Lee, GM

Early Wake-Up Decision Algorithm for ONUs in TDM-PONs With Sleep Mode

http://researchonline.ljmu.ac.uk/id/eprint/3287/

Article

Citation (please note it is advisable to refer to the publisher's version if you intend to cite from this work)

Lee, GM (2016) Early Wake-Up Decision Algorithm for ONUs in TDM-PONs With Sleep Mode. Journal of Optical Communications and Networking, 8 (5). pp. 308-319. ISSN 1943-0620

Early Wake-up Decision Algorithm for ONUs in TDM-PONs with Sleep Mode

Alaelddin Fuad Yousif Mohammed, *Student Member, IEEE,* S.H. Shah Newaz, *Member, IEEE,*
Mohammad Rakib Uddin, *Member, IEEE,* Gyu Myoung Lee, *Senior Member, IEEE,*
and Jun Kyun Choi, *Senior Member, IEEE*

***Abstract*—Recent IEEE and ITU-T standards for TDM-Passive Optical Network (TDM-PON) with sleep mode recommend that the Optical Line Terminal (OLT) in a TDM-PON should be in charge of invoking an Optical Network Units (ONU) to move into sleep state in absence of frames. It is considered that, on upstream frame arrival, a sleeping ONU can leave sleep state, in which an ONU turns off its transmitter or both transmitter and receiver immediately, prior to its assigned sleep interval length. In this paper, we refer to this approach as immediate early wake-up (IMEW). According to the standards, the OLT may or may not allow an ONU to trigger early wake-up function (EWF) on upstream frames arrival. If the OLT does not allow EWF (not support early wake-up (NSEW)), an ONU should stay in sleep state during its assigned sleep duration and buffer all the upstream frames while it is in this state. In IMEW, upstream frames experience small delay but ONU's energy consumption increases remarkably. Conversely, in NSEW, an ONU consumes less energy compared to IMEW at the price of increasing upstream frame delay and possibility of its buffer overflow. In this paper, the limitations of IMEW and NSEW have motivated us to propose a novel Early Wake-up Decision (EWuD) algorithm which aims at meeting upstream frame delay requirement while reducing ONUs' energy consumption as much as possible. The role of EWuD algorithm is to select an appropriate time for triggering EWF taking into consideration two factors: (1) buffer overflow probability and (2) delay requirement violation of upstream frames. We evaluate EWuD performances using our TDM-PON OPNET Modular based simulation model under wide range of scenarios. Findings demonstrate that our proposed EWuD can successfully meet delay requirement of all upstream frames while reducing ONU's energy consumption significantly.**

***Index Terms*—TDM-PON, Sleep mode, Early wake-up.**

I. INTRODUCTION

ENERGY saving is an important aspect for developing future information and communication technology equipment. TDM-Passive Optical Network (TDM-PON) (e.g. Ethernet PON (EPON), Gigabit-capable PON (GPON)) is a successful access network technology which provides high speed access to end users and consumes comparatively less energy than other access technologies (e.g. Worldwide Interoperability for Microwave Access (WiMAX) [1]). This has motivated many network operators to deploy TDM-PONs widely in many countries throughout the world [2].

A TDM-PON is equipped with an Optical Line terminal (OLT) which is located in the Central Office (CO) of service provider and plays as a master equipment that controls connected Optical Network Units (ONUs) (placed at the customer side) through a passive splitter. In a TDM-PON, the bandwidth of fiber link is shared within all connected ONUs. Consequently, the downstream frames (from OLT to ONUs) follows broadcast manner in which all the ONUs receive broadcast frames from the OLT and filter-out their belonging frames (e.g. using Logical Link Identifier in EPON). Whereas, the upstream frames (from the ONU to the OLT) are forwarded by ONUs during a dedicated slot assigned by the OLT (the OLT uses Dynamic Bandwidth Allocation algorithm (DBA) to decide each ONU's dedicate upstream transmission slot).

Stupendous increment of energy demand of access network equipment (e.g. ONU) has triggered many research initiatives to maximize energy efficiency in network equipment. In this area, we can already find several research efforts from industries and academia in maximizing energy saving performance in TDM-PONs. For instance, Service Interoperability in EPONs (SIEPON) IEEE 1904.1 [3] standardized two power-saving modes: (1) TRx sleep mode (transmitter and receiver sleep mode), and (2) Tx sleep mode (transmitter sleep mode). Note that TRx sleep and Tx sleep modes are equivalent to cyclic sleep mode and doze mode, respectively, defined in ITU-T Recommendation G.sup 45 [4]. Thus, an ONU with a power-saving mode is flipping between active state and sleep state. In active state, an ONU is fully functional (all components are on). In the sleep state of TRx sleep mode, an ONU powers down its TRx unit and some other power hungry components. Whereas, in the sleep state of Tx sleep mode, an ONU switches off its Tx unit, while keeping the Rx unit fully utilitarian [3].

TDM-PON standards with power saving modes in ONUs (e.g. [3]–[5]) and several other energy efficient research proposals (e.g. [6], [7]) considered that the OLT should be in charge of deciding sleep interval lengths of an ONU. According to many researchers, the criteria for the OLT to decide whether an ONU should move into sleep state or not relies on presence or absence of downstream frames only (e.g. [6], [7]). In case of upstream, it is considered in [3]–[7] that an ONU should leave sleep state immediately on arrival of any upstream frame from customer premises. Leaving sleep state prior to the OLT's defined sleep period is termed as

---

F. Y. M Alaelddin, S. H. S. Newaz and J. K. Choi are with Korea Advanced Institute of Science and Technology (KAIST), Daejeon, South Korea (e-mail: alaelddin@kaist.ac.kr; newaz@kaist.ac.kr; jkchoi59@kaist.edu).

M. R. Uddin is with the Dept. of Electrical and Electronic Eng., Faculty of Engineering, Universiti Teknologi Brunei, Brunei (e-mail:rakib.uddin@itb.edu.bn).

G. M. Lee is with Liverpool John Moores University (LJMU), Liverpool, United Kingdom (e-mail:G.M.Lee@ljmu.ac.uk).

S.H. Shah Newaz is the corresponding author.

Early Wake-up Function (EWF) in SIEPON IEEE 1904.1 [3] and ITU-T G.988 [5]. In fact, the conditions for triggering EWF of ONUs are not part of the scope of the SIEPON IEEE 1904.1 and ITU-T G.988. Likewise, proposals in [6], [7] do not define ONU's EWF triggering condition. We refer to triggering EWF immediately on arrival of upstream frames as Immediate EWF (IMEW). Note that IMEW can lead to reduce upstream frame delay noticeably [8], [9]. The most important limitation of IMEW lies in the fact that ONUs in this approach will end up spending significant amount of energy due to frequent sleep to active state transition[1] under high frame arrival rate scenarios. Furthermore, when an ONU uses IMEW, number of control messages associated with sleep mode could increase in a TDM-PON due to frequent wake-up, thereby, reducing overall network energy efficiency and minimizing bandwidth utilization [10]. Authors in [11] show how sleep mode associated control messages could contribute in increasing ONUs' energy consumption.

In this paper, we argue that an ONU needs to take into consideration the delay requirement of arrived upstream frames and its buffer occupancy in order to make decision whether it should execute EWF (leave sleep state) or not, instead of leaving sleeping state immediately on upstream frame arrival. The reason behind is that upstream frames can have different delay requirements. Then, on arrival of upstream relaxed delay requirement frames (e.g. HTTP traffic) at an ONU (assuming that there is no strict delay requirement frames in ONU's buffer and the ONU's buffer is not totally occupied), the ONU can defer its EWF execution, instead of leaving sleep state immediately. By doing so, the ONU could extend its sleep period, thereby reducing number of sleep to active state transitions and sleep mode related signaling with the OLT. And consequently, the ONU will end up consuming less energy than the ONU that follows IMEW.

SIEPON IEEE 1904.1 standard identifies that the OLT can disable or enable EWF of an ONU. In this paper, we refer to the case where an ONU is not allowed to trigger EWF on upstream frame arrival as Not Support EWF (NSEW). Thus, an ONU under NSEW is forced to stay in sleep state during OLT's defined sleep period. If the sleep period is very long, upstream frames can experience unrestrained delay (specially high priority Class of Service (CoS) frames), and subsequently, some frames can be dropped due to ONU's finite buffer size. However, under NSEW, the ONUs having long sleep periods can save significant amount of energy, because they can complete OLT's defined sleep period without any interruption from EWF. Figure 1 presents the operation of NSEW and IMEW including control messages exchange between the OLT and an ONU.

