Efficient Bandwidth Management for Ethernet Passive Optical Networks

Dissertation by

Amr Elsayed Mostafa Elrasad

In Partial Fulfillment of the Requirements for the Degree of

Doctor of Philosophy

King Abdullah University of Science and Technology, Thuwal,

Kingdom of Saudi Arabia

May, 2016

The dissertation of Amr Elsayed Mostafa Elrasad is approved by the examination committee

Committee Chairperson: Prof. Basem Shihada

Committee Member: Prof. David Keyes

Committee Member: Prof. Mohamed-Slim Alouini

Committee Member: Prof. Biswanath Mukherjee

Copyright ©2016

Amr Elsayed Mostafa Elrasad

All Rights Reserved

ABSTRACT

Efficient Bandwidth Management for Ethernet Passive Optical Networks

Amr Elsayed Mostafa Elrasad

The increasing bandwidth demands in access networks motivates network operators, networking devices manufacturers, and standardization institutions to search for new approaches for access networks. These approaches should support higher bandwidth, longer distance between end user and network operator, and less energy consumption.

Ethernet Passive Optical Network (EPON) is a favorable choice for broadband access networks. EPONs support transmission rates up to 10 Gbps. EPONs also support distance between end users and central office up to 20 Km. Moreover, optical networks have the least energy consumption among all types of networks.

In this dissertation, we focus on reducing delay and saving energy in EPONs. Reducing delay is essential for delay-sensitive traffic, while minimizing energy consumption is an environmental necessity and also reduces the network operating costs. We identify five challenges, namely excess bandwidth allocation, frame delineation, congestion resolution, large round trip time delay in long-reach EPONs (LR-EPONs), and energy saving. We provide a Dynamic Bandwidth Allocation (DBA) approach for each challenge. We also propose a novel scheme that combines the features of the proposed approaches in one highly performing scheme.

Our approach is to design novel DBA protocols that can further reduce the delay and be simultaneously simple and fair. We also present a dynamic bandwidth allocation scheme for Green EPONs taking into consideration maximizing energy saving under target delay

constraints.

Regarding excess bandwidth allocation, we develop an effective DBA scheme called Delayed Excess Scheduling (DES). DES achieves significant delay and jitter reduction and is more suitable for industrial deployment due to its simplicity. Utilizing DES in hybrid TDM/WDM EPONs (TWDM-EPONs) is also investigated. We also study eliminating the wasted bandwidth due to frame delineation. We develop an interactive DBA scheme, Efficient Grant Sizing Interleaved Polling (EGSIP), to compensate the unutilized bandwidth due to frame delineation. Our solution achieves delay reduction ratio up to 90% at high load. We also develop a Congestion Aware Limited Time (CALT) DBA scheme to detect and resolve temporary congestion in EPONs. CALT smartly adapts the optical networking unit (ONU) maximum transmission window according to the detected congestion level. Numerical results show that CALT is more robust at high load compared to other related published schemes.

Regarding LR-EPONs, the main concern is large round trip delay mitigation. We address two problems, namely bandwidth over-granting in Multi-Thread Polling (MTP) and on-the-fly void filling. We combine, with some modifications, EGSIP and DES to resolve bandwidth over-granting in MTP. We also manage to adaptively tune MTP active running threads along with the offered load. Regarding on-the-fly void filling, Our approach, Parallel Void Thread (PVT), achieves large delay reduction for delay-sensitive traffic. PVT is designed as a plus function to DBA and can be combined with almost all DBA schemes proposed before. The powerful feature of our proposed solutions is integrability. We integrate our solutions together and form a multi-feature, robust, fairly simple, and well performing DBA scheme over LR-TWDM-EPONs.

Our final contribution is about energy saving under target delay constraints. We tackle the problem of downstream based sleep time sizing and scheduling under required delay constraints. Simulation results show that our approach adheres to delay constraints and achieves almost ideal energy saving ratio at the same time.

ACKNOWLEDGEMENTS

All praise and thank to Allah who has given me the patience and the strength to finish this work.

This work is dedicated to my late parents, Elsayed Elrasad and Soad Elsinity. I would like to express my sincere gratitude and appreciation to my wife. She stood by me in the difficult moments and always showed full support, love, and understanding. I will be forever thankful to her.

This work would not be possible without my advisor Prof. Basem Shihada. I would like to thank him for his advice, mentoring, and support during my PhD journey.

I would also like to thank my PhD advisory committee members Prof. Biswanath Mukherjee, College of Engineering, University of California, Davis, Prof. David Keyes, and Prof. Mohamed-Slim Alouini, Computer, Electrical and Mathematical Science and Engineering Division, KAUST for the insightful discussions and guidance. I am honored to have a world class committee members to evaluate my dissertation.

I also appreciate the help of my friends, Ehab Abdelhamid, Ibrahim Abdelaziz, Dr. Zouheir Rezki, Ahmad Abdelfattah, Hassan Oubei, Amin Allam, Lau Chun Pong and Mohammad Khafagy.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

Fiber-to-the-X (FTTX) is proposed as a solution to increasing bandwidth demand in access networks. FTTX aims to deeply penetrate optical fiber to the user premises. X could be home (FTTH), curb (FTTC), or user premises (FTTP). Passive Optical Network (PON) is considered one of the main promising approaches to apply FTTX in the first mile. PON is popular for many reasons: it uses passive components between central office (CO) and user premises; it can support remote users over long distances up to 20 Km in legacy PONs and 100 Km in long-reach PONs (LR-PON). There exist two approaches for PON: Ethernet Passive Optical Network (EPON), standardized by IEEE; Gigabit-Capable Passive Optical Networks (GPON), standardized by ITU-T.

Conventional PON consists of single optical line terminal (OLT), located at a central office (CO), connected to a set of optical network units (ONUs) located at the end-user locations via a feeder optical fiber and passive splitter as shown in Figure 1.1 (presented in [1]). PON employs wavelength division multiplexing (WDM) to establish full duplex connectivity between OLT and ONUs. The passive splitter nature allows the downstream signal to be splitted and delivered to all ONUs (see Figure 1.2), however the upstream signal is only received by the OLT (see Figure 1.3). PON is considered as point to multi point in downstream direction, while considered multi point to one point in upstream direction. In order to coordinate bandwidth access between ONUs to avoid upstream data collisions, dynamic bandwidth allocation (DBA) algorithm is needed. In EPONs, OLT and ONUs

