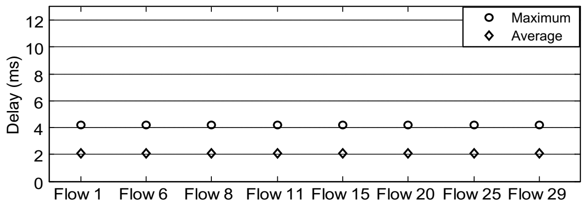
We use the widely adopted ON/OFF source model to generate VBR traffic as considered in [22,23]. It is assumed that the average ON time is 350 ms during which a source generates a frame every 10 ms, while the average OFF time is 650 ms as proposed in [23]. Furthermore, similar to [23] it is considered that the ON period and the OFF period follow exponential distribution. Finally, we establish a relaxed delay requirement of 30 ms (D_T = 30 ms) as assumed for VBR traffic in [24]. As D_T = 30 ms is a relaxed delay requirement, the proposed Algorithm 1 favors saving energy while satisfying delay requirement of VBR traffic flows. As we did for CBR traffic, we conducted simulation for 350 s and during that time the number of flows for a particular ONU goes from 3 to 30, following the pattern shown in Fig. 11.

Fig. 12a and b show the average and maximum frame delay for FTS-SOOA under 3 and 30 flows respectively. Fig. 12c and d do the same for FTS-LOOA. Fig. 12e and f represent the average and maximum frame delay respectively for ADAEE. In this scenario, in all three solutions the first frame after a silent period[6] of VBR traffic experiences longer delay than the following frames generated during a particular ON period of a VBR source. It is because during the long silent period, an ONU will have more chance to take longer sleep (no downlink traffic, hence sleep interval increases). It must be noted that in case of FTS-SOOA and FTS-LOOA after the first frame, all the following frames until the next silent period experience less delay than the proposed ADAEE. The reason is that when operator delay requirement is relaxed, ADAEE tunes T_{min} and T_{max} (but not at the same time) to privilege energy saving by sacrificing QoS. As arrival rate is $\lambda \geqslant \lambda_T$ (see Fig. 11), in this case ADAEE tunes T_{min}. Unlike the previous strict delay requirement scenario, here ADAEE uses longer T_{min}; so that, an ONU can be kept in sleep mode longer to save more energy. As a result, all the following frames after the first frame experience longer delay in ADAEE than in the other two solutions. We can notice from the Fig. 12a, c, and e that average frame delay of each flow in FTS-SOOA and FTS-LOOA is less than the average frame delay in ADAEE. Whereas, in all these three solutions the results of maximum frame delays of those 3 VBR flows are

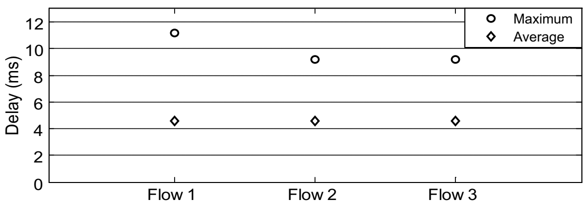
[6] The silent period is the time when all the VBR sources are in OFF state (i.e. generating no VBR traffic).



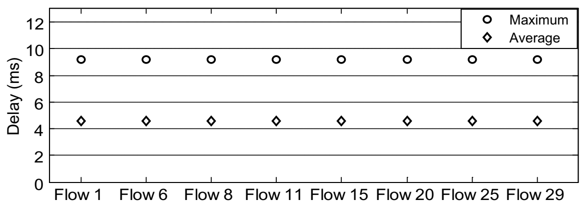
(a) FTS-SOOA with 3 CBR flows.



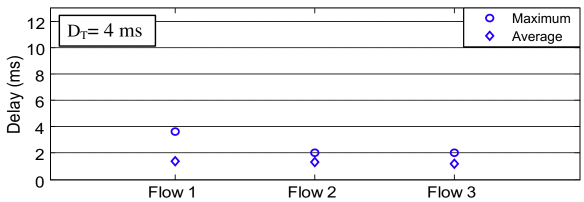
(b) FTS-SOOA with 30 CBR flows.



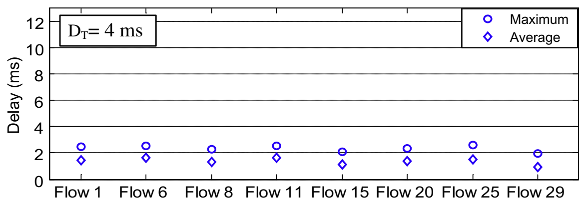
(c) FTS-LOOA with 3 CBR flows.



(d) FTS-LOOA with 30 CBR flows.

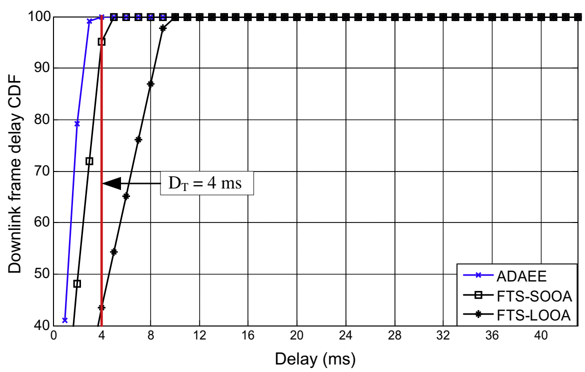


(e) ADAEE with 3 CBR flows.