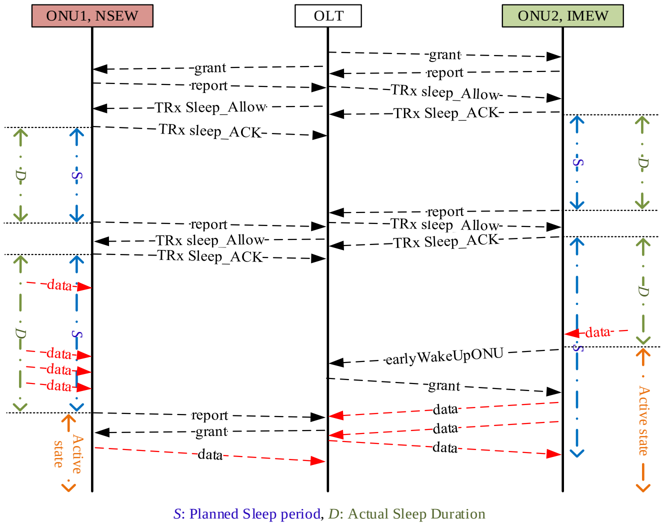


Fig. 1: Comparison between IMEW and NSEW.

Both of the aforementioned approaches (IMEW and NSEW) have strengths and weaknesses. Therefore, it is very critical to propose a solution which not only reduces ONUs' energy consumption by allowing them to complete OLT's defined sleep period whenever possible, but also meets upstream frames delay requirement. In this paper, we propose a novel algorithm to reduce energy consumption of ONUs in a TDM-PON with sleep mode. In particular, our algorithm follows a procedure to select an appropriate time to trigger EWF after taking into consideration delay requirements of upstream frames and an ONU's buffer overflow probability. To the best of our knowledge, no former research has defined rules to trigger EWF of ONUs in TDM-PONs with sleep mode. Performance results obtained by means of our state-of-the-art OPNET Modular based TDM-PON simulator [8], [12] show that our proposal can reduce ONU's energy consumption significantly, while meeting upstream frames delay requirement.

This paper is organized as follows. Section II reviews related work. In Section III, we present the system model, assumptions and algorithm. Section IV shows performance evaluation of our proposal. Section V presents discussion associated with our algorithm. In Section VI, we conclude our findings.

II. RELATED WORK

TDM-PON uses DBA algorithm to grant upstream transmission duration to all connected ONUs. Therefore, numerous DBA algorithms have been developed to cope with higher bandwidth utilization and Quality of Service (QoS) requirement. Authors in [13] noted that Interleaved Polling with Adaptive Cycle Time (IPACT) [14] is one of the most robust bandwidth utilizing DBA algorithms. In IPACT, ONUs send *REPORT* control messages to monitor their buffer status. The OLT collects ONUs' *REPORT* control messages and uses IPACT DBA algorithm to decide the upstream grant time for each ONU. However, having an ONU with high bandwidth request can lead to bandwidth saturation and that ONU can sew up the upstream channel. Therefore, authors in [14] set an upper-bound of bandwidth request.

TDM-PONs utilizing OLT-driven power-saving technique facilitate existing master-slave OLT-ONU(s) architecture including control messages provided by Multipoint Control Protocol (MPCP). For instance, SIEPON IEEE 1904.1 and ITU-T G.988 standards introduce bunch of energy-saving control messages (e.g. SLEEP_ALLOW control message in SIEPON

---

[1] The time required to gain synchronization and OLT's clock recovery when an ONU enters into active state from sleep state. In this paper, we refer this amount of time as overhead time (O_T).

[3]). In this manner, the OLT is in charge of deciding an ONU's eligibility to let in power-saving cycle. Each power-saving cycle consists of one sleep period (T_s) during which an ONU stays in sleep state, and one active period (T_{listen}) in which an ONU can listen to the OLT's further instruction and inform its current buffer situation to the OLT.

An ONU in a TDM-PON with TRx sleep mode requires O_T amount of time (overhead time) to shift from sleep state to active state. For instance, an ONU with conventional architecture requires significant amount of O_T due to synchronization and clock recovery [15]. In this Long Overhead ONU Architecture (LOOA), O_T is between 2.125 ms to 5.125 ms [15], and the ONU consumes around 0.7 W during sleep state [4], [15]. As a solution to decrease O_T, authors in [15] proposed Short Overhead ONU Architecture (SOOA) in which O_T= 125 μs. However, the small O_T value in SOOA came at the price of consuming 1.28 W during sleep state (i.e. consuming 83% more energy than that of LOOA during sleep state).

SIEPON IEEE 1904.1 [3] identifies that PON equipment should support EWF. Moreover, it recommends that the OLT is responsible for disabling or enabling EWF of ONUs. However, conditions for triggering EWF are not in the scope of the standards. In this paper, we propose a novel algorithm that can choose a proper time to execute EWF. Our solution can significantly reduce ONUs' energy consumption while meeting the delay requirements of upstream frames.

III. SYSTEM MODEL

Similar to [3], [6], [7], we assume that in each upstream DBA cycle (T_C) there are gratuitous upstream grants T_E for ONUs in sleep state (of TRx sleep mode) in order to send their *earlyWakeupONU* control message to the OLT for requesting upstream grant (upstream transmission slot) on arrival of upstream frames. We use the notation T_e^i to define the gratuitous early wake-up grant during DBA cycle 'i' of the current sleep cycle for a particular ONU (see Fig. 2(a)), where T_E= T_e^1, T_e^2, ..., T_e^i. We assume that the OLT invokes the ONU to enter into TRx sleep mode based on the absence of upstream and downstream frames. Furthermore, in this paper, we consider that the OLT uses IPACT DBA algorithm to quantify upstream transmission slots for ONUs. The following subsections provide detailed explanations of the algorithm that we propose in this paper.

A. Buffer Overflow Probability

With the objective to develop a probability of finite buffer overflow for an ONU, we assume, similar to [2], [7], [11], [16], that the upstream frame arrival rate (λ) follows Poisson distribution. Hence, the probability that x frames arrive during sleep period T_s is expressed as follows [16]:

$$P(x\ arrival\ during\ T_s) = \frac{(\lambda T_s)^x e^{-\lambda T_s}}{x!}. \tag{1}$$

Suppose that the ONU has finite buffer size denoted by B_{size}, and B_o defines its current buffer occupancy. We assume here that the ONU can accommodate at most B_o+1 upstream frames (B_o frames are in the ONU's buffer and 1 frame is in

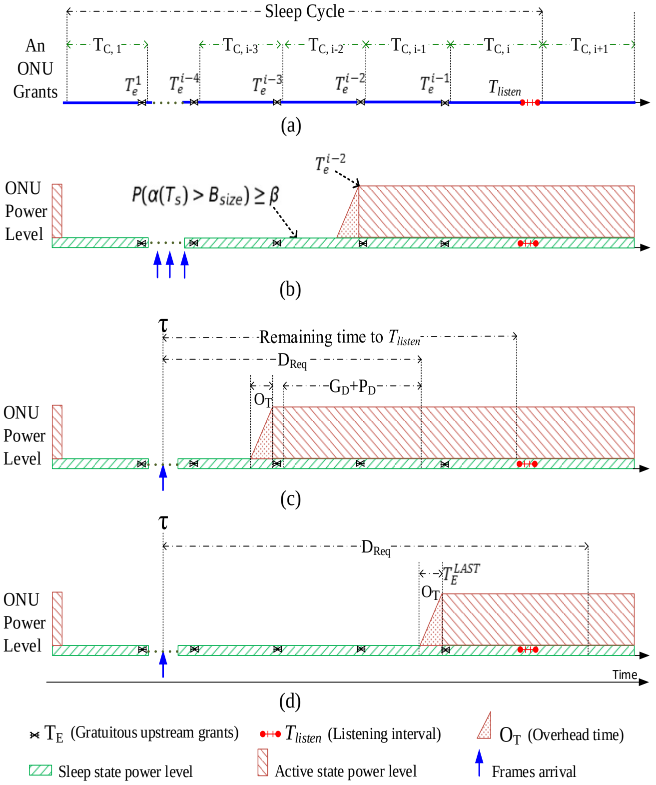


Fig. 2: (a) ONU grants during one sleep cycle. (b) ONU sleep behavior under EWuD having heavy frame arrival. (c) ONU sleep behavior under EWuD when D_{Req} violation point is less than T_{listen}. (d) ONU sleep behavior under EWuD when D_{Req} violation point is larger than T_{listen} by less than one T_C.

service). In order to experience buffer overflow at the ONU during sleep period T_s, the following expression should hold [16]:

$$x \geq B_{size} - B_o + 1 = B_{size} - (B_o - 1). \tag{2}$$

The buffer overflow probability can be stated as the probability of buffer occupancy (after T_s) exceeding the buffer size of an ONU. Therefore, the probability of buffer overflow is expressed as follows:

$$\begin{aligned} P(\alpha(T_s) > B_{size}) &= P(x \geq B_{size} - (B_o - 1)) \\ &= \sum_{x=B_{size}-(B_o-1)}^{\infty} \frac{(\lambda T_s)^x e^{-\lambda T_s}}{x!}, \end{aligned} \tag{3}$$

where, $\alpha(T_s)$ is the expected number of arrived upstream frames after T_s. Suppose that the ONU sets upper-bound of buffer overflow probability (β). Then, we can express the buffer overflow probability as in Eq. (4).

$$P(\alpha(T_s) > B_{size}) \geq \beta. \tag{4}$$

An illustration of this situation is delineated in Fig. 2(b). As expressed in Fig. 2(b), the execution of EWF should take place at the next available T_E (T_E^{Next}) when buffer overflow probability overpasses β. At this point, it is worth mentioning that how to get β value is out of the scope of this paper (many research efforts have been devoted in finding a value of β (e.g. [17]–[20])).