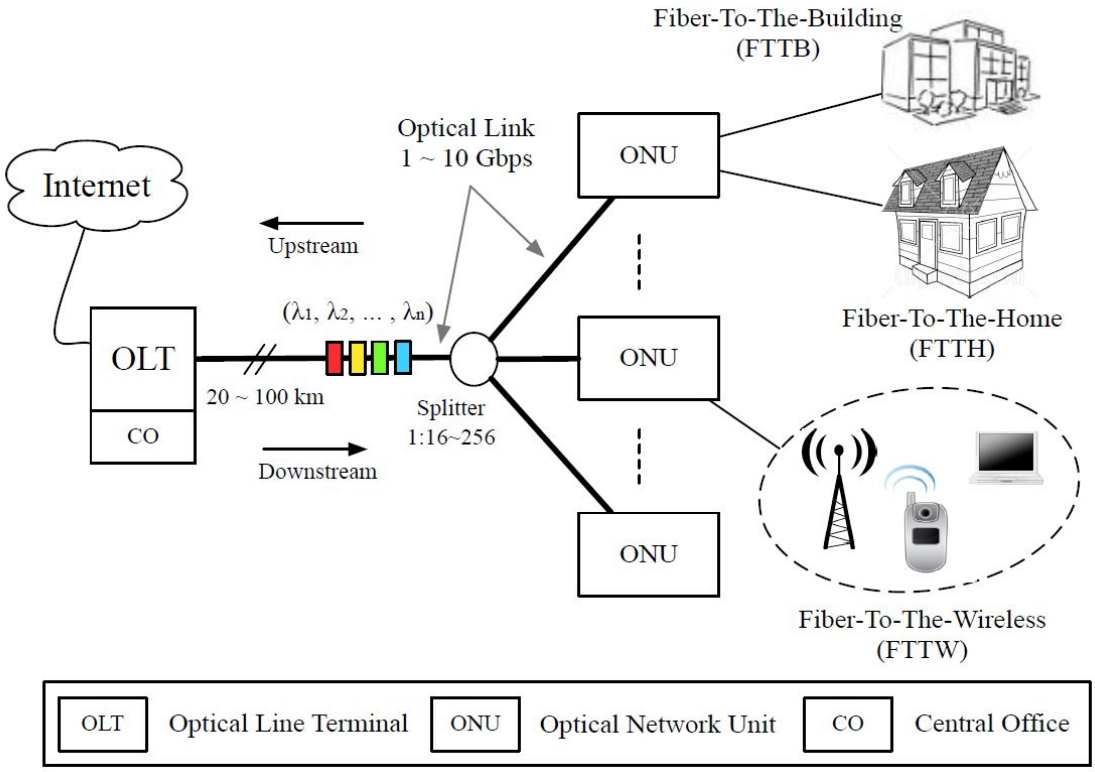


Figure 1.1: Passive Optical Network.

communicate through *Multi point Control Protocol* (MPCP), while the DBA algorithm is left to be vendor specific task.

1.1 Motivation and Challenges

DBA in EPONs is considered a very vital task and crucial to network performance. Both IEEE 802.3ah and 802.3av standards left the DBA to be vendor specific task. This opens the path to propose various DBAs that differ in their goals. These goals could be reducing delay, increasing bandwidth allocation fairness, supporting multiple services, and energy saving.

The main research theme of this dissertation is reducing delay and saving energy. Reducing delay improves network performance and leverage the network ability to carry delay sensitive traffic such as voice-over-IP (VoIP) and live video streaming. Saving energy has

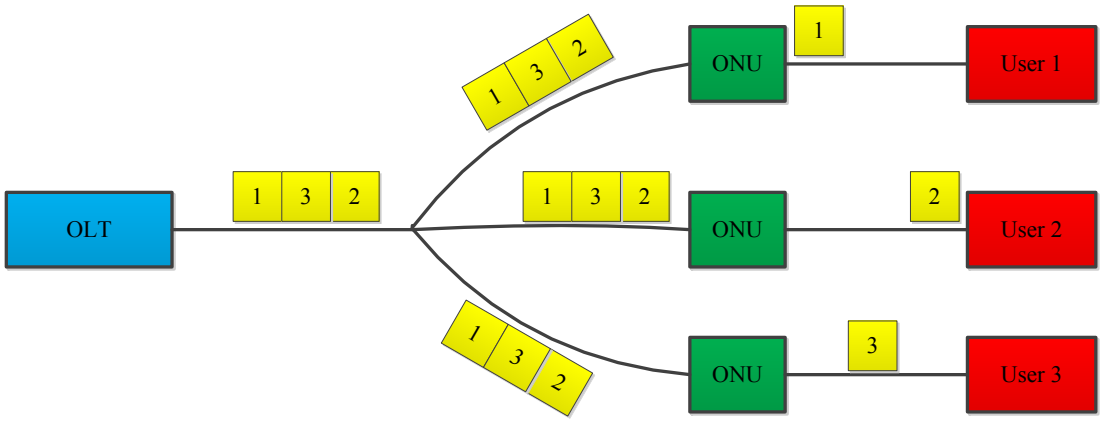


Figure 1.2: Downstream transmission.

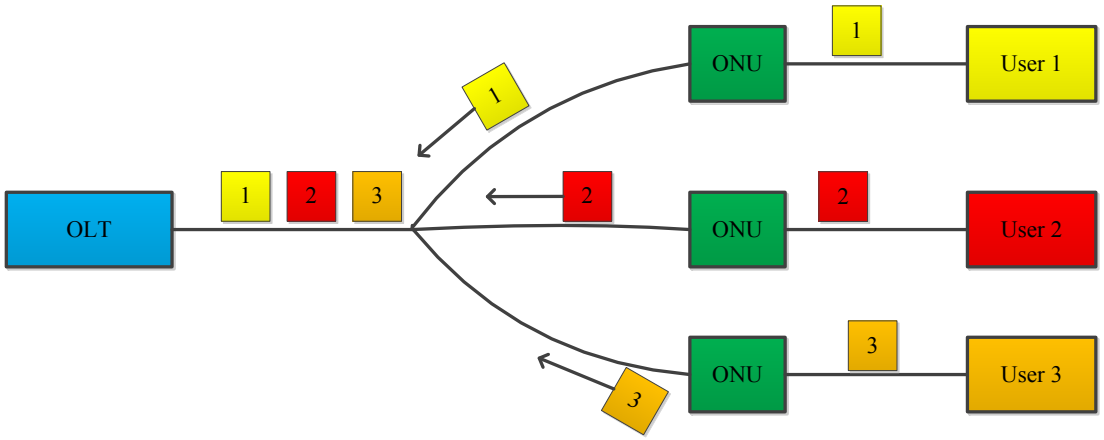


Figure 1.3: Upstream transmission.

two fold benefit: reducing CO_2 emissions and reducing operating costs for both network operator and end users.

In this dissertation, we identify five challenges, namely excess bandwidth allocation, frame delineation, congestion, large round trip time (RTT) effect in LR-EPONs, and downstream based DBA in Green EPONs.

Excess bandwidth is the unclaimed bandwidth of lightly loaded ONUs. Due to the bursty nature of access network traffic, some ONUs can be temporary overloaded, while the others are lightly loaded. Excess bandwidth of lightly loaded ONUs can be redistributed among the overloaded ONUs to quickly flush their upstream buffered data. This mechanism

has high impact on delay reduction at high load. Most of the proposed schemes for excess bandwidth allocation incur adding extra idle time. In this dissertation, we discuss how to allocate excess bandwidth in fair and simple way without incurring extra idle time.

Since Ethernet frames cannot be fragmented, ONUs suffer from frame delineation problem when a node is allocated its maximum transmission window. Frame delineation means that some of the allocated bandwidth cannot be utilized because the last frame cannot be fitted in. In [2], author investigated the wasted bandwidth caused by frame delineation and showed that frame delineation wastes 4% of bandwidth. We address this issue in this dissertation to better utilize the bandwidth and also to reduce delay at high load.

One of the interesting EPON research challenges is congestion. The bursty nature of access networks traffic could turn the network from lightly loaded to congested in very short time interval. We present a study about how to detect and resolve network congestion.

Long-reach EPONs is an attractive research trend in passive optical networks. The real challenge in LR-EPONs is to mitigate the round trip time (RTT) effect on average delay. Results presented in [3, 4] show that no presented single-thread DBA scheme is able to achieve delay below 1.5 $\overline{RTT}$, where $\overline{RTT}$ is the average round trip time. In LR-EPON, RTT can be up to 1 ms. This would negatively impact the delay-sensitive traffic performance.

Energy saving while maintaining required average delay constraint is a hard dilemma. Putting some ONUs to sleep might affect the average delay. The key challenge is how to schedule sleep/wakeup time without violating the required delay constraints.