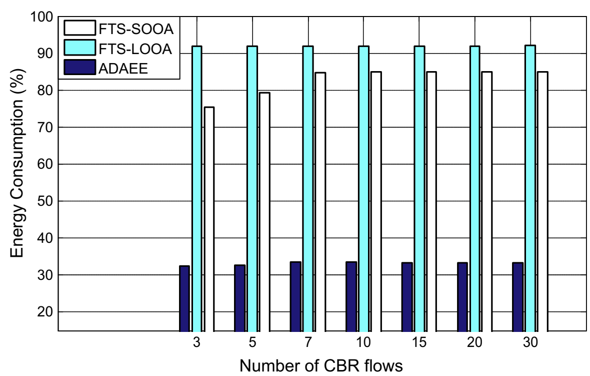


(f) ADAEE with 30 CBR flows.

Fig. 9. Performance comparison with 3 CBR and 30 CBR flows.



(a) Number of frames satisfy delay requirement.



(b) Relative energy consumption in different solution compared to AOS.

Fig. 10. Frame delay of CBR traffic and energy consumption evaluation ($D_T = 4$ ms).

almost similar. It is worth to mention here that in this 3 VBR flows case we have found that almost 99.4% frames satisfy delay requirement in ADAEE and FTS-SOOA. Whereas, in FTS-LOOA around 99.25% frames satisfy delay requirement. However, if we notice the energy consumption results for this 3 VBR flows case from Fig. 13b, we can notice that ADAEE is the most energy efficient solution among all these three solutions.

The performance of the three solutions is measured again when there are 30 VBR flows for an ONU. However, in this case we present results of maximum and average delay for a subset of flows (we have selected 8 VBR flows out of 30 flows randomly). As mentioned earlier, ADAEE forces ONU to sleep longer by selecting large T_{min}, so that energy saving can be maximized. As shown in Fig. 12b, d and f, unlike the 3 VBR flows scenario, in all three solutions maximum frame delay never crosses delay requirement boundary in this scenario. It implies that for this 30 VBR flows case in all these three solutions all of the frames satisfy delay requirement (i.e. $D_T = 30$ ms).

Finally, Fig. 13a shows the delay CDF for the three solutions for the arrived frames during those 350 s. In all the cases, less than 0.1% of the frames had a delay that exceeded the operator requirement. Under this relaxed delay

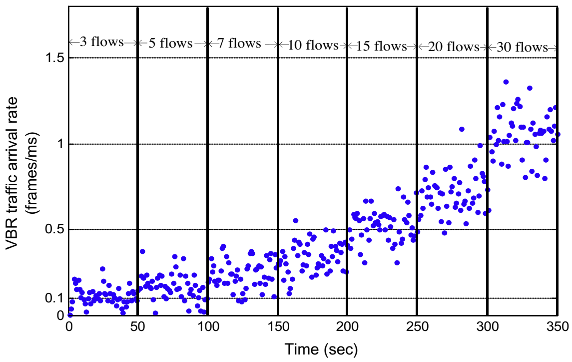


Fig. 11. Arrival rate of VBR traffic during 350 s.

requirement, there is plenty of room to save energy since ONUs can sleep longer times. Fig. 13b presents the energy consumption for the three studied proposals. ADAEE again achieves less energy consumption compared to the other two solutions. In particular, it is interesting to notice that ADAEE always consumes around 20% of the maximum energy, regardless of the number of flows. However, FTS-LOOA and FTS-SOOA show increasing energy consumption for larger number of flows. This is because FTS-SOOA and FTS-LOOA reduce the length of sleep intervals as the number of flows increases, i.e. the opportunity of increasing sleep interval decreases as arrival rate increases. After a certain number of flows (for FTS-LOOA after 15 flows, for FTS-SOOA after 20 flows) there will always be at least some frames to receive from the OLT after waking up from sleep mode. Note that if we consider a given time window, we will find that the smaller the sleep interval, the more energy an ONU ends up consuming for different activities, for example, PON synchronization, OLT clock recovery and listening for downlink arrivals. These phenomena affect FTS-SOOA and FTS-LOOA severely compared to ADAEE, as they always use fixed T_{min}. Whereas, ADAEE selects large T_{min} in this relaxed delay requirement scenario, so that ONUs can sleep longer to save more energy. When there are many flows (15, 20 and 30) ADAEE saves around 70% and 65% more energy, compared to FTS-LOOA and FTS-SOOA respectively.

In summary, ADAEE appears as a very adaptive solution while meeting the operator requirements to save a considerable amount of energy.

4.1.3. Performance evaluation using traffic traces

In this subsection we use the real traces of traffic introduced at the beginning of this section. Then using those traces we evaluate ADAEE, FTS-LOOA and FTS-SOOA under three different operator delay requirements (D_T): two very strict of 4 ms and 10 ms and one relaxed of 30 ms. Our simulation takes into account both downlink and uplink traffic. Whenever an ONU has some uplink frames, it wakes up and uses the assigned mini-slot to send a bandwidth request as considered in [3,6,10,12]. In turn, the OLT assigns the solicited bandwidth for that sleeping ONU in the next uplink transmission cycle (if there is any available bandwidth in the next cycle). Fig. 14 shows the downlink and uplink frames arrival pattern for the 24 h traffic captures