B. Delay Violation

In this paper, similar to [6], we define delay requirement (D_{Req}) as is the maximum allowed forwarding delay between an ONU and the OLT. Our objective is to minimize ONUs' energy consumption by reducing number of interruptions during sleep state due to upstream frames arrival, while meeting the delay requirement of upstream frames. Therefore, our algorithm should select an appropriate T_E taking into consideration the delay requirement of arrived upstream frames, so that it can secure the departure of upstream frames from an ONU before violating their delay requirements.

Suppose that at time τ (see Fig. 2(c)) in sleep period T_s a frame arrives with delay requirement D_{Req}. We can observe if the D_{Req} can be violated based on the following expression:

$$(T_{listen} - \tau) - (O_T + G_D + P_D) < D_{Req}, \tag{5}$$

where, G_D and P_D present upstream grant allocation delay, and OLT to ONU propagation delay, respectively. If the condition in Eq. (5) becomes true, the ONU needs to leave sleep state prior to T_{listen}. In such case, the ONU needs to choose a proper T_E for sending *earlyWakeupONU* control message to inform the OLT of its transition from sleep state to active state and claim upstream transmission slot. The decision criteria for choosing the value of T_E relies mainly on O_T, G_D and P_D.

To understand the factors that influence G_D, let us consider that the point of delay violation ($\tau + D_{Req}$) of an upstream frame is located somewhere in a T_C as presented in Fig. 2(c). Note that, after an ONU wakes up due to upstream frame arrival and claim for an upstream grant, it can get the upstream grant in the next T_C [6]. Therefore, the ONU requires at most one DBA cycle to send its frames after leaving sleep state. However, it could happen that the OLT's allocated slot for an ONU in next DBA cycle resides after $\tau + D_{Req}$, then the ONU's upstream frames will end up violating D_{Req}. Taking this into account, we consider that an ONU should select T_E at most two DBA cycles earlier than the delay violation point. Furthermore, it is worth noting that, if the upstream frames volume is high, it could happen that the allocated upstream grant for an ONU making bandwidth request (using T_E during $T_{C,i}$) resides after one or more DBA cycle later (i.e. $T_{C,i+2}$ or $T_{C,i+3}$). To avoid this, we consider that the OLT should allocate an upstream slot in $T_{C,i+1}$ to the ONU that leaves sleep state using EWF during $T_{C,i}$.

Similarly, when the delay requirement violation point of an upstream frame in an ONU falls within the following T_C after the T_C where T_{listen} resides, then, D_{Req} of the frames could be violated if the ONU makes bandwidth request during the T_{listen}. We refer to the end point of the following T_C as $T_{C,f}^{end}$. In this particular case (see Eq. (6)), an ONU should wake-up during the last T_E (T_E^{LAST}) of the sleep interval in order to avoid delay requirement violation of the ONU's upstream frames. An example of this scenario is depicted in Fig. 2(d).

$$T_{listen} < \tau + D_{Req} \leq T_{C,f}^{end}. \tag{6}$$

Algorithm 1: Early Wake-up Decision (EWuD) Algorithm

```
Data: $B_{size}, B_o, \lambda, D_{Req}, \beta$
Result: Early Wake-up Decision
{Find $T_E^{Next}$}
$T_E^{Calc}$ /* $T_E^{Calc}$: calculated $T_E$ */
while $T_E^{Next} < T_{listen}$ do
    $D\_V$ = $false$            /* delay violated */
    $Overflow$ = $false$        /* buffer overflow */
    /* Delay requirement violation */
    if $D_{Req} < (T_{listen} - \tau) - (O_T + G_D + T_P)$ then
        $D\_V$ = $true$         /* Delay Violated */
        {Find current DBA cycle (n) and the total number of DBA cycles (k) within the $T_s$}
        for $i = n$ to $k - 1$ do
            if $D_{Req} > (T_e^n - \tau) - (O_T + G_D + T_P)$ then
                $i \leftarrow n - 1$
                break
            $T_E^{Calc} \leftarrow T_e^i$
        end
    else if $(T_{listen} < \tau + D_{Req} \leq T_{C,f}^{end})$ then
        $D\_V$ = $true$
        $T_E^{Calc} \leftarrow T_E^{LAST}$
    /* buffer overflow probability */
    if $P(\alpha(T_s) > B_{size}) \geq \beta$ then
        $Overflow$ = $true$
    /* making decision */
    if $(D\_V = true)$ and $(Overflow = true)$ then
        {Wake-up in $T_E^{Next}$}
    else if $(D\_V = true)$ and $(Overflow = false)$ then
        {Wake-up in $T_E^{Calc}$}
    else if $(D\_V = false)$ and $(Overflow = true)$ then
        {Wake-up in $T_E^{Next}$}
    else
        {Wake-up in $T_{listen}$}
end
```

C. Early Wake-up Decision Algorithm

Based on the two criteria explained in Sections III-A and III-B, we propose our novel Early Wake-up Decision (EWuD) algorithm for ONUs in a TDM-PON with TRx sleep mode. On the arrival of upstream frames, the EWuD algorithm checks the ONU's current buffer occupancy (B_o), λ and D_{Req}, which can be obtained from the CoS of arrived frames, as noted in [21]. The ONU finds next available T_E (T_E^{Next}) from current time T where $T + O_T \leq T_E^{Next}$. And then, the EWuD verifies first if upstream frames' D_{Req} can be violated or not using Eq. (5) and Eq. (6). Next, the EWuD algorithm measures the possibility of buffer overflow using Eq. (4). Based on these collected information, this algorithm takes into account the following four cases to come up with the final decision:

(a) The first case refers to the scenario in which there exists high CoS upstream frames (strict D_{Req}) in the ONU buffer and ONU buffer overflow is likely to occur (measured based on Eq. (4)). In this case, the EWuD algorithm uses the next available T_E for ONU's EWF execution since the ONU's buffer overflow probability overpasses β. The ONU sends *earlyWakeupONU* control message in the T_E^{Next} after leaving sleep state. Therefore, in this case, mostly, upstream frames are forwarded to the OLT far earlier than their delay violation points.

(b) If EWuD algorithm notices that delay requirement of ONU's upstream frames could be violated (based on Eq. (5) and Eq. (6)) but there is no possibility of buffer overflow, it finds proper T_E in order to execute EWF (the procedure is explained in the algorithm).

(c) The ONU only has low priority CoS frames (relaxed D_{Req}); however, it is likely that buffer overflow could occur. In this case, the EWuD follows the same procedures as in the case (a).

(d) When the conditions presented in Eq. (4), (5) and (6) do not hold, the proposed EWuD algorithm let the ONU stay in sleep state up to T_{listen}. The ONU wakes up during T_{listen} to get OLT's further instruction and forward its upstream frames if the buffer is not empty.

IV. PERFORMANCE EVALUATION

In this section, we compare performance results of EWuD in front of IMEW and NSEW. We evaluate their performance results using our outstanding TDM-PON OPNET Modular based simulator model (this model presented in [12] and used in [8], [22]). For performance evaluation, we assume that the OLT assigns always a fixed sleep interval period T_s for an ONU similar to [11]. We suppose that T_s= 50 ms[2].