1.2 Contribution

In this dissertation, our goal is to propose DBA schemes that mainly reduce the average packet delay, simple, fair, and operate under energy saving policy. Our research covers all kind of EPONs flavors. We investigate DBA in Time Division Multiplexed (TDM) EPONs,

Hybrid Time/Wavelength Division Multiplexed (TWDM), standard specified reach [5, 6], Long-Reach EPONs (LR-EPONs), and Green EPONs. Our approach to achieve our goal is to go through two phases. The first phase is to study and understand the queuing model of EPON and to identify the parameters controlling both the delay and bandwidth utilization. The second phase is to formulate the research challenges related to those parameters and develop novel schemes to resolve these challenges.

In this dissertation, we provide solutions to the challenges mentioned above. We propose a solution for each individual challenge and compare it to the benchmark related solutions in literature. The key point of this dissertation is integrability. Our proposed solutions are integrable and can be combined in multi-feature DBA scheme the resolves all the discussed challenges.

Regarding excess bandwidth allocation, we propose a simple, yet effective DBA scheme called Delayed Excess Scheduling (DES) [7]. DES reduces both the average delay and the delay jitter. Moreover, DES is practical and robust to be implemented in industry. We also present a wavelength based version of DES named Flexible DES (FDES). FDES is applied within the context of TWDM-EPONs that applies Dynamic Wavelength Assignment (DWA).

We investigate frame delineation problem in TWDM-EPONs. Our research leads to introduction of a novel concept of bandwidth compensation and a new DBA scheme called Efficient Grant Sizing Interleaved Polling (EGSIP). Our simulation experiment results indicate that removing and compensating wasted bandwidth can significantly reduce the average packet delay.

Dealing with temporary congestion is not an easy task for three reasons. First, DBA scheme has to quickly and accurately detect that there is congestion in upstream direction. Second, resolution should be also fast and fair. Third, what are the tuning parameters to resolve the congestion?. We develop a framework called Congestion Aware Limited Time (CALT) to address this challenge. CALT has its own criteria to detect temporary conges-

tion, defines its tuning parameters, and has a fast acting resolution mechanism. CALT smartly tunes the ONU maximum transmission window among pre-defined values when congestion is detected.

In LR-EPONS, large RTT highly impacts the average packet delay at low and medium load. We contribute to resolving this challenge in two directions. The first direction is Multi-Thread Polling (MTP) [8]. MTP is proposed to reduce polling delay and hence reduce the total average delay. MTP suffers from bandwidth over-granting among running threads. This problem leads to performance degradation. Moreover, MTP is not robust at high load compared to Single-Thread Polling (STP) [4] due to the increased bandwidth overhead. Our solution, Integrated Bandwidth Allocation (IBA), takes care of this problem and can achieve the minimum theoretical delay bounds of MTP. We also develop a framework to adaptively tune the number of active thread to enhance the performance along the whole load range. In fact, IBA can be seen as a combination of both DES and EGSIP. The second direction, is to detect and fill the voids (idle times) between bandwidth grants. We call our approach Parallel Void Thread (PVT) [9]. We provide a mathematical analysis of this technique and show that it can lead to significant delay reduction under reasonable conditions. One of PVT major advantages is that it can be considered as a plus function that and be combined with many DBA schemes appeared in literature.

Our next contribution focuses on solution integrability as we mentioned earlier. Within the context of LR-TWDM-EPONs, we combine, with some modifications, EGSIP, FDES, and PVT to constitute a high performing DBA scheme compared to the recent proposed DBA schemes in literature.

Our final contribution is to design a downstream scheduling policy for Green EPONs while maintaining a certain delay constraints. Our results show that the network can reach the ideal energy saving ratio.

1.3 Dissertation Outline

The rest of this dissertation is divided as follows: Chapter 2 presents a background review of EPONs and DBA schemes. DES scheme is presented in details in Chapter 3. Frame delineation problem is discussed in Chapter 4. Chapter 5 discusses congestion problem in EPONs. In Chapter 6, we present an effective solution to bandwidth over-granting problem associated with Multi-Thread Polling (MTP). A detailed investigation and analysis of void filling problem is presented in Chapter 7. In Chapter 8, we present an integrated DBA scheme that resolves all the features we discussed in the previous chapters. Energy saving in EPONs is discussed in Chapter 9. Finally, conclusions and future research directions are presented in Chapter 10.

Chapter 2

Background and State of the Art

In this chapter, we present an overview on Passive Optical Networks (PONS). This includes signaling protocols, mathematical model of PON, and benchmark dynamic bandwidth allocation (DBA) schemes .

2.1 Multi Point Control Protocol (MPCP)

In both IEEE 802.3ah [5] and IEEE 802.3av [6], MPCP is defined as a signaling protocol to exchange information between ONUs and OLT. MPCP defines six control messages: gate, report, register, register_req, and register_ack. Register, register_req, and register_ack control messages are used when an ONU first joins the network. In normal operation mode, both gate and report messages are used.

Report message, shown in Figure 2.1, is sent by ONU to inform OLT about its buffer size. The standard supports multiple threshold reporting. This means that the ONU can report multiple requests based on pre-defined threshold values. The standard also supports differentiated services (DiffServ) reporting.

Gate message, shown in Figure 2.2, is sent by the OLT to single ONU to let the ONU know the time and duration of its next upstream transmission. According to standard, gate message can support up to four grant allocations.

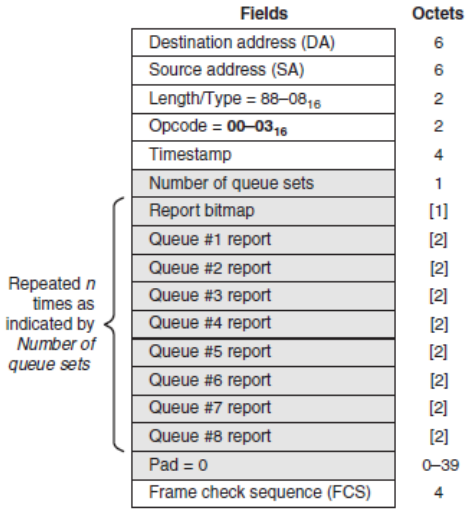


Figure 2.1: Report control frame.

2.2 Dynamic Bandwidth Allocation

As we mentioned before, a coordination is needed among ONUs to avoid upstream data collisions. The most straight forward method is to assign fixed transmission window for each ONU. This solution has poor bandwidth utilization as some ONUs might not have data to send [10] due to the bursty nature of access network traffic [11]. A dynamic approach that adapts the transmission window according to each ONU buffer size is much more convenient.

Dynamic bandwidth allocation (DBA) is the most vital task in EPONs. It mainly consists of two subtasks: grant sizing and grant scheduling. Grant sizing task determines the allocated bandwidth grant to each ONU in each cycle. Grant scheduling is the one responsible for scheduling both the start and end times of bandwidth grant.

DBA can be either run online, offline, or hybrid. In online scheme, the OLT runs the DBA algorithm after receiving the bandwidth request (report message) from an ONU (see Figure 2.3). The online scheme reduces the idle time gabs (voids) between grants but on the other hand the OLT decision is not optimal as it is based on single ONU request. In offline DBA, the OLT runs the DBA algorithm after collecting all bandwidth requests

Fields	Octets
Destination address (DA)	6
Source address (SA)	6
Length/type = $88\text{–}08_{16}$	2
Opcode = $\mathbf{00\text{–}02}_{16}$	2
Timestamp	4
Number of grants/flags	1
Grant #1 start time	[4]
Grant #1 length	[2]
Grant #2 start time	[4]
Grant #2 length	[2]
Grant #3 start time	[4]
Grant #3 length	[2]
Grant #4 start time	[4]
Grant #4 length	[2]
Pad = 0	15/39
Frame check sequence (FCS)	4

(b)

Figure 2.2: Gate control frame.

from all ONUs (see Figure 2.4). This gives the OLT full control and enables it to allocate bandwidth in more fair way. On the other hand, offline schemes poorly utilize bandwidth as there exits large void between each consecutive cycles. This cycle is RTT long, where RTT is the round trip time delay. Hybrid schemes allow some bandwidth requests to be allocated online, while the rest are allocated offline. Interested readers are encouraged to refer to [12, 13] for more information about DBA approaches in literature.