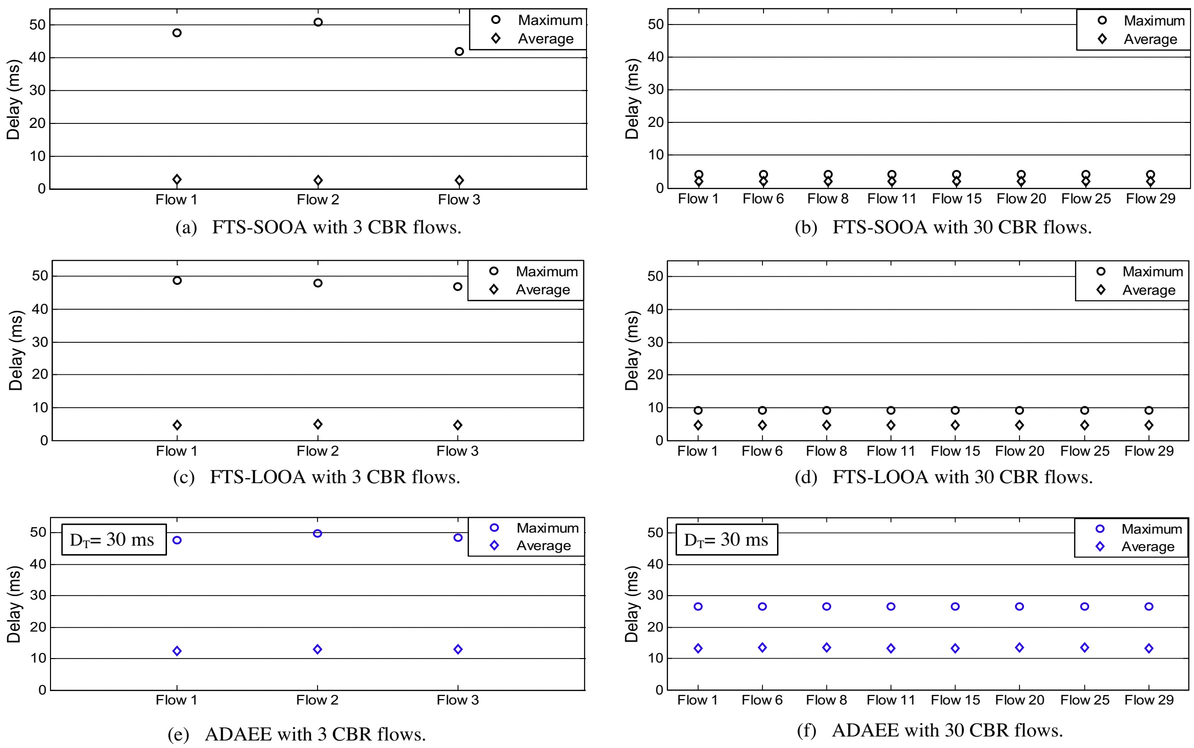
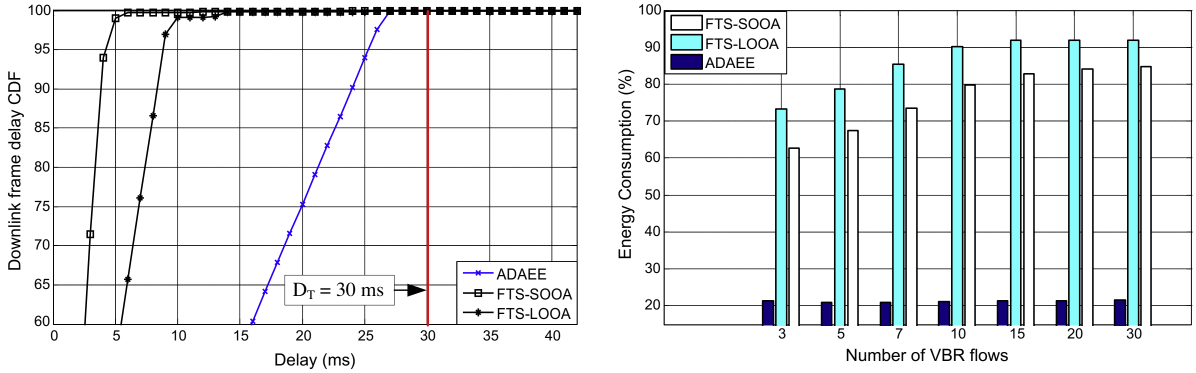


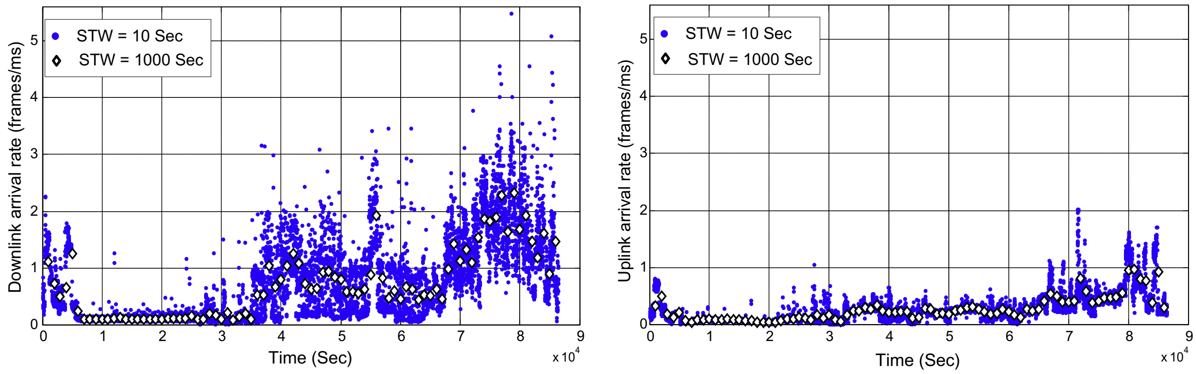
Fig. 12. Performance comparison of different solution when there exist 3 VBR and 30 VBR flows.



(a) Number of frames satisfy delay requirement.

(b) Relative energy consumption in different solution compared to AOS.