Similar to [14], [24], we assume that there are three different CoSs of upstream frames having different delay requirements (D_{Req}) in a TDM-PON. Consequently, we consider that an ONU maintains a separate queue inside its buffer for each of the CoSs. These CoSs are:

- *Best Effort (BE):* Since frames with this class are not delay sensitive, we consider them as the lowest priority. Further, this class is served only when the higher priority class queues are empty. Moreover, as all the queues share the same buffer, it is possible that higher priority frames can supersede BE frames when higher priority class queues are full.
- *Assured Forwarding (AF):* We suppose that AF frames have higher priority than that of BE frames. In this paper, we assume that D_{Req} of AF frames is 25 ms (i.e. maximum delay to attain an acceptable quality for voice over Internet Protocol in a local area network [25]).
- *Granted Forwarding (GF):* We consider that GF frames have D_{Req}= 10 ms (i.e. the delay requirement for some delay sensitive Smart Grid applications [26]). This class has the highest priority among all and can supersede AF and BE frames in case when its queue is full.

We evaluate the performance of our proposed algorithm under two different ONU architectures: LOOA and SOOA (we briefly introduced these architectures in Section II). Table I presents the simulation parameters.

TABLE I: Simulation Parameters.

Description	Value
Sleep period (T_s)	50 ms
Power consumption in active state	4.69 W [4]
Power consumption in sleep state under LOOA	0.7 W [4]
Power consumption in sleep state under SOOA	1.28 W [15]
O_T under LOOA	2.125 ms [15]
O_T under SOOA	125 μs [15]
Upstream DBA cycle length (T_C)	1 ms
Number of ONUs	16
ONU buffer size	100 KBytes [27]
OLT-ONUs link speed	1 Gbps
Propagation time between the OLT and ONU	100 μs [13]

A. Performance under BE Frames

In this subsection, under BE frames, we evaluate the average sleep duration and energy consumption of an ONU for all three solutions (EWuD, IMEW and NSEW). To evaluate proposed EWuD, we are interested to observe the performances under wide range of β and λ. In our performance evaluation, β and λ range between 0.05 to 0.8 frame/ms and 0.1 to 3 frame/ms, respectively.

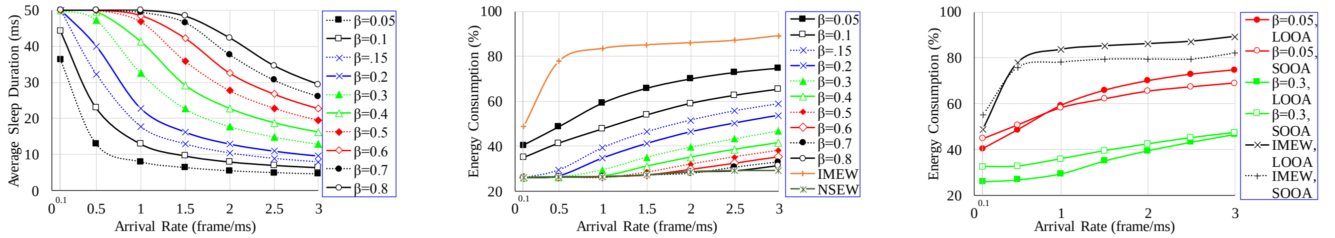
Figure 3(a) demonstrates ONU's average sleep duration under LOOA. We can notice from this figure that when β= 0.05 and λ= 0.1 frame/ms the average sleep duration is around 35 ms. However, with the increment of λ, ONU's sleep duration rapidly decreases. We can observe similar phenomenon when β= 0.1. Note that a very small β value (e.g. 0.05, 0.1) enforces the proposed EWuD to trigger EWF very frequently as the frame arrival increases (i.e. case(c) of EWuD as explained in subsection III-C), thus resulting in reducing the average sleep duration. However, we can clearly observe from Fig. 3(a) that when $\beta \geq 0.15$, ONU's average sleep duration could reach up to 50 ms over the low frame arrival region (e.g. $\lambda \leq 1.5$ frame/ms). This indicates that proposed EWuD lets an ONU stay in sleep state during its assigned sleep period (50 ms) without any interruption under low λ regions when β value is relatively high. That is the case(d) of EWuD as explained in subsection III-C.

Results presented in Fig. 3(a) lead us to conclude that in EWuD the β is a very performance influential parameter. β value can directly affect average sleep duration of an ONU, and consequently, it can affect ONU's energy consumption performance.

Energy consumption performances of three solutions (EWuD, IMEW and NSEW) are presented in Fig. 3(b). Similar to [6], energy consumption results of each solution are presented as percentage of energy consumption of an ONU that does not apply energy saving techniques (e.g. an ONU in IEEE 802.3ah standard). We refer to this solution as Always Active Solution (AAS). This figure indicates that IMEW is the one consuming the maximum amount of energy among all these solutions. This is because, in IMEW, an ONU is forced to leave sleep state whenever an upstream frame arrives. During high frame arrival the frequency of executing EWF increases, thus an ONU ends up spending significant amount of time for sleep to active state transition in IMEW. This results in increasing energy consumption in IMEW stupendously as illustrated in Fig. 3(b). Figure 3(b) also depicts that in case of NSEW,

---

[2] The maximum time between two *Report* control messages sent from an ONU to the OLT is defined in early EPON standards (IEEE Std 802.3ah [23]) as 50 ms by default. Therefore, the maximum sleep period for ONUs assigned in many research is 50 ms (e.g. [6]).



(a) Average sleep duration under LOOA.

(b) Energy performance under LOOA.

(c) LOOA and SOOA energy rendering comparison.

Fig. 3: Average sleep duration and relative energy consumption compared to AAS under BE frames and different β values.

an ONU consumes the lowest amount of energy among all three solutions. Note an ONU using NSEW can prolong its sleep duration as it does not have EWF (ONU can stay in sleep state during OLT's assigned sleep duration without any interruption). Consequently, an ONU using NSEW can save significant amount of energy compared to EWuD and IMEW. The major limitation of NSEW lies in the fact that it could lead to increase upstream frame delay and loss during high frame arrival regions. It is because an ONU using NSEW buffers all incoming upstream frames when it stays in sleep state.

Figure 3(b) also demonstrates energy consumption performance when an ONU uses proposed EWuD. Results of EWuD reveal that the higher the β value the less the energy consumption that an ONU can have, and vice versa (β= 0.05 presents the maximum energy consumption, whereas β= 0.8 presents the minimum energy consumption). Then, based on results presented in Fig. 3(a) and 3(b), we can also realize that there exists a strong relationship between average sleep duration and energy consumption performance of an ONU.

Next, we compare energy consumption performance of IMEW and EWuD under LOOA and SOOA in Fig. 3(c). For simplicity, we present results for β= 0.05 and 0.3. These results confirm that EWuD always outperforms IMEW under both ONU architectures. It is interesting to notice from this figure that EWuD in LOOA provides better energy performance than that of the results of EWuD in SOOA over the low arrival rate region (e.g. 0.1 frame/ms $\leq \lambda \leq$ 1 frame/ms in case of β= 0.05). However, as λ increases, we can observe that EWuD in SOOA ends up showing less energy consumption compared to EWuD in LOOA. Authors in [6], [15] report that SOOA consumes less energy than LOOA when sleep interval length is short, and vice versa. Therefore, we conclude that LOOA's energy consumption performance is better than that of SOOA in low λ regions in both IMEW and EWuD. Additionally, β value in EWuD controls when LOOA can outperform SOOA under frame arrival rate (see Fig. 3(c)).

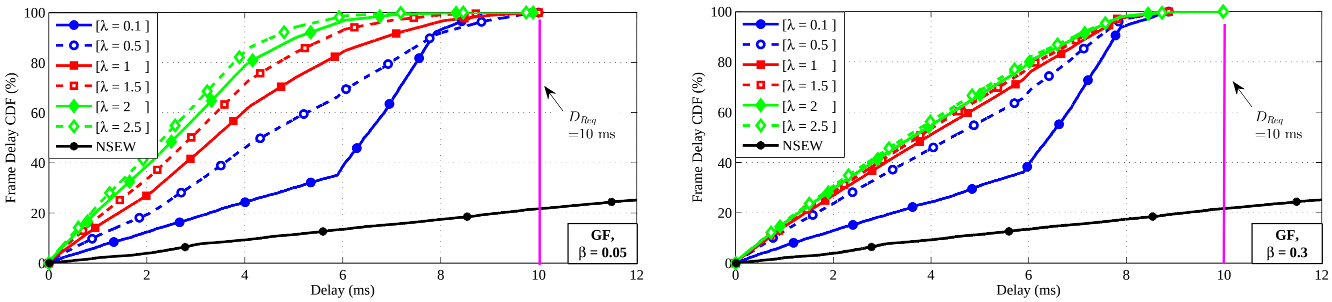
B. *Performance under High Priority Frames*

In this subsection, we are interested to investigate the performance of EWuD, NSEW and IMEW under GF and AF frames. Here, for simplicity, we present delay and energy performances under β= 0.05 and 0.3. Performance results are obtained based on LOOA. Furthermore, we assumed that D_{Req} for GF and AF frames is 10 ms and 25 ms, respectively. In this case, we present results for λ= {0.1, 0.5, 1, 1.5, 2, 2.5} frame/ms.