2.2.1 Interleaved Polling With Adaptive Cycle Time (IPACT)

Interleaved Polling With Adaptive Cycle Time (IPACT) [10, 14] is the one of the well known benchmark online allocation schemes. During cycle $n-1$, the OLT receives bandwidth request for next cycle, $R(i,n)$. IPACT applies a limited time grant sizing policy, where the allocated bandwidth grant to ith ONU cannot exceed G_i^{max}. Hence, the allocated

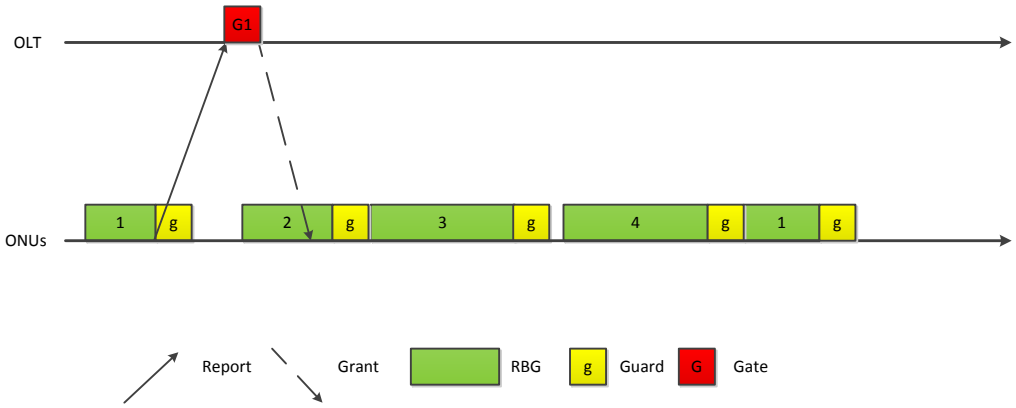


Figure 2.3: Online bandwidth allocation.

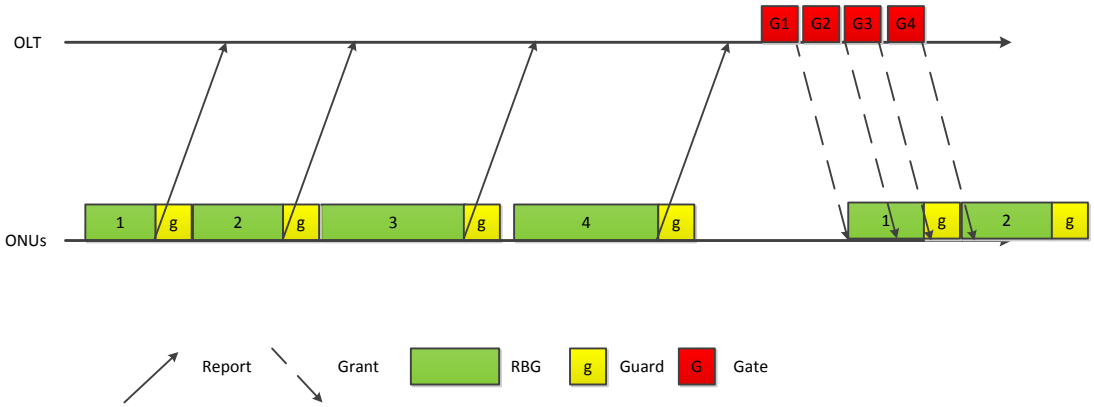


Figure 2.4: Offline bandwidth allocation.

bandwidth grant during cycle n is:

$$G(i,n) = \begin{cases} G_i^{max} & R(i,n) > G_i^{max}, \\ R(i,n) & R(i,n) \leq G_i^{max}, \end{cases} \tag{2.1}$$

2.2.2 Upstream Bandwidth Allocation As A Polling System

Upstream bandwidth allocation in passive optical networks can be modeled as a polling system [15, 14]. In a polling system, shown in Figure 2.5 (presented in [16]), the server (OLT) polls the queues (ONUs) according to some service policy (limited time) [17].

The main parameters of EPON is the offered load (ρ), the round trip time (RTT), the

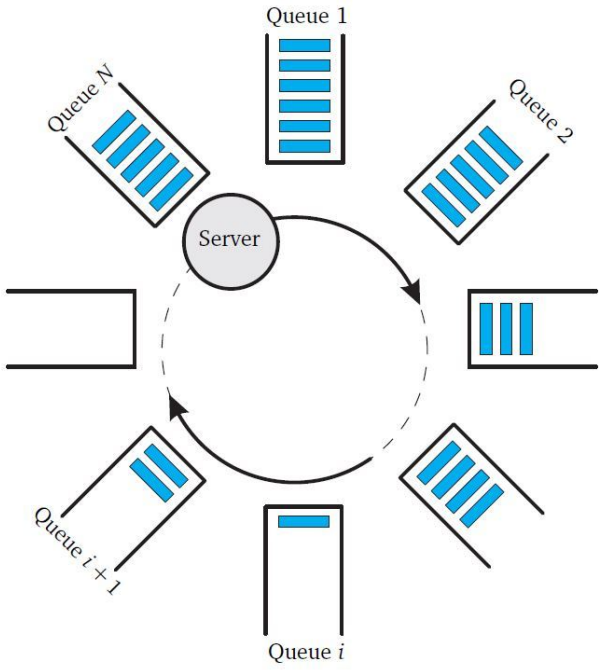


Figure 2.5: Polling system model.

maximum cycle time (T_{cycle}^{max}), and the voids between consecutive grants. This dissertation is about controlling these parameters to further improve network performance.

2.3 Excess Bandwidth Allocation

Excess bandwidth allocation [18, 19, 20, 21, 22] is proposed to reduce the average delay at medium and high load. Excess bandwidth is the unclaimed bandwidth of lightly loaded ONUs ($R(i,n) < G_i^{max}$). This bandwidth can be allocated to heavily loaded ONUs ($R(i,n) > G_i^{max}$). In [18], the authors presented two different excess bandwidth allocation schemes: DBA1, and DBA2. Both of them classifies ONUs into underloaded (**U**) and overloaded (**O**) according to the above mentioned criteria. DBA1 is offline scheme, while DBA2 is hybrid scheme. In DBA1, the OLT computes the cumulative excess bandwidth during cycle n ($E(n)$) as:

$$E(n) = \sum_{i \in \mathbf{U}} G_i^{max} - R(i,n) \tag{2.2}$$

This value is distributed among overloaded ONUs as:

$$E_i(n) = \begin{cases} \frac{w_i}{\sum_{j \in \mathbf{O}} w_j} E(n) & i \in \mathbf{O}, \\ 0 & i \in \mathbf{U}, \end{cases} \tag{2.3}$$

Hence, the bandwidth allocated to each ONU is:

$$G(i,n) = \begin{cases} G_i^{max} + E_i(n) & i \in \mathbf{O}, \\ R(i,n) & i \in \mathbf{U}, \end{cases} \tag{2.4}$$

DBA2 applies the same DBA1 grant sizing approach. However, DBA2 early allocates underloaded ONUs at the horizon time (first free time). Numerical results in [18] show that DBA2 outperforms DBA1 in terms of average packet delay at low and moderate load. This is because DBA2 better utilizes the available bandwidth. At high load, DBA2 suffers from idle time (void) between grants because the ONUs are less likely to be underloaded. Hence, DBA2 will be acting as an offline scheme.