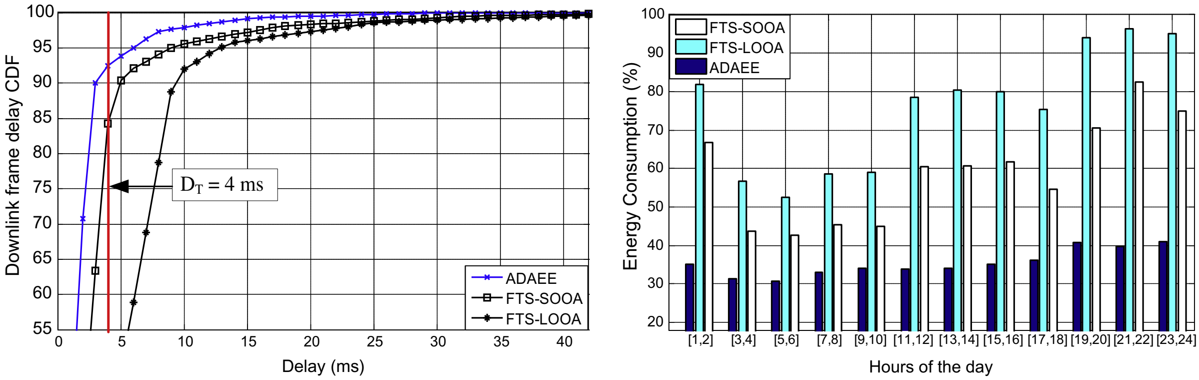
Fig. 13. Frame delay of VBR traffic and energy consumption evaluation ($D_T = 30$ ms).



(a) Downlink arrival rate.

(b) Uplink arrival rate.

Fig. 14. Downlink and uplink frame arrival rate of 16 users under an ONU during 24 h.



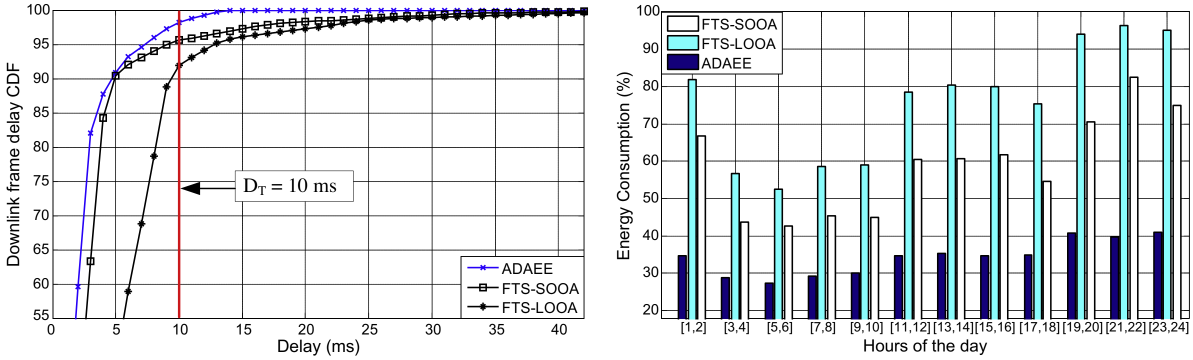
(a) Number of frames satisfy delay requirement.

(b) Relative energy consumption in different solutions compared to AOS.

Fig. 15. Downlink frame delay and energy consumption evaluation using real traffic traces ($D_T = 4$ ms).

using STW (to compute the average arrival rate) equal to 10 and 1000 s.

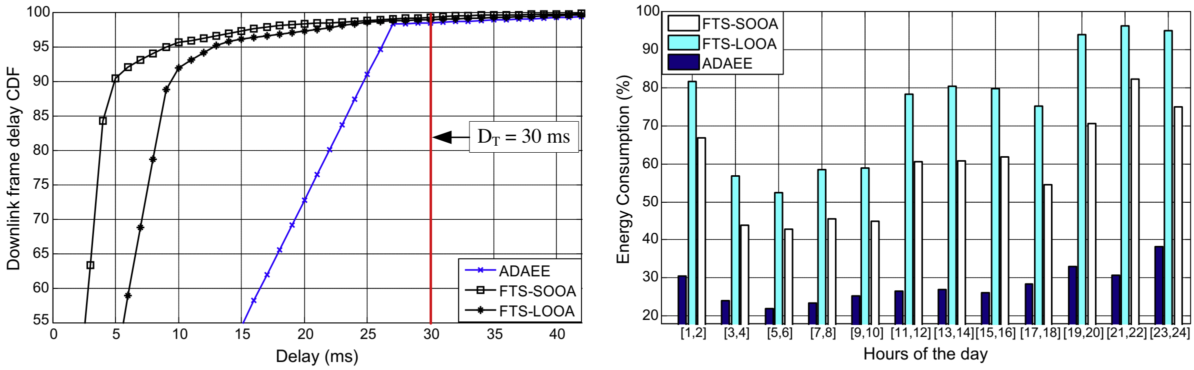
Fig. 15a and b show the frame delay CDF and the energy savings with respect to the AOS (always on ONU case) respectively, for the three compared solutions, when $D_T = 4$ ms. Even for this extremely strict delay requirement, around 92.5% of the frames have a delay below 4 ms in ADAEE. This percentage is still good for FTS-SOOA (85%),



(a) Number of frames satisfy delay requirement.

(b) Relative energy consumption in different solutions compared to AOS.

Fig. 16. Downlink frame delay and energy consumption evaluation using real traffic traces ($D_T = 10$ ms).



(a) Number of frames satisfy delay requirement.

(b) Relative energy consumption in different solutions compared to AOS.

Fig. 17. Downlink frame delay and energy consumption evaluation using real traffic trace ($D_T = 30$ ms).

but very poor for FTS-LOOA (40%). This reinforces the claim that FTS-LOOA cannot be used with strict delay requirements. When analyzing the energy savings, we again find (like for CBR and VBR traffic) that ADAEE is the most efficient solution. In any 2-h time intervals, it always shows an energy consumption below 40% compared to AOS. ADAEE reduces the energy consumption between 2× and 3× compared to the other two solutions.