Frame delay Cumulative Distribution Function (CDF) results presented in Fig. 4(a) for GF frames when β= 0.05 show that proposed EWuD successfully meet delay requirement of 100% of upstream frames for all λ values. If we notice frame delay CDF result of NSEW, we can observe that 78% of upstream frames violate 10 ms delay requirement. This is because under NSEW, whenever the ONU moves into sleep state it has to stay in that state until sleep period expires, thereby increasing upstream frame delay significantly.

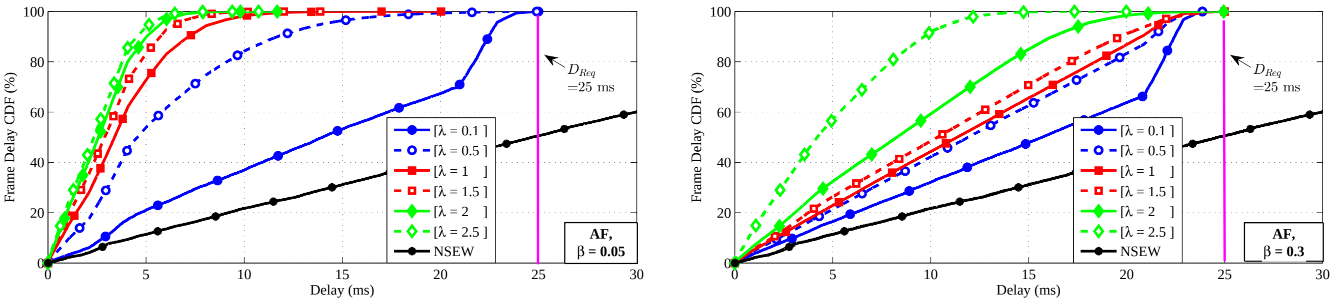
Even, in our solution, delay requirement is satisfied for all λ values, we can notice that CDF results of EWuD for different λ values significantly differ from each other. For example, in case of λ= 0.1 frame/ms, only 38% frames have delay below 6 ms, whereas, almost 99% of frames experience delay below 6 ms when λ= 2.5 frame/ms. The reason to explain this is as follows. We have noticed earlier in Fig. 3(a) that with the increment of λ, an ONU's average sleep duration reduces. This actually happens because as λ increases, the time between when the ONU moves into sleep state and when the ONU execute EWF gets shorter in EWuD due to increasing possibility of buffer overflow probability overpassing β (i.e. the condition presented in Eq. (4) holds). This implies that at high λ regions, average amount of time the upstream frames wait in ONU's buffer to be forwarded is less than that of low λ regions. Consequently, during high λ regions, upstream frames experience less delay than a low λ regions (see Fig. 4(a)). Conversely, in case when λ is low (e.g. λ= 0.1 frame/ms), in EWuD, EWF triggering decision is mostly influenced by delay violation possibility (i.e. the conditions presented in Eq. (5) and (6) hold) when there exists high priority upstream frames (e.g. GF frames) in an ONU. This allows an ONU in EWuD to hold upstream frames in its buffer longer for a low λ value (e.g. λ= 0.1 frame/ms) than a relatively high λ value (e.g. λ= 2 frame/ms) in which ONU's buffer overflow probability overpasses β frequently. As a result, in low λ regions, upstream frames experience higher delay than that of relatively higher λ regions in proposed EWuD.

Next, we present performance evaluation of EWuD and NSEW under GF frames when β= 0.3 in Fig. 4(b). Similar to β= 0.05 case, the EWuD meets delay requirement of all upstream frames successfully. However, it is interesting to notice from this figure that apart from λ= 2.5 frame/ms case,



(a) GF frames satisfy delay requirement when β= 0.05.

(b) GF frames satisfy delay requirement when β= 0.3.



(c) AF frames satisfy delay requirement when β= 0.05.

(d) AF frames satisfy delay requirement when β= 0.3.

Fig. 4: Delay performance under high priority frames.

in all other cases the maximum amount of frame delay reaches up to 9 ms. The reason to explain this is that, in this particular case, the execution of EWF in EWuD is mainly based on D_{Req} violation possibility. Note that here the buffer overflow probability is less likely overpass a large value of β (i.e. 0.3), thus β is not playing key role to execute EWF in EWuD's decision process when λ < 2.5 frame/ms. It is worth noting that when EWuD executes EWF considering D_{Req} violation possibility, it executes EWF two DBA cycles earlier than delay requirement violation point, so that none of the frames exceeds its delay requirement (the purpose of executing EWF two DBA cycles earlier is explained in subsection III-B). As a consequence, the maximum frame delay could reach up to 9 ms when λ < 2.5 frame/ms in EWuD.

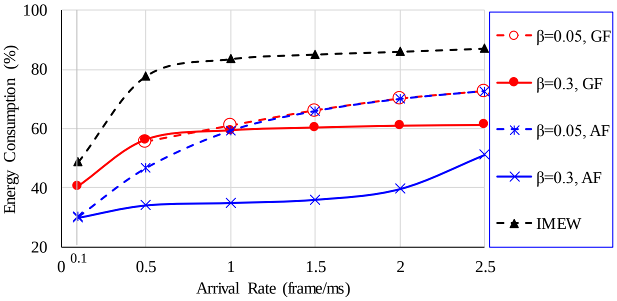


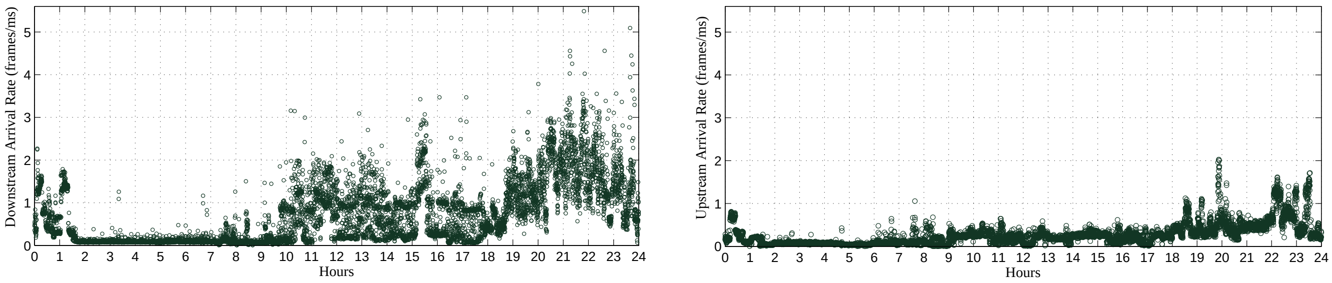
Fig. 5: Energy consumption of EWuD and IMEW under GF and AF frames.

Figure 4(c) presents EWuD and NSEW delay CDF results when β= 0.05. Similar to GF frames case, for β= 0.05, the execution of EWF in EWuD is mainly triggered due to buffer overflow probability overpassing β over high λ regions. As this is a relaxed delay requirement case compared to GF frames case, D_{Req} violation possibility does not have any influence on EWuD's EWF triggering decision process under a small β value (e.g. 0.05). That is the case(a) of EWuD as explained in subsection III-C. For instance, when λ= 1 frame/ms, the maximum frame delay could reach up to 20 ms which is far less than the AF delay requirement value (i.e. 25 ms) (see Fig. 4(c)). This implies that over high λ regions, D_{Req} violation possibility never affects EWuD's EWF triggering decision process. However, when λ is very low (e.g. 0.1 frame/ms), D_{Req} violation possibility dominates EWuD's EWF triggering decision process, similar to GF frame case which we can observe from Fig. 4(a) when λ= 0.1 frame/ms and β= 0.05.

Figure 4(d) demonstrates results under AF frames when β= 0.3. As β is larger than the previous case, EWuD's EWF triggering occurs less frequent than the previous case, thus forcing upstream frames to stay longer in ONU's buffer. Consequently, these results in worsening frame delay CDF performance in β= 0.3 than the results in β= 0.05 case (but none of the frames violates delay requirement in β= 0.3 case).