Both DBA1 and DBA2 employ uncontrolled excess allocation. This means that some overloaded ONUs might get more bandwidth than they requested. This negatively impacts the network performance due to unnecessary allocated idle time. In [19], the authors presented a fair weighted excess bandwidth sizing scheme that can be combined with both DBA1 and DBA2. In [23], the authors proposed dividing the ONUs into two subgroups. Each subgroup allocate excess bandwidth in offline manner as DBA1. This technique tends to reduce the impact of idle bandwidth due to offline excess allocation.

In order to solve the problem of idle time (t_{idle}) at high load, Zheng [24] proposed an early allocation scheme (E-DBA2) to remove this undesired effect. The proposed scheme keeps track of last assigned grant end time, t_{end}, and the current time, t. If $t_{end} - t < T_{idle}$, The OLT will schedule one of the overloaded ONUs without assigning excess bandwidth to it. One of the proposed scheme drawbacks is its need of precise timing to remove T_{idle} completely. The scheme might introduce some idle time if the gate message is delayed by

a long Ethernet downstream frame.

In [21], the authors presented three DBA schemes, namely WDBA-1, WDBA-2, and WDBA-3. WDBA-1 and WDBA-2 are the WDM versions of DBA-1 and DBA-2 respectively. The authors also considered three ways to allocate excess bandwidth namely uncontrolled excess (UE), controlled excess (CE), and fair excess (FE). The most efficient algorithm is WDBA-2(FE), which we will refer to as WDBA-2. The performance excel of WDBA-2 comes on the expense of complexity as WDBA-2 has to keep track of all overlapped cycles [21]. Their results show that WDBA-2 outperforms IPACT With A Single Polling Table (IPACT-st) [25] for K = 2, while it achieved the opposite performance for higher values of K. The reasons of such behavior are the idle time wasted waiting for cycle end and what we call the overload delay penalty. The overload delay penalty, which is common for all early allocation approaches, is that overloaded OUNs, with more buffered frames in general, are delayed till cycle end and excess bandwidth can be allocated. This means that many frames in overloaded ONUs will encounter additional delay for the sake of additional frames to be transmitted in this cycle. In [26], the authors modified WDBA-1 by sending the report message in the beginning of the time slot in order to reduce the idle time.

In [22], the author proposed an online excess allocation scheme to allocate excess bandwidth on-the-fly. In this scheme, the OLT maintains an excess bandwidth allocation pool used to both accumulate and distribute excess bandwidth. The excess pool has a maximum value defined by the network operator. Upon receiving bandwidth request from underloaded ONU, the OLT increases the excess pool value by $G_i^{max} - R(i,n)$ taking into consideration the maximum value of the excess pool. When overloaded ONU sends bandwidth request, the OLT assigns on-the-fly its maximum transmission window plus excess bandwidth share not exceeding $w_i E_{pool}$, where E_{pool} is the accumulated excess bandwidth and w_i is the ONU relative weight according to its service level agreement. Upon doing so, the OLT updates the excess allocation pool. Online excess allocation does not incur

idle bandwidth, however it has a fairness problems and does not efficiently utilize excess bandwidth.

2.4 Frame Delineation

Multiple threshold reporting is an efficient technique to eliminate the wasted bandwidth. In [27], a threshold reporting based DBA (TDBA) is proposed. Each ONU can send up to 5 different queue sets within the same report frame. Upon receiving all report frames, the OLT calculates the total excess bandwidth and the cumulative request per threshold from all overloaded ONUs. The OLT starts to find a matching threshold not greater than the available bandwidth in the next cycle for the overloaded ONUs. If the OLT finds a matching threshold, it grants each overloaded ONU the reported amount in that matching threshold, otherwise it equally divides available bandwidth among the overloaded ONUs. The matching threshold concept is not fair since some less overloaded ONUs may receive less grants due to other highly overloaded ONUs. In other words, TDBA treats the granting cycle bandwidth as single pool and tries to get to a compromise between ONUs bandwidth requests without considering the fair bandwidth allocation. Another drawback, which is common between DBA1 and TDBA, is that OLT has to wait for all report frames to run the DBA, hence each granting cycle has idle time of round-trip-time (RTT) plus the DBA calculation time.

Nikolova et al. [28] presented a similar idea to TDBA. In order to avoid the idle time problem, they run the DBA before collecting all report frames. The unreported ONU will not be assigned a bandwidth grant. They changed the ONUs polling order randomly to equally give ONUs the chance to report before the DBA run time. In [29], Kramer et al. presented a solution to the frame delineation problem by using two stage buffer. They are able to remove the wasted bandwidth on the expense of increasing the average delay. They suggested to decrease the maximum cycle time to remedy this problem, although this will

decrease the bandwidth utilization.

2.5 Congestion Resolution in EPONs

There is little research work done on changing the maximum cycle time value. In [30], TADBA is proposed. TADBA sets two values for maximum cycle time: T_{MIN} and T_{MAX}. The value for each cycle is determined based on the total requested bandwidth. If the total requested bandwidth is less than one fourth T_{MAX}, the maximum cycle time is set to T_{MIN}. TADBA treats the cycle bandwidth as single pool and build its decisions based on the total requested bandwidth. This concept does not consider the SLA differences between users. TADBA also does not take into account that the larger requested bandwidth can be because of one highly loaded ONU and switching to T_{MAX} will benefit only this node and leads to unfair bandwidth allocation.

2.6 DBA Schemes for long-reach passive Optical Networks

Recently, Long-reach passive Optical Networks (LR-PONs) has gained more attention in research. One of the goals of next generation PONs is to increase the distance (reach) between the OLT and the ONUs. The reach can be up to 100 Km, this means almost 1 ms of $\overline{RTT}$. Since DBA is run at the OLT, each ONU cannot be polled less than every $\overline{RTT}$. This is very crucial to delay sensitive traffic and leads to performance degradation. In the following, we present a brief of the benchmark DBA schemes for LR-EPONs.

Multi-Thread polling (MTP) [8] (referred to as MTP-off) is suggested to reduce average packet delay. The idea is to run two or more independent **offline** DBA process (threads) to poll ONUs. The numerical results show that MTP outperform offline Single-Thread Polling (STP). MTP-off is more complex than STP as it requires thread tuning, inter-thread coordination, and bandwidth over-grant problems [8, 3]. In [3], the authors deeply investigated offline MTP and showed that IPACT (online DBA) outperform MTP-off. The also

showed that all proposed schemes cannot reduce delay below 1.5 $\overline{RTT}$. In [4], the authors proposed an online MTP algorithm (MTP-on). Their results show that online MTP significantly outperforms offline MTP. They have also show that STP outperforms MTP at T_{cycle}^{max} below 4 ms.

In addition to MTP approach, there are decentralized techniques, optical coding techniques, and request based void filling. In [31, 32], the authors proposed a decentralized scheme where bandwidth is allocated through ONUs coordination. In order to do that, the passive splitter is modified to allow the upstream signal to be reflected in the down stream direction to all ONUs. In [33], the authors proposed that ONUs should use optical coding to transmit frequent Queue Increment Request (QIR). On the other hand, the OLT applies Just-In-Time scheduling approach [34]. QIRs help updating the OLT with the most recent ONUs queue size. The ideal minimum delay bound of this technique is $\overline{RTT}$. On the other hand, optical coding needs highly accurate synchronization between the OLT and ONUs especially for long-reach distance.