Fig. 16a and b show the delay CDF and the energy saving for a $D_T = 10$ ms. Although it still represents a strict delay requirement, it is more relaxed than the previous one. The conclusions are the same as those in the previous case. The percentages of frames meeting the delay requirement are 98%, 95% and 92% for ADAEE, FTS-SOOA and FTS-LOOA, respectively. Therefore, ADAEE is still the best one but with a smaller margin. It is interesting to notice that energy saving results are very similar in both cases, with $D_T = 4$ ms and $D_T = 10$ ms. However, during 3–10 h periods, we can see that ADAEE with $D_T = 10$ ms consumes less energy than the $D_T = 4$ ms case. Note that ADAEE sets $T_{min} = T_{min\_Threshold}$, and tunes T_{max} only in strict delay requirement scenarios. ADAEE selects larger T_{max} for $D_T = 10$ ms than for $D_T = 4$ ms. In fact, in high arrival rate scenarios (e.g. $\lambda \geqslant 1$ frames/ms), an ONU's sleep interval never reaches T_{max}. Therefore, under high arrival rate for both cases ($D_T = 4$ ms and $D_T = 10$ ms), an ONU will always have downlink traffic to receive after completing first sleep interval which is equal to size of T_{min} (see Eq. (3)). Therefore, energy consumption in high arrival cases (i.e. 1–2 h, 11–24 h) for both levels of delay requirement, ADAEE will consume the same amount of energy. However, energy consumption results during 3–10 h highlight the significance of ADAEE's effort of tuning T_{max}. In case of $D_T = 10$ ms ADAEE selects larger T_{max} than when $D_T = 4$ ms, therefore when the arrival rate is low (e.g. 3–10 h), an ONU gets chances to have longer sleep for $D_T = 10$ ms case. Consequently, an ONU ends up spending less energy during this low arrival region in $D_T = 10$ ms scenario than $D_T = 4$ ms.

In summary, the obtained results demonstrate that in the case of strict delay requirements, ADAEE appears as the most efficient solution in terms of energy consumption and PON delay performance.

Fig. 17 depicts PON delay performance and energy expenditure for a relaxed operator delay requirement

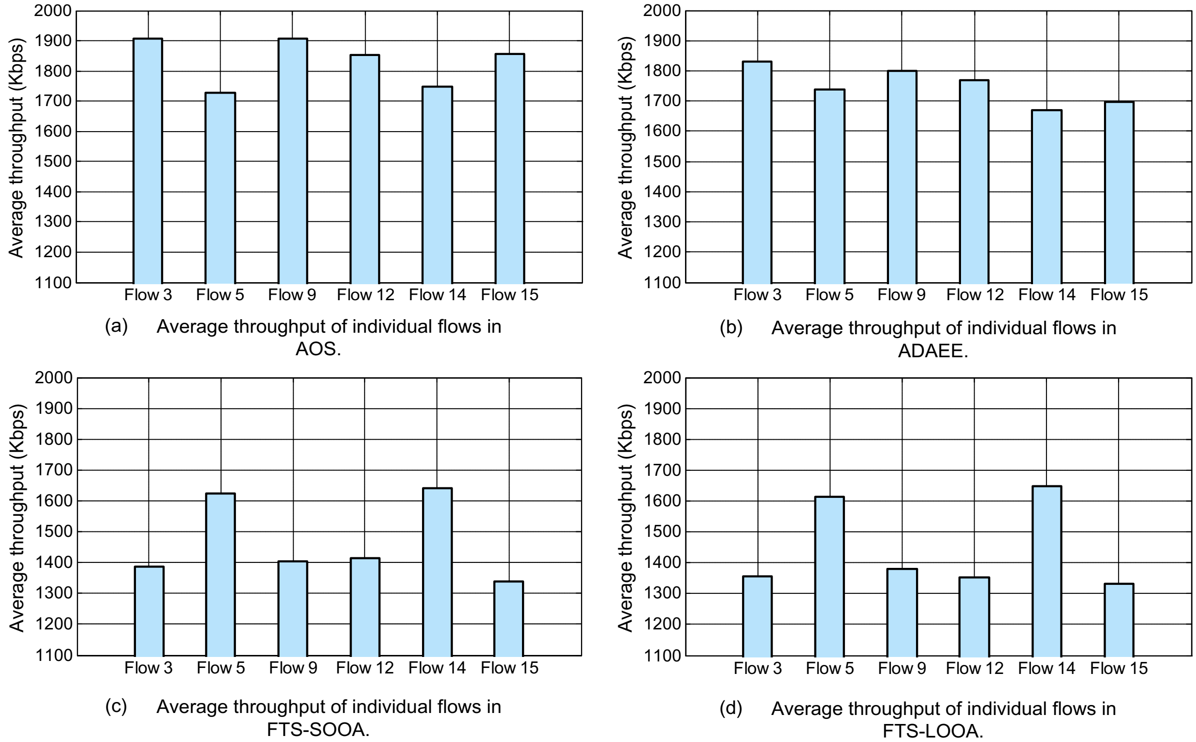


Fig. 18. Relative throughput comparison among different TCP flows in different solutions.

case (i.e. $D_T = 30$ ms). In this case it is possible to save a bigger amount of energy while meeting the delay requirement. Looking at the downlink frame delay CDF for the different solutions in Fig. 17a, we can observe that all the solutions perform well. All solutions meet the delay requirement equally, with less than 0.5% of the frames exceeding that limit. When we compare the energy consumption (see Fig. 17b) over the 24 h for ADAEE, FTS-SOOA, and FTS-LOOA, they reach up to 38%, 82%, and 96% of the AOS energy expenditure, respectively. Global results show that the global energy consumption relative to AOS are: a 27.8% for ADAEE, a 59% for FTS-SOOA and a 75% for FTS-LOOA. Hence, it is clear that ADAEE is an energy efficient solution, because whenever possible it favors energy saving, but still it meets operator delay requirements. Note that in this case ADAEE saves more energy than in the strict delay requirement scenarios.