We can observe from Fig. 4(c) and 4(d) that NSEW fails to meet delay requirement of all upstream frames (it only meets delay requirement of 51% of frames). Results show that EWuD once again outperforms NSEW (all upstream frames meet delay requirement).

Results in Fig. 5 present relative energy consumption of an ONU under GF and AF frames. When β= 0.05, we can notice that under low λ (i.e. 0.1 and 0.5 frame/ms) ONU under AF frames consumes less energy than that of the results under GF frames. The reason behind this is that AF frames



(a) Downstream frame arrival rate.

(b) Upstream frame arrival rate.

Fig. 6: Downstream and upstream frame arrival rates of an ONU having 16 users during 24 h [6].

have relaxed D_{Req} (25 ms) compared to GF frames (10 ms). This allows EWuD to prolong EWF execution more under AF frames case than that of under GF frames case, thus resulting in increasing average sleep duration of an ONU under AF frames. As a consequence, energy consumption results of an ONU is noticeably less under AF frames than that of the consumption results when the ONU has GF frames. However, it is interesting to notice from Fig. 5 that with the increment of λ, when β= 0.05, an ONU's energy consumption results under both GF and AF frames show similar behavior. This happens because in this case EWuD's EWF triggering decision process is mainly due to buffer overflow probability overpassing β (as we noticed in Fig. 3(a)). This leads us to conclude that when λ is high (e.g. λ = 2 frame/ms) and β has a small value (e.g. β= 0.05), under both AF and GF frames, an ONU ends up showing similar energy consumption performance in proposed EWuD.

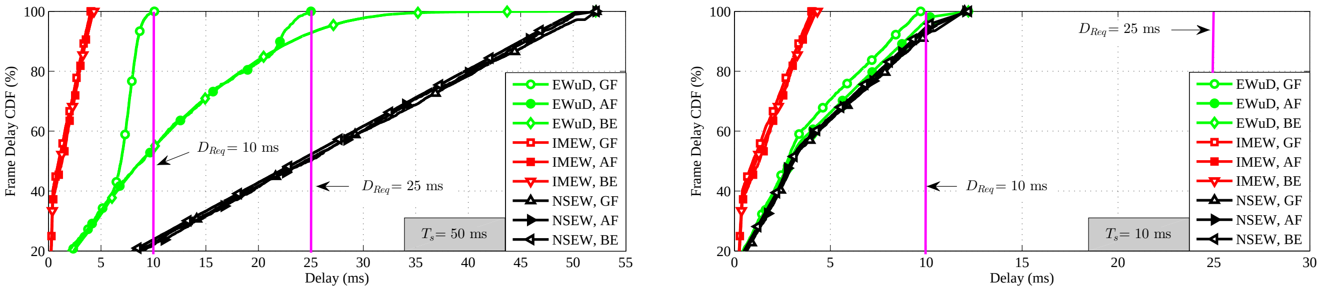
An ONU's energy consumption performance under GF and AF frames when β= 0.3 is also depicted in Fig. 5. This figure demonstrates that under GF frames an ONU consumes less energy under low λ (i.e. $\lambda < 0.5$ frame/ms). However, when $\lambda \geq 1$ frame/ms, energy consumption performance slowly increases as λ grows up. Similarly, in case of AF frames when β= 0.3, energy consumption slowly rises as λ increases. It is worth noticing that in this case we set large β value (0.3), thereby reducing the possibility of buffer overflow probability overpassing β under both AF and GF frames cases (i.e. the possibility of occurring the case(a) of EWuD reduces (see subsection III-C)). Therefore, in this particular case, delay violation dominates EWuD's EWF triggering decision process (i.e. the case(b) of EWuD), and thus allowing an ONU under both AF and GF frames to stay in sleep state as long as D_{Req} violation possibility does not occur under low λ regions (e.g. $\lambda \leq 2.5$ frame/ms under GF frames and $\lambda < 2.0$ frame/ms under AF frames). Consequently, in EWuD, an ONU stays in sleep state almost the same amount of time always regardless of any λ value below 2.5 frame/ms under GF frames (this also happens to an ONU under AF frame when $\lambda < 2.0$ frame/ms). However, we can notice that under AF frames when β= 0.3 and $\lambda \geq 2.0$ frame/ms, an ONU's energy consumption performance rapidly increases. The reason to explain this is that when λ is high, EWuD's EWF triggering decision process is mainly due to buffer overflow probability overpassing β (i.e. the case(a) of EWuD), thereby, reducing ONU's average sleep duration significantly. And thus, this leads towards increasing ONU's energy consumption noticeably under AF frames. This figure also shows that IMEW always consumes significantly large amount of energy compared to our proposed EWuD.

C. *Performance under Combined Frames Scenarios*

In a real deployed TDM-PON, there would be presence of combined CoS frames. Therefore, here, we put an effort to understand how proposed EWuD performs when there exist combined CoS frames in a TDM-PON. In order to do so, we evaluate performance of EWuD under wide range of arrival rates which we obtain based on real network traffic trace provided in [6] (authors in [6] collected both upstream and downstream traffic trace for 24 hours time span (see Fig. 6), considering that there exist 16 users served by an ONU). In our simulation, we choose the maximum arrival rate value during each hour for both upstream and downstream traffic from the provided traffic trace in [6]. In this manner, we consider 24 arrival rate values for each TDM-PON communication link in our simulation environment. In this performance evaluation, we set β= 0.3. Furthermore, here, we configure the presence of upstream frames assuming the ratios of GF, AF and BE frames are 13%, 34.8% and 52.2%, respectively, as considered in [28].

Figure 7(a) presents delay CDF results for EWuD, IMEW and NSEW when T_s= 50 ms. This figure imparts that, in EWuD, both GF and AF frames meet their delay requirement successfully. Similarly, we can observe that, under IMEW, 100% of GF and AF frames meets their delay requirement. In IMEW, the maximum frame delay can reach up to 5 ms. This happens because an ONU in IMEW executes EWF on an upstream frame arrival instantly, thereby, reducing upstream frames delay noticeably at the price of maximizing an ONU's energy consumption. Delay CDF performance when NSEW is in place is not satisfactory compared to EWuD and IMEW, as can be noticed from this figure. In this case, only 23% and 50% of GF and AF frames, respectively, can meet their corresponding delay requirement.

Relative energy consumption results of EWuD, IMEW and NSEW compared to AAS under this combined frame scenario are presented in Fig. 8. Figure 8(a) presents relative energy consumption of these three solutions in each hour of the day,



(a) Delay performance under T_s= 50 ms

(b) Delay performance under T_s= 10 ms

Fig. 7: Delay performance under combined CoS frames with different T_s.

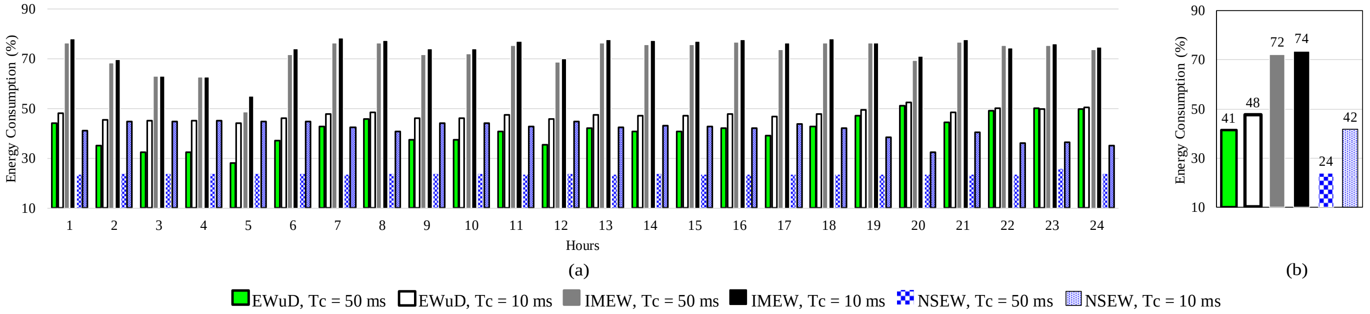


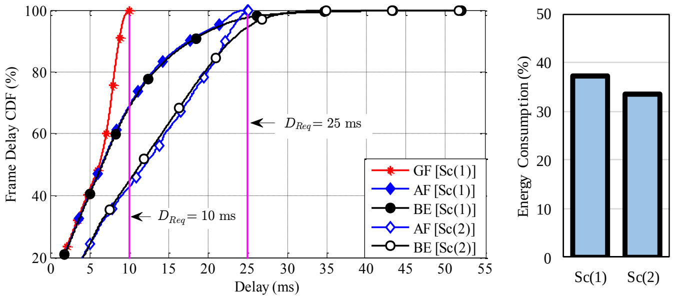
Fig. 8: Relative energy consumption of EWuD IMEW and NSEW compared to AAS under GF and AF frames: (a) Energy consumption every hour. (b) Global energy consumption.

and Fig. 8(b) depicts the global energy consumption. If we observe global energy consumption results presented in Fig. 8(b), we can notice that NSEW is showing the least energy consumption among all these solutions (it consumes 24% of AAS energy consumption). We can also notice from this figure that IMEW is showing the worst energy consumption, accounting 72% of AAS's consumption. On the other hand, proposed EWuD consumes 41%.