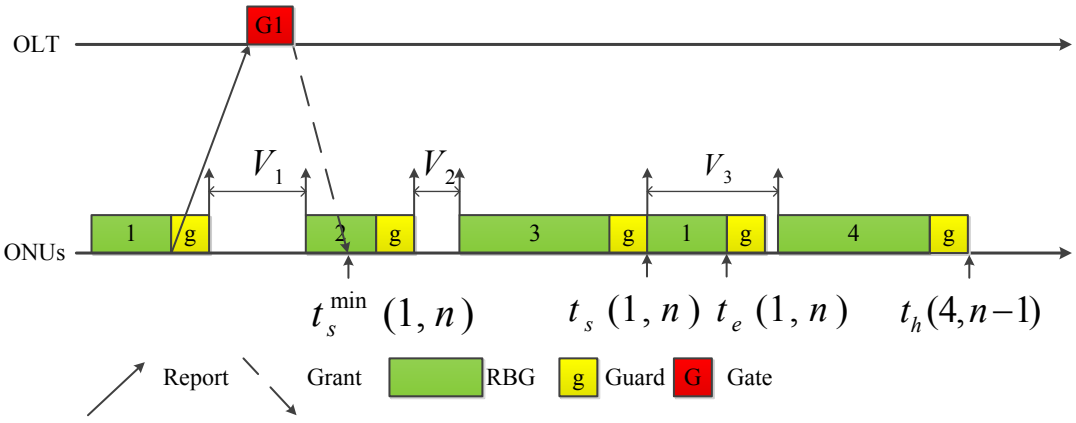


Figure 2.6: Request based void filling.

In [35, 36, 37], a request based void filling (RBVF) is proposed to fit grant requests from relatively nearer ONUs into available voids as shown in Figure 2.6. Upon receiving bandwidth request and employing grant sizing policy to determine $G(i, n)$, the OLT begins to search for eligible void that fits the bandwidth grant. A void V_k is considered usable if it

matches two conditions described as follows:

$$V_e(k,n) - V_s(k,n) \geq G(i,n) \tag{2.5}$$

and

$$V_e(k,n) - G(i,n) \geq t_s^{min}(i,n), \tag{2.6}$$

where $V_e(k,n)$ and $V_s(k,n)$ represent the void start and end times, respectively. If V_k is satisfies the above conditions, the grant start ($t_s(i,n)$) and end times ($t_e(i,n)$) are given by

$$t_s(i,n) = max(t_s^{min}(i,n), V_s(k,n)) \tag{2.7}$$

and

$$t_e(i,n) = t_s(i,n) + G(i,n). \tag{2.8}$$

The selected void is the one that has minimum $t_s(i,n)$ among the set of eligible voids. The computation complexity of RBVF is $O(N)$. If binary search tree is used, the computation complexity is reduced to $O(logN)$ [35]. The results in [35] show that RBVF reduces the average delay compared to NVF. However, it does not reduce the average delay below the 1.5 $\overline{RTT}$ bound. In the best case, each ONU cannot transmit less than every RTT_i. RBVF performance improvement decreases with ONUs with small distance variations. If all ONUs have the same D_i, RBVF cannot reduce the average delay compared to NVF [35].

In [38, 39], authors proposed a request based partial void filling (RBPVF) scheme to further improve RBVF. Their algorithm is a mix between RBVF and Multi-Thread Polling (MTP) [8]. Upon receiving bandwidth request, the OLT invokes RBVF to fit the grant within one of the available voids. If the OLT cannot find suitable void, it invokes RBPVF that can divide $G(i,n)$ into at most P chunks and switch the corresponding ONU to Multi-Thread mode. During Multi-Thread mode, the OLT cannot invoke RBPVF and it can use RBVF only. The OLT starts killing the threads requesting zero bandwidth until there is

only one thread left. The OLT switches this ONU back to single-thread mode again. This method enables long grants that do not fit in a single void to be divided into two or more grants. Their results show a delay improvement compared to Earliest Finishing Time with Void Filling (EFT-VF) [35], but the minimum delay bound can be reached is 1.5 $\overline{RTT}$ for the same reasons listed above.

2.7 Energy Saving in EPONs

In EPON, energy saving can be achieved by putting ONU into sleep mode while watching the network QoS condition. Most of the DBA studies assume that all ONUs can sleep except the one transmitting data to the OLT [40, 1, 41, 42, 43, 44, 45, 46, 47]. The fundamental assumption in all of these studies is that the OLT can buffer the downstream traffic of the sleeping ONU until it returns back to the operational mode. It is important to obtain the optimal sleeping time while considering certain traffic QoS requirements.

In [48], the authors developed a new MAC-Layer protocol that uses fixed bandwidth allocation scheme under low load operation. The OLT goes to the energy saving mode if it finds that the throughput is below certain threshold. In sleeping mode, each ONU wakes-up periodically in predefined periods to send and receive upstream and downstream traffic, respectively. The authors in [49] tackled the energy saving problem in Gigabit-PON (GPON) using the adaptive link rate (ALR) approach. They use a simple mechanism to enable the ONU to adaptively change the optical link rate based on the traffic load to increase the sleep mode period. However, the proposed scheme did not consider any specific traffic QoS requirements.

Two energy saving downstream MAC-layer schemes are proposed in [50]. In the first scheme, which is called downstream centric scheduling (DCS), the downstream traffic for certain ONU is buffered until an upstream time slot is assigned to it. In the second scheme, which is called the upstream centric scheduling (UCS), the ONU switches to active mode

whenever there is an upstream or downstream transmission slots scheduled. Although UCS achieves better energy saving, it causes higher delay when downstream and upstream time slots are not efficiently scheduled. The research study in [51] developed a downstream traffic scheduling protocol with the limited service discipline under the UCS-based green bandwidth allocation framework. The developed scheme investigates the maximum sleep time of each ONU and maximum possible energy saving.

Chapter 3

Excess Bandwidth Distribution for EPONs

3.1 Introduction

As we mentioned in Chapter 2, bandwidth allocation (BA) is considered the most vital task in EPONs functionality. It coordinates between ONUs to transmit their upstream data to the OLT. Bandwidth allocation can be static (SBA) or dynamic (DBA). DBA is more suitable for access networks due to its bursty traffic. The concept of excess bandwidth distribution [18] is introduced to increase bandwidth utilization. The ONUs are divided into underloaded and overloaded according to the reported buffer size ($R(i,n)$). In this chapter, we present an effective yet simple excess bandwidth allocation and scheduling scheme called Delayed Excess Scheduling (DES).

3.2 Delayed Excess Scheduling (DES)

DES main idea is to delay scheduling of excess bandwidth to the next cycle and use the assigned excess bandwidth as increase to G_i^{max} [7]. In other words, the maximum assigned bandwidth during cycle n is $G_i^{max}(n) = G_i^{max} + E_i(n-1)$, where $E_i(n-1)$ is the excess bandwidth share of ONU i from cycle $(n-1)$. DES divides ONUs into three groups:

1. Underload (U): an ONU is considered underloaded if $R(i,n) \leq G_i^{max}$
2. Satisfied (S): an ONU is satisfied if $G_i^{max} < R(i,n) \leq G_i^{max}(n)$
3. Overloaded (O): an ONU is considered overloaded if $R(i,n) > G_i^{max}(n)$

The algorithm works as follows:

- During cycle $(n-1)$, The OLT receives bandwidth requests for cycle n.
- The OLT assigns on the fly bandwidth grant $G(i,n) = min(R(i,n), G_i^{max}(n))$.
- The OLT identify the state of each ONU: **U**, **S**, or **O**.
- After receiving all report messages, the OLT calculates the excess bandwidth share of each overloaded ONU as $E_i(n) = \frac{w_i}{\sum\limits_{k \in O} w_j} \sum\limits_{j \in U} G_i^{max} - R(i,n)$, where w_i is the weight assigned to each ONU according to service level agreement.
- The OLT updates $G_i^{max}(n+1)$ as

$$G_i^{max}(n+1) = \begin{cases} G_i^{max} + E_i(n) & i \in \mathbf{O} \\ G_i^{max} & i \in \mathbf{S} \cup \mathbf{U}. \end{cases} \tag{3.1}$$

It is important to note that DES does not encounter any idle time and does not allocate overloaded ONUs unnecessary bandwidth. DES is very simple and does not require any precise timing procedure and can be implemented easily.