We have observed that for the given amount of uplink traffic, arrival rate of which is depicted in Fig. 14b, the maximum delay that an uplink frame experiences in ADAEE is around 4 ms when delay requirement is strict (e.g. $D_T = 4$ ms), while the maximum delay reaches up to 10 ms in a relaxed delay requirement scenario (i.e. $D_T = 30$ ms).

Therefore, we can conclude that ADAEE, which is the most efficient solution with the best QoS in cases of strict delay requirements, is also the one presenting higher energy saving in all scenarios. This good performance is reached since ADAEE uses the physical layer (novel ONU architecture) together with the MAC layer (novel algorithm to compute T_{min} and T_{max} shown in Algorithm 1) in order to improve its efficiency.

4.2. Influence of sleep mode on TCP performance

Several researches (e.g. [25,26]) have been conducted in mobile network environment to understand how sleep mode affects TCP performance (e.g. TCP throughput, fairness among competing TCP flows).[7] However, to the best of our knowledge no previous works in the literature addressing energy efficiency in PON networks have studied the effect of the sleep mode on TCP traffic. We believe that this is a very important and challenging issue that must be definitely covered in future research.

Although our paper does not focus on evaluating PON traffic performance, we present a first effort in order to understand how sleep mode affects the throughput of different TCP flows that are sent from different sources to a particular ONU. We define a simulation environment in which the evaluated ONU serves 16 users that are receiving a TCP flow of the same characteristic each one (i.e. for an ONU there are 16 TCP flows). The TCP traffic source uses TCP Reno with maximum window size 64 packets and a packet size of 1460 Bytes. In addition, it is reasonable to assume that there would be UDP background traffic along with TCP flows in the PON network, and all the TCP connections have different propagation delays.

[7] Study performed based on mobile network environment in [26] shows that when the round trip time becomes prominent due to long sleep interval both TCP throughput and fairness might decline.

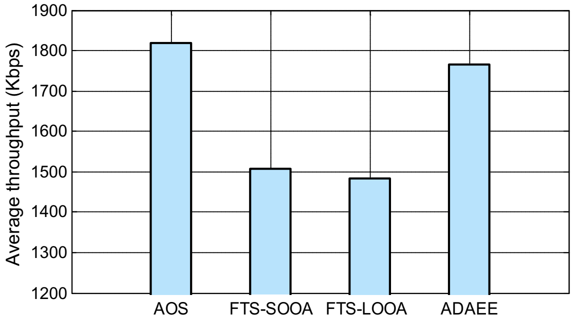


Fig. 19. Influence of sleep mode on 16 TCP flows' average throughput.

It must be noted that in the simulations due to the instantaneous background traffic and the defined sleep intervals we could face some packet losses due to bandwidth saturation and the associated OLT buffer overflow. Thus, it may induce timeout in TCP Reno. Besides, as the length of sleep intervals can significantly influence the delay in frame receiving, and consequently, it may cause prolonging Round Trip Time. All these phenomena might degrade TCP performance.

Fig. 18a–d present average throughout results of a subset of all the competing TCP flows for AOS (the least energy efficient approach), ADAEE, FTS-SOOA and FTS-LOOA, respectively (we have randomly selected results of 6 TCP flows out of 16 flows). It can be observed from the Fig. 18a that in AOS solution average throughput values of all TCP flows are almost similar. In case of ADAEE, average throughput values of all flows are also alike as it is observed from Fig. 18b. However, in this case most of the TCP flows experience slight throughout degradation compared to the throughout results of AOS solution. Whereas, from the results of FTS-SOOA and FTS-LOOA (see Fig. 18c and d) we can notice that average throughput results of all flows are reduced perceptibly compared to AOS. Besides, from Fig. 18c and d noticeable unfairness among the competing TCP flows can be observed for both FTS solutions.

Global results of 16 TCP flows' average throughput for AOS, ADAEE, FTS-SOOA and FTS-LOOA are presented in Fig. 19. As it was expected AOS solution is the one maximizing the average throughput. ADAEE shows a very small reduction of the average throughput of 2.88% compared to AOS solution. This demonstrates that our solution can provide a similar QoS performance than AOS in terms of throughput and delay while reducing a lot the energy consumption. Finally, FTS-SOOA and FTS-LOOA show a TCP throughput degradation of a 16.95% and a 18.38% respectively compared to AOS.

The obtained results present ADAEE as the most suitable solution since it shows a throughput performance very close to AOS, and at the same time appears to be the most efficient solution in terms of energy savings.

5. Conclusions

Energy saving proposals in TDM-PON domain are only meaningful if while dealing with energy savings they also care about QoS performance. Towards this end, ADAEE appears as the first solution that deeply addresses energy saving in PON and at the same time considers QoS restrictions. This paper has analyzed the performance of ADAEE under different types of traffic (i.e. CBR, VBR and real traffic traces) and compared it in front of previous proposals. The obtained results demonstrate that ADAEE is the most energy efficient solution, and in addition, is the one that presents a better QoS performance in terms of delay. It must be noted that some of the proposed solutions in the literature would never be implemented in practice since they sacrifice users QoS (e.g. bad real time traffic quality) in order to reduce the energy consumption.

This paper has also shown that in order to meet both requirements (i.e. QoS and energy efficiency) MAC layer techniques such as sleep mode management must be combined with physical layer improvements (e.g. more energy efficient components, new ONU architectures offering shorter response time).

Moreover, this paper has presented the first effort towards understanding the effects that energy efficient mechanisms based on sleep mode cause to TCP traffic (i.e. the most important in the Internet). We show that while other proposals highly degrade TCP performance, ADAEE presents a performance close to the standard solution in which ONUs remains always active.