To understand the behavior of proposed EWuD under a small sleep interval length, we configured T_s= 10 ms, as considered in [29]. We can observe from Fig. 7(b) that EWuD can still successfully meet the delay requirement of all upstream frames. Likewise to the previous case (T_s= 50 ms), IMEW outperforms EWuD and NSEW in terms of delay CDF results. However, when T_s= 10 ms, NSEW shows better performance compared to its performance results when T_s= 50 ms. The reasons to explain this is that NSEW allows an ONU to forward upstream frames after sleep interval completion. Therefore, the smaller the T_s, the less the delay experienced by upstream frames in NSEW. Nevertheless, once again, NSEW fails to meet the delay requirement of all upstream frames even when T_s= 10 ms. Note that, in this case, an ONU's sleep interval length (T_s) and delay requirement (D_{Req}) of GF frames are the same. Therefore, one could expect that all GF frames should meet delay requirement in NSEW. However, we can notice from Fig. 7(b) that around 5% of GF frames still fails to meet the delay requirement. The reason behind this is that when an ONU leaves sleep state it requires at most two DBA cycles to get an upstream transmission grant from the OLT (the reason behind this is explained in subsection III-B). Therefore, even when T_s= D_{Req} in NSEW, the frames at the head of the queue of an ONU's buffer could experience additional delay if the ONU is not allocated an upstream transmission grant immediately after moving into active state. It is worth to mention here that some applications in real communication network has very strict delay requirement (e.g. delay sensitive Smart Grid applications [26]). This indicates that even T_s is set equal to D_{Req}, NSEW can end up not meeting the delay requirement 100% of frames. This leads us to conclude that NSEW should not be considered as a practical solution in a TDM-PON system, where meeting frame delay requirement is given more importance over its energy saving performance.

We can notice the relative global energy consumption of these three solutions compared to AAS's energy consumption when T_s= 10 ms in Fig. 8(b). Results presented in this figure show that energy consumption of EWuD, IMEW and NSEW is 48%, 74% and 42%, respectively. Although, it appears that NSEW provides the best energy saving performance, delay CDF performance of this solution is not satisfactory (it cannot meet 100% of frame delay requirement of both AF and GF frames (see Fig. 7)). Therefore, this leads us to conclude that proposed EWuD not only meets delay requirement under all kind of frames but also can reduce energy consumption of an ONU significantly.

At this point, we are interested to observe how presence of high priority frames (e.g. GF frames) could affect delay performance of comparatively low priority frames. Additionally, we want to understand the influence of high priority frames on ONU's energy consumption performance. To do



(a) (b)

Fig. 9: Performances under *Sc(1)* and *Sc(2)*: (a) Delay performance. (b) Relative energy consumption.

so, we consider two scenarios: (1) an ONU has GF, AF, and BE frames (we refer to this as *Sc(1)*) and (2) an ONU has only AF and BE frames (we refer to this as *Sc(2)*). In this particular case, we set T_s= 50 ms, β= 0.3 and λ= 2 frame/ms (overall upstream frame arrival). In case of *Sc(1)*, we assume the ratios of GF, AF and BE frames are 13%, 34.8% and 52.2%, respectively. On the other hand, in case of *Sc(2)*, we suppose that there are no GF frames, and the ratios of AF and BE frames is 34.8% and 65.2%, respectively.

We can notice from Fig. 9(a) that AF frames delay CDF results under *Sc(2)* are showing noticeably worse performance compared to that of the AF frames delay results under *Sc(1)*. For instance, in *Sc(2)*, only 43% of AF frames have delay below 10 ms, whereas, under *Sc(1)*, around 68% of upstream AF frames have delay below 10 ms (see Fig. 9(a)). The reason behind is that whenever an ONU in EWuD has GF frames, it should set the EWF execution time taking into consideration delay requirement of GF frames (GF frames have more strict delay requirement than AF frames). Then, in this case, if the allocated upstream slot for the ONU is not totally occupied with the GF frames, the AF and BE frames are forwarded along with the GF frames. And consequently, AF and BF frames end up experiencing less delay in *Sc(1)* compared to *Sc(2)*. On return, relative ONU energy performance result depicted in Fig. 9(b) shows that an ONU consumes slightly more energy in *Sc(1)* than in *Sc(2)*. Therefore, we can conclude that, in EWuD, the presence of different CoS frames can noticeably influence upstream frame delay and ONU's energy consumption performance.

V. DISCUSSION ON SELECTING β VALUE

We have found from results that, in EWuD, along with presence of upstream frames having high CoS priorities (e.g. AF and GF), β can influence delay and energy consumption performance of an ONU. As mentioned in subsection III-A, finding β value is not in the scope of this paper. However, unlike authors in [16] (where β is set to 0.25 and 0.6), our aim is to observe the performance of EWuD under wide range of β (specially for BE frames). In order to set β value for ONUs in a TDM-PON when our solution is in place, in this section, we provide a brief discussion which can be useful for TDM-PON operators.

In fact, over the last several years there have been noticeably dedicated efforts in finding β (e.g. [17]–[20]). Most of the research efforts adopted different optimization approaches to find the value of β. The most common parameters to decide β value are: arrival rate, buffer size, link speed [17]–[20].

A TDM-PON operator needs to take into account that different ONUs can have different traffic arrival behavior throughout a day. Additionally, it is realistic that, in a single TDM-PON system, ONUs may not have the same buffer size. Therefore, we recommend that operators who adopt EWuD need to set β value dynamically for each of the ONUs taking into consideration those aforementioned factors. Optimization tools from the existing research (e.g. [17]–[20]) can be useful means to find β value for TDM-PON operators. Otherwise, they can rely on a heuristic technique (algorithm) in order to set β value for each of the ONUs dynamically. This heuristic algorithm may work as follows. The operator can choose an initial β value, which is very small, for an ONU. This initial value can be increased for a particular arrival rate at a given time as long as a frame drop ratio overpasses a threshold value. This threshold value can be set based on required service level agreement for a particular ONU. Note that comparing between different approaches to find β is also not the scope of our research in this paper. We leave this for future research, as our primary concern here is to design early wake-up triggering algorithm for ONUs.

VI. CONCLUSION

To the best of our knowledge, EWuD is the first solution towards introducing novel EWF execution decision process in order to meet frames delay requirement and avoid buffer overflow, while saving ONU's energy as much as possible. In this regard, our proposed EWuD triggers EWF based on delay requirement of upstream frames and ONU buffer overflow probability. Our proposed EWuD has been evaluated under different frames' CoS priorities and ONU architectures (i.e. LOOA and SOOA). We have compared proposed EWuD results with NSEW and IMEW solutions. Results reveal that EWuD and IMEW meet 100% of delay requirements of all upstream frames under wide range of β and λ. Although IMEW can satisfy delay requirement of all frames, it shows the worst energy consumption performance among all three solutions. On the other hand, NSEW consumes slightly less energy than EWuD. However, a noticeable amount of upstream frames does not meet delay requirement in NSEW. Therefore, it appears that EWuD can be a very practical solution as it minimizes ONU's energy consumption noticeably and satisfies always upstream frame delay requirement. In our future research, to find the most suitable β value finding approach for EWuD, we want to conduct a rigorous performance evaluation of EWuD under different existing β value finding approaches.

ACKNOWLEDGMENT

This work was supported by the ICT R&D program of MSIP/IITP, Republic of Korea. [1391104001, Research on Communication Technology using Bio-inspired Algorithm].

REFERENCES

[1] Y. Zhang, P. Chowdhury, M. Tornatore, and B. Mukherjee, "Energy efficiency in telecom optical networks," *Communications Surveys & Tutorials, IEEE*, vol. 12, no. 4, pp. 441–458, 2010.

[2] L. Shi, B. Mukherjee, and S.-S. Lee, "Energy-efficient PON with sleep-mode ONU: progress, challenges, and solutions," *Network, IEEE*, vol. 26, no. 2, pp. 36–41, 2012.