3.3 Delayed Excess Allocation for hybrid TDM/WDM EPONs

Excess bandwidth allocation in hybrid TDM/WDM EPONs (TWDM-EPONs) with dynamic wavelength assignment (DWA) is more complex than in TDM EPONs. This is

because TWDM-EPONs with DWA have no cyclic polling order anymore. In other words, bandwidth allocation cycles overlap. This is why cyclic based excess bandwidth allocation schemes suffer from performance degradation when applied to TWDM-EPONs [21]. The other alternative is to apply an online excess allocation scheme as in [22]. However, online excess allocation has two major drawbacks. First, it does not fully utilize the accumulated excess bandwidth. Second, it does not allocate excess bandwidth fairly among all overloaded ONUs. In the following, we explain our approach to extend DES so that it can be applied to TWDM-EPONs.

Our TWDM approach for excess bandwidth allocation is called Flexible Delayed Excess Scheduling (FDES). FDES has the same ONU classification criteria as DES. Unlike DES, FDES works on segments rather than on cycles. A new segment starts when the OLT receives report message from all ONUs. It is important to note that the OLT might receive more than one report message from some ONUs during one segment. In order to detect a new segment start, the OLT maintains a Boolean check table for all ONUs.

Upon receiving bandwidth request, the OLT does the following:

- The OLT updates the check table entry for the reporting ONU. It also increments the checked ONUs counter if the check table entry was previously false.
- The OLT identifies the state of the reporting ONU: **U**, **S**, or **O**.
- The OLT assigns on the fly bandwidth grant $G(i,n) = min(R(i,n), G_i^{max}(n))$.
- For overloaded and satisfied ONUs, the OLT updates $G_i^{max}(n)$ to be:

$$G_i^{max}(n) = \begin{cases} G_i^{max} & i \in \mathbf{O} \\ G_i^{max}(n) + G_i^{max} - G(i,n) & i \in \mathbf{S}. \end{cases} \tag{3.2}$$

When the checked ONUs counter becomes N (the number of all ONUs), the OLT detects a new segment start and acts as following:

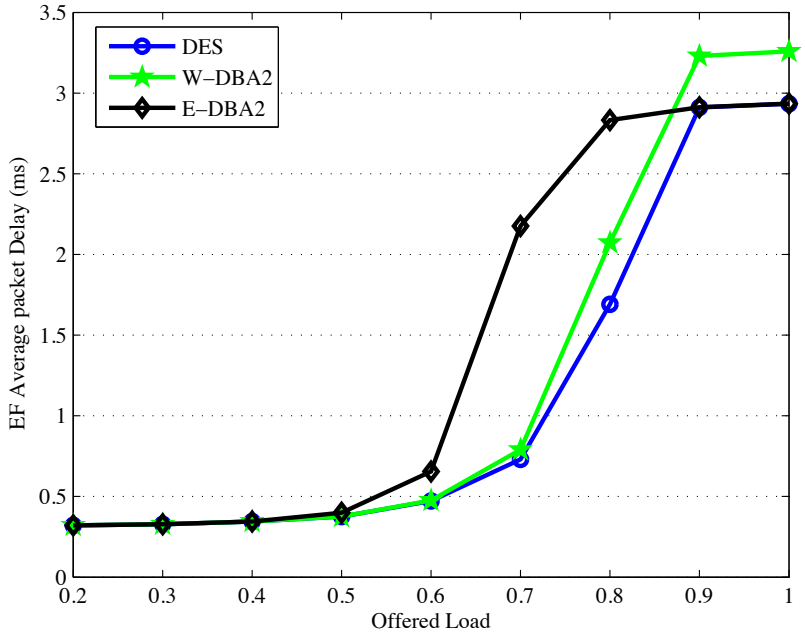


Figure 3.1: EF average packet delay.

- The OLT calculates the excess bandwidth share of each overloaded ONU as $E_i(n) = \frac{w_i}{\sum_{k \in O} w_j} \sum_{j \in U} G_i^{max} - R(i,n)$, where w_i is the weight assigned to each ONU according to service level agreement.

- The OLT updates $G_i^{max}(n+1)$ for the new segment as

$$G_i^{max}(n+1) = \begin{cases} G_i^{max} + E_i(n) & i \in \mathbf{O} \\ G_i^{max} & i \in \mathbf{S} \cup \mathbf{U}. \end{cases} \tag{3.3}$$

For TDM-EPONs and TWDM-EPONs with static wavelength assignment, FDES converges exactly to DES.

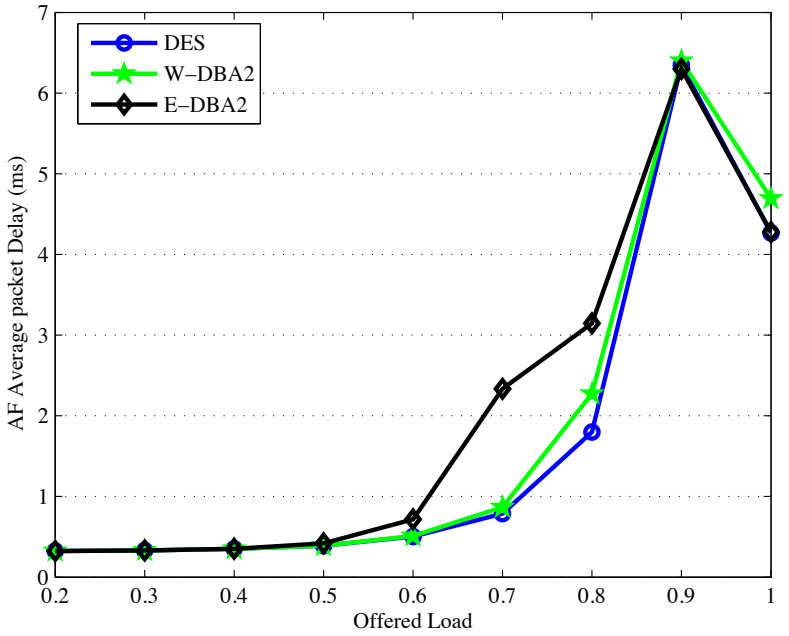


Figure 3.2: AF average packet delay.

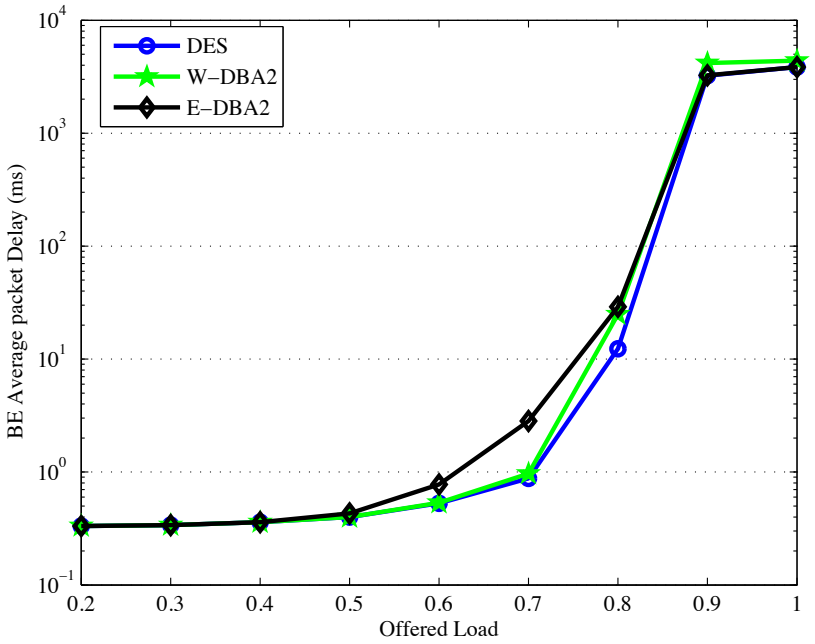


Figure 3.3: BE average packet delay.