Finally, this paper has led us to start a new research line to fully understand the effects of ADAEE (and by extension other PON solutions based in sleep mode) in user level traffic. Among other things we are planning to: (i) study what the impact of ADAEE is on the traffic fairness among users located in different ONUs attached to the same OLT, and (ii) the performance of ADAEE under different OLT scheduling approaches and how it impacts on traffic throughput and fairness. We have demonstrated in this paper that it is feasible to present an energy efficient solution with the main goal of meeting QoS (in terms of delay requirements) imposed by the PON operator.

Acknowledgments

The authors would like to thank the anonymous referees and Md. Arifur Rahman, M.Sc. student at Carleton University, for their insightful criticisms and suggestions, which have helped greatly to improve the quality of the paper. The authors would also like to thank to Rebecca COPELAND for improving overall quality of presentation and language consistency of this paper.

This work was supported by the IT R&D program of MKE/KEIT (10039160, Research on Core Technologies for Self-Management of Energy Consumption in Wired and Wireless Networks) and the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency) (NIPA-2012-H0301-12-1004).

References

[1] M. Gruber, O. Blume, D. Ferling, D. Zeller, M.A. Imran, E.C. Strinati, EARTH — energy aware radio and network technologies, in: IEEE 20th International Symposium on Personal, Indoor and Mobile Radio, Communications, 13–16 September 2009, pp. 1–5.

[2] P. Chowdhury, M. Tornatore, S. Sarkar, B. Mukherjee, Building a green Wireless-Optical Broadband Access Network (WOBAN), Journal of Lightwave Technology 28 (16) (2010) 2219–2229.
[3] Z. Jingjing, N. Ansari, Toward energy-efficient 1G-EPON and 10G-EPON with sleep-aware MAC control and scheduling, IEEE Communications Magazine 49 (2) (2011) s33–s38.
[4] S. Wong, L. Valcarenghi, S-H. Yen, D-R. Campelo, S. Yamashita, L. Kazovsky, Sleep mode for energy saving PONs: advantages and drawbacks, in: IEEE GLOBECOM, Workshop, November 2009, pp. 1–6.
[5] L. Valcarenghi, D.P. Van, P. Castoldi, How to save energy in passive optical networks, Transparent Optical Networks (ICTON), in: 13th International Conference on, June, 2011, pp. 1–5.
[6] R. Kubo, J.-i. Kani, H. Ujikawa, T. Sakamoto, Y. Fujimoto, N. Yoshimoto, H. Hadama, Study and demonstration of sleep and adaptive link rate control mechanisms for energy efficient 10G-EPON, IEEE/OSA Journal of Optical Communications and Networking 2 (9) (2010) 716–729.
[7] S. S.W. Lee, A. Chen, Design and analysis of a novel energy efficient Ethernet passive optical network, in: International Conference on Networking, vol. 0, 2010, pp. 6–9.
[8] J.D. Angelopoulos, H.-C. Leligou, T. Argyriou, S. Zontos, E. Ringoot, T. Van Caenegem, Efficient transport of packets with QoS in an FSAN-aligned GPON, IEEE Communications Magazine 42 (2) (2004) 92–98.
[9] IEEE Std. 802.3ah-2004, Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications, Institute Computer Society, 7 September 2004.
[10] R. Kubo, J. Kani, Y. Fujimoto, N. Yoshimoto, K. Kumozaki, Adaptive power saving mechanism for 10 Gigabit class PON systems, IEICE Transactions on Communications E93-B (2) (2010) 280–288.
[11] D. Ren, H. Li, Y. Ji, Power saving mechanism and performance analysis for 10 Gigabit-class Passive Optical Network systems, in: IEEE International Conference on Network Infrastructure and Digital Content, 24–26 September 2010, pp. 920–924.
[12] ITU-T Recommendation G Suppl. 45, Supplement on GPON, Power Conservation, May 2009.
[13] S. Wong, S-H. Yen, P. Afshar, S. Yamashita, L.G. Kazovsky, Demonstration of energy conserving TDM-PON with sleep mode ONU using fast clock recovery circuit, in: Optical Fiber Communication (OFC), Collocated National Fiber Optic Engineers Conference, 2010, 21–25 March 2010, pp. 1–3.
[14] B. Skubic, D. Hood, Evaluation of ONU power saving modes for gigabit-capable passive optical networks, IEEE Network 25 (2) (2011) 20–24.
[15] Y. Xiao, Energy saving mechanism in the IEEE 802.16e wireless MAN, IEEE Communication Letter 9 (2005) 595–597.
[16] IEEE Std 802.3av-2009, Physical layers Specifications and Management Parameters for 10 Gb/s Passive Optical Networks, October 2009.
[17] ITU-T Recommendation G.983 Series, Broadband Optical Access Systems Based on Passive Optical Networks (PON), January 2005.
[18] N. Suzuki, K. Kobiki, E. Igawa, J. Nakagawa, Dynamic sleep-mode ONU with self-sustained fast-lock CDR IC for burst-mode power saving in 10 G-EPON systems, IEEE Photonics Technology Letters 23 (23) (2011) 1796–1798.
[19] G.C. Sankaran, K.M. Sivalingam, ONU buffer elimination for power savings in passive optical networks, in: IEEE International Conference on Communications (IEEE ICC), 2011, 5–9 June 2011, pp. 1–5.
[20] L.X. Cai, S. Xuemin, J.W. Mark, L. Cai, Y. Xiao, Voice capacity analysis of WLAN with unbalanced traffic, IEEE Transactions on Vehicular Technology 55 (3) (2006) 752–761.
[21] C. Assi, M. Maier, A. Shami, Toward quality of service protection in Ethernet passive optical networks: challenges and solutions, IEEE Network 21 (5) (2007) 12–19.
[22] W. Willinger, M. Taqqu, R. Sherman, D. Wilson, Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level, in: Proc. ACM SIGCOMM, Cambridge, MA, August 1995, pp. 100–113.
[23] D. Chen, S. Garg, M. Kappes, K.S. Trivedi, Supporting VBR VoIP traffic in IEEE 802.11 WLAN in PCF mode", in: Proceedings of OPNETwork2002, Washington DC, August 2002.
[24] A.R. Dhaini, C.M. Assi, M. Maier, A. Shami, Admission Control in Ethernet Passive Optical Networks (EPONs), in: IEEE International Conference on Communication (IEEE ICC), 24–28 June 2007, pp. 2162–2167.
[25] E. Tan, L. Guo, S. Chen, X. Zhang, PSM-throttling: minimizing energy consumption for bulk data communications in WLANs, in: IEEE International Conference on Network Protocols, (ICNP) 2007, 16–19 October 2007, pp. 123–132.
[26] M.N. Alam, R. Jäntti, J. Kneckt, J. Nieminen, Performance study of IEEE 802.11s PSM in FTP-TCP, in: IEEE Vehicular Technology Conference (IEEE VTC Fall), 3–6 September 2012, pp. 1–5.