[3] "IEEE Standard for Service Interoperability in Ethernet Passive Optical Networks (SIEPON)," *IEEE Std 1904.1-2013*, pp. 1–834, 2013.

[4] ITU-T, "G.Sup45: GPON power conservation," 2009. [Online]. Available: http://www.itu.int/rec/T-REC-G.Sup45-200905-I/en

[5] ITU-T Recommendation G988, "ONU management and control interface (OMCI) specification," 2012. [Online]. Available: https://www.itu.int/rec/T-REC-G.988/en

[6] S. S. Newaz, Á. Cuevas, G. M. Lee, N. Crespi, and J. K. Choi, "Adaptive Delay-Aware Energy Efficient TDM-PON," *Computer Networks*, vol. 57, no. 7, pp. 1577–1596, 2013.

[7] R. Kubo, J.-i. Kani, H. Ujikawa, T. Sakamoto, Y. Fujimoto, N. Yoshimoto, and H. Hadama, "Study and demonstration of sleep and adaptive link rate control mechanisms for energy efficient 10G-EPON," *Journal of Optical Communications and Networking*, vol. 2, no. 9, pp. 716–729, 2010.

[8] F. Y. M. Alaelddin, S. S. Newaz, J. Lee, M. R. Uddin, G. M. Lee, and J. K. Choi, "Performance analysis of TCP traffic and its influence on ONUs energy saving in energy efficient TDM-PON," *Optical Fiber Technology*, vol. 26, Part B, pp. 190 – 200, 2015.

[9] S. Nishihara, N. Miura, N. Tanaka, K.-I. Suzuki, and N. Yoshimoto, "Effectiveness of SIEPON package B compliant ONU sleep technique and impact on latency and transmission efficiency from a theoretical point of view," *Journal of Optical Communications and Networking*, vol. 6, no. 4, pp. 362–370, 2014.

[10] J.-i. KANI, Y. FUJIMOTO, N. YOSHIMOTO, and K. KUMOZAKI, "Adaptive power saving mechanism for 10 gigabit class pon systems," *IEICE transactions on communications*, vol. 93, no. 2, pp. 280–288, 2010.

[11] S. S. Newaz, Á. Cuevas, G. M. Lee, N. Crespi, and J. K. Choi, "Evaluating energy efficiency of ONUs having multiple power levels in TDM-PONs," *Communications Letters, IEEE*, vol. 17, no. 6, pp. 1248–1251, 2013.

[12] F. Y. M. Alaelddin, S. S. Newaz, and C. Jun Kyun, "Modeling and simulation of EPON with sleep mode enabled using OPNET," in *ICTC, 2014*, Conference Proceedings, pp. 16–21.

[13] B. Skubic, J. Chen, J. Ahmed, L. Wosinska, and B. Mukherjee, "A comparison of dynamic bandwidth allocation for EPON, GPON, and next-generation TDM PON," *Communications Magazine, IEEE*, vol. 47, no. 3, pp. S40–S48, 2009.

[14] G. Kramer, B. Mukherjee, and G. Pesavento, "Interleaved polling with adaptive cycle time (IPACT): a dynamic bandwidth distribution scheme in an optical access network," *Photonic Network Communications*, vol. 4, no. 1, pp. 89–107, 2002.

[15] S.-W. Wong, L. Valcarenghi, S.-H. Yen, D. R. Campelo, S. Yamashita, and L. Kazovsky, "Sleep mode for energy saving PONs: advantages and drawbacks," in *GLOBECOM, 2009*. IEEE, Conference Proceedings, pp. 1–6.

[16] G. K. Wong, Q. Zhang, and D. H. Tsang, "Switching cost minimization in the IEEE 802.16 e mobile WiMAX sleep mode operation," *Wireless Communications and Mobile Computing*, vol. 10, no. 12, pp. 1576–1588, 2010.

[17] B. Tsybakov and N. D. Georganas, "Self-similar traffic and upper bounds to buffer-overflow probability in an ATM queue," *Performance Evaluation*, vol. 32, no. 1, pp. 57–80, 1998.

[18] A. Bhattacharjee and S. Nandi, "Self-similar traffic and buffer overflow: A bounded buffer allocation approach," in *India Conference (INDICON), 2009 Annual IEEE*, Dec 2009, pp. 1–4.

[19] A. A. Wahid and J. A. Schormans, "Bounds on accuracy when estimating the loss probability in a packet buffer," in *Computer Science and Electronic Engineering Conference (CEEC), 2015 7th*. IEEE, 2015, pp. 24–28.

[20] L. Globa and M. Slukysh, "Nodal routing with traffic classification," *Polish association for knowlage management Series: Studies&Proceedings*, no. 42, pp. 37–46, 2011.

[21] S.-i. Choi, "Cyclic polling-based dynamic bandwidth allocation for differentiated classes of service in ethernet passive optical networks," *Photonic Network Communications*, vol. 7, no. 1, pp. 87–96, 2004.

[22] S. S. Newaz, A. Mohammed, G. M. Lee, and J. K. Choi, "Energy efficient and latency aware tdm-pon for local customer internetworking," in *Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on*, 2015, pp. 1184–1189.

[23] "IEEE Std 802.3ah-2004, Part 3: CSMA/CD Access Method and Physical Layer Specifications Amendment: Media Access Control Parameters, Physical Layers, and Management Parameters for Subscriber Access Networks," 7 September 2004.

[24] I. Hwang, J.-Y. Lee, Z.-D. Shyu *et al.*, "A novel dynamic bandwidth allocation mechanism for star-ring-based EPON," *ISRN Communications and Networking*, vol. 2011, p. 12, 2011.

[25] L. Cai, Y. Xiao, J. W Mark *et al.*, "VoIP over WLAN: Voice capacity, admission control, qos, and MAC," *International Journal of Communication Systems*, vol. 19, no. 4, pp. 491–508, 2006.

[26] Alcatel/Lucent, "Smart choices for the smart grid," Feb. 2011. [Online]. Available: http://www.smartgrids-cre.fr/media/documents/Alcatel-Lucent_SmartChoicesfortheSmartGrid_TWP_EN.pdf

[27] R. Deepalakshmi and D. S. Rajaram, "A novel medium access control protocol for routing multimedia traffic in optical networks by exploiting delays with improved dynamic bandwidth allocation," *ARPN Journal of Systems and Software*, vol. 1, no. 7, 2011.

[28] A. T. Liem, I.-S. Hwang, and A. Nikoukar, "Multiple link faults restoration mechanism in an enhanced EPON architecture," in *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 1, 2014, pp. 227–232.

[29] A. Nikoukar, I.-S. Hwang, C.-J. Wang, M. S. Ab-Rahman, and A. T. Liem, "A SIEPON based transmitter sleep mode energy-efficient mechanism in EPON," *Optical Fiber Technology*, vol. 23, pp. 78 – 89, 2015.

Alaelddin Fuad Yousif Mohammed is a Ph.D candidate at Korea Advanced Institute of Science and Technology (KAIST), South Korea. He received B.Sc. degree in computer engineering from university of Gezira, Sudan, M.Sc. degree in computer network from Royal Institute of Technology (KTH), Sweden and M.Sc. degree engineering and management of information systems, from KTH, Sweden. He works as a lecturer at the university of Gezira, Sudan since 2007 to date. His research interests include energy saving in access networks, modeling and simulation of computer networks and Internet protocols.

S.H. Shah Newaz is a Post-doctoral researcher at Korea Advanced Institute of Science and Technology (KAIST), South Korea. His research interests include energy-efficient passive optical networks, optical and wireless converged networks, mobility and energy efficiency issue in wireless network, local cloud/fog computing, smart grid and content delivery network, all with specific focus, mainly on protocol design and performance aspects.

Mohammad Rakib Uddin is an associate professor at the department of electrical and electronic engineering, faculty of engineering, Universiti Teknologi Brunei (UTB), Brunei Darussalam. His research interest includes the fiber optics, optical communications and, micro photonic/electronic devices and integrated circuits.

Gyu Myoung Lee (S02, M07, SM12) is with the Liverpool John Moores University (LJMU), UK, as a Senior Lecturer from 2014 and with KAIST, Korea, as an adjunct professor from 2012. His research interests include Internet of things, future networks, multimedia services, and energy saving technologies including smart grids. He has been actively working for standardization in ITU-T, IETF and oneM2M, etc.

Jun Kyun Choi is a professor at the Korea Advanced Institute of Science and Technology (KAIST). His research interests include next-generation network (NGN) issues, energy-efficient networks, and the Internet of Things. 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.