3.4 Numerical Results

3.4.1 TDM EPON

We consider an EPON with a single OLT and 16 ONUs with 10 MB buffer size. The upstream and downstream transmission rates are symmetric with 1 Gbps. We use the packet size distribution reported in [11], with a minimum Inter Frame Gap (IFG) of 12 B and preamble of 8 B. The distance between OLT and all ONUs is set to 20 Km, which corresponds to 200 μs $\overline{RTT}$. The generated traffic is DiffServ with three classes of service: Expedited Forward (EF), Assured Forward (AF), and Best Effort (BE). EF ONU offered load share is 20%, while the rest is divided equally between AF and BE. EF is constant-bit-rate (CBR) traffic with Poisson arrivals and fixed packet size of 70 B, while AF and BE traffic are self-similar with long range dependence and hurst parameter of 0.8. G_i^{max} is set to 15000 B and the guard bandwidth is set to 625 B. DES is evaluated against E-DBA2 [24] and a combination of DBA-2 [18] and W-DBA [19] (referred to as W-DBA2). Regarding simulation results, the sample size is 1 million packets per ONU. Consequently, the confidence interval bars are not shown through all the dissertation because it will not be visible compared to the average value.

Figures 3.1, 3.2, and 3.3 show the average packet delay for EF, AF, and BE traffic, respectively. It can be seen that DES shows superior performance against E-DBA2 and W-DBA2. DES reduces the EF average delay by 1.5 ms compared to E-DBA2 at 0.7 load and by 0.4 ms compared to W-DBA2 at 0.8 load. At load beyond 0.9, we notice that DES and E-DBA2 are similar while W-DBA2 is lagging due to idle time problem. The reason behind DES excel is its ability to remove idle time and incorporating controlled excess allocation as well. DES shows similar performance in both AF and BE traffic. For AF, DES reduces the average delay by 1.1 ms and 0.3 ms compared to E-DBA2 and W-DBA2 respectively at 0.8 load. At 0.8 load, DES reduces the average delay by 20 ms for BE traffic.

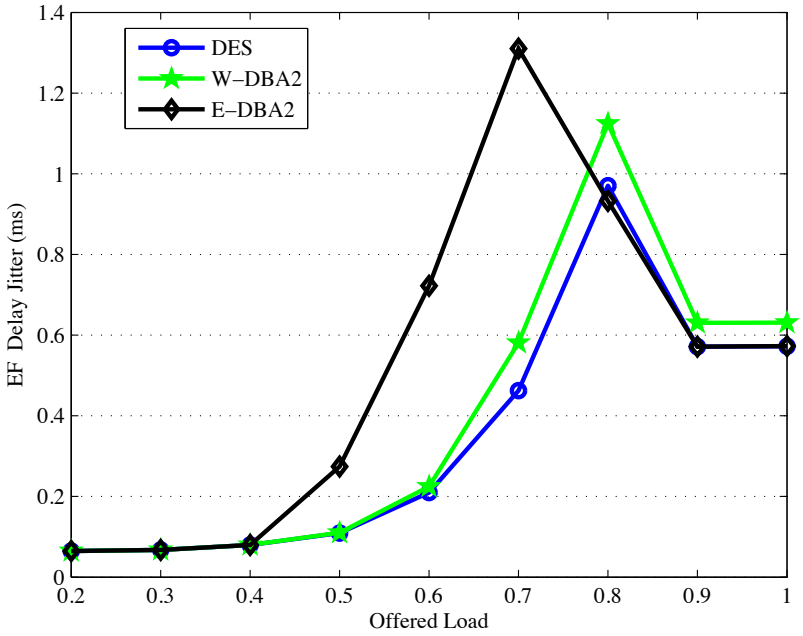


Figure 3.4: EF average delay jitter.

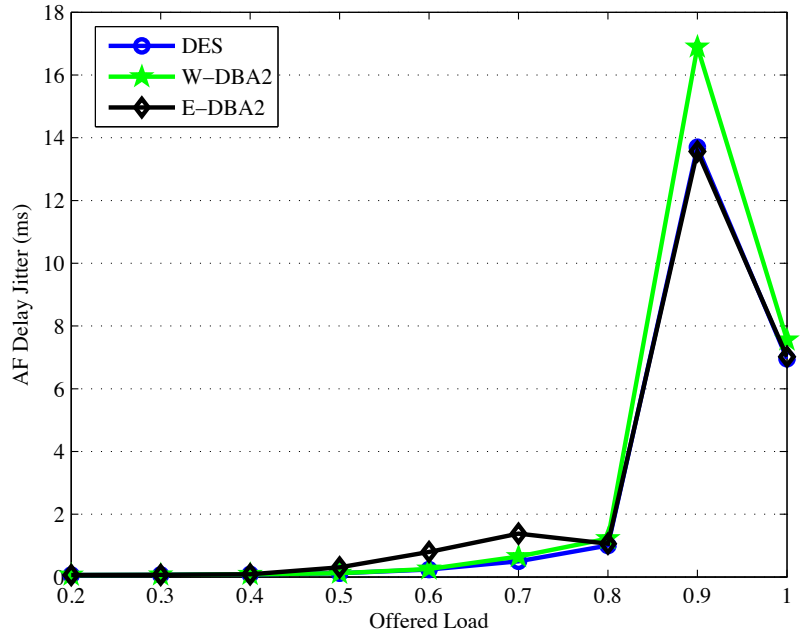


Figure 3.5: AF average delay jitter.

Delay jitter is shown in Figures 3.4, 3.5, and 3.6. For EF traffic, DES jitter reduction ranges from 0.1 ms to 0.8 ms at load 0.5-0.7 compared to E-DBA2. The EF jitter of DES

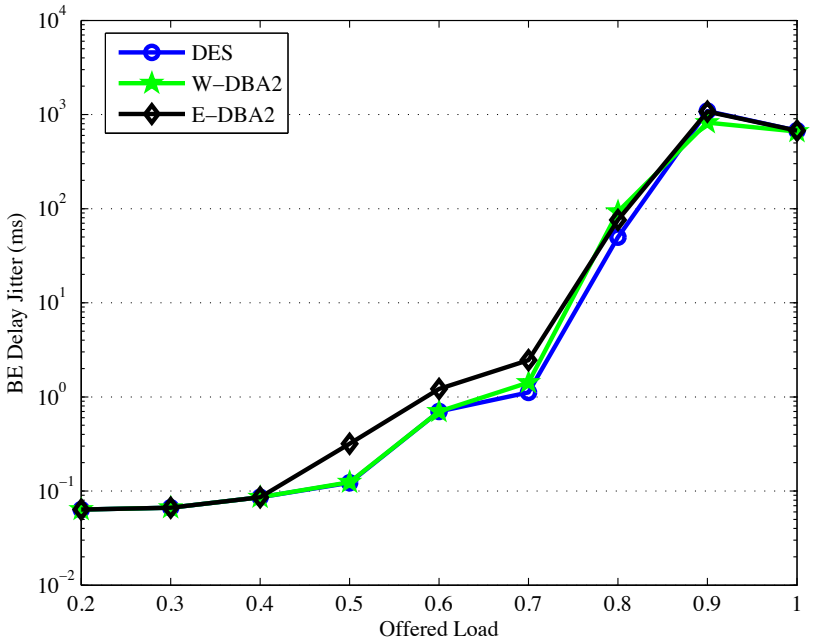