S.H. Shah Newaz received his B.Sc. degree in Information and Communication Engineering from East West University, Dhaka, Bangladesh, in 2008. In his B.Sc, he received Summa Cum Laude award. He received his M.Sc. in Information and Communication Engineering from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea, in 2010. For pursuing his M.Sc. at KAIST, he received Samsung Electronics Scholarship. He is currently working toward the Ph.D. degree in the Department of Electrical Engineering, KAIST. In 2011, he received best paper award at ICNS. He has written and co-authored more than 20 prestigious international journals and conferences such as IEEE Internet Computing, Elsevier Computer Network, Journal of Networks, ACM SIGCOMM, and IEEE VTC. He also served as a reviewer of Springer Wireless Network journal, IEEE VTC, and ICACT. His research interests include Energy efficient Passive Optical Networks, IP mobility, Wireless Sensor Networks, Quality of Experience, and Internet of Things. He is a student member of IEEE and ACM. Contact him at newaz@kaist.ac.kr (Corresponding author).

Ángel Cuevas received his M.Sc. in Telecomunication Engineering, M.Sc. in Telematics Engineering, and Ph.D. in Telematics Engineering from the Universidad Carlos III de Madrid in 2006, 2007, and 2011 respectively. He is currently a tenure-track Visiting Professor at the Department of Telematic Engineering at Unviersidad Carlos III de Madrid. Prior to that he was Postdoc researcher at Institut Mines-Telecom, Telecom Sudparis from March 2011 until Jan 2013. His research interests focus on On-line Social Networks, P2P Networks, Wireless Sensor Networks and Internet measurements. He is co-autorh of more than 30 papers in prestigirous international journals and conferences such as IEEE/ACM Transactions on Networking, Elsevier Computer Networks, IEEE Network, IEEE Communications Magazine, ACM CONEXT, ACM MSWiM, IEEE ISCC and IEEE ICC. He is co-recipient of the Best Paper Award in ACM MSWiM 2010. Contact him at acrumin@it.uc3m.es.

Gyu Myoung Lee received B.S. degree in electronic and electrical engineering from Hong Ik University, Seoul, Korea, in 1999 and M.S., and Ph.D. degree in school of engineering from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2000 and 2007, respectively. He is currently with the Institut Mines-Telecom, Telecom SudParis and the KAIST as an adjunct associate professor. He was an invited researching staff at the Electronics & Telecommunications Research Institute (ETRI), Korea for international standardization and also worked as a guest researcher at the National Institute of Standards and Technology (NIST), USA. His research interests include Internet of things, Web of things, cloud-based multimedia services and energy saving networks. Prof. Lee has actively participated in standardization meetings including ITU-T SG 13 (Future Networks) as a Rapporteur and IETF. He has contributed more than 250 proposals for standards and published more than 80 papers in academic journals and conferences. He is an IEEE senior member. Contact him at gm.lee@it-sudparis.eu (Corresponding author).

Noël Crespi holds a "diplôme d'ingénieur" from ENST, a Ph.D. and habilitation from Paris VI University. He worked from 1993 in CLIP, Bouygues Telecom and joined France Telecom R&D in 1995 where he led the Mobicarte prepaid service project and took an active role in various standardisation committees. In 1999, he joined Nortel Networks as Telephony Program manager. He joined Institut Telecom in 2002 and is currently professor and Program Director, leading the Network and Service Architecture lab. He is appointed as coordinator for the standardisation activities in ETSI and 3GPP. He is also a Visiting Professor at Asian Institute of Technology. Contact him at noel.crespi@it-sudparis.eu.

Jun Kyun Choi received the B.Sc. (Eng.) from Seoul National University in electronics engineering, Seoul, Korea in 1982, and M.Sc. (Eng.) and Ph.D. degree in 1985 and 1988, respectively, in electronics engineering from Korea Advanced Institute of Science and Technology (KAIST). From June 1986 until December 1997, he was with the Electronics and Telecommunication Research Institute (ETRI). In January 1998, he joined the Information and Communications University (ICU), Daejeon, Korea as a Professor. He is a Senior Member of IEEE, the executive member of The Institute of Electronics Engineers of Korea (IEEK), Editor Board of Member of Korea Information Processing Society (KIPS), Life member of Korea Institute of Communication Science (KICS). His research interests are concerned with broadband network architecture and technologies particular emphasis on performance and protocol problems. Secondary interests include the international standard activities on broadband network architecture and protocol with mobility support. He is an active member of ITU-T SG 13 as a Rapporteur or Editor from January 1993 on the ATM, MPLS, and NGN issues. He had also submitted more than 30 drafts on IETF during last few years. Contact him at jkchoi59@kaist.edu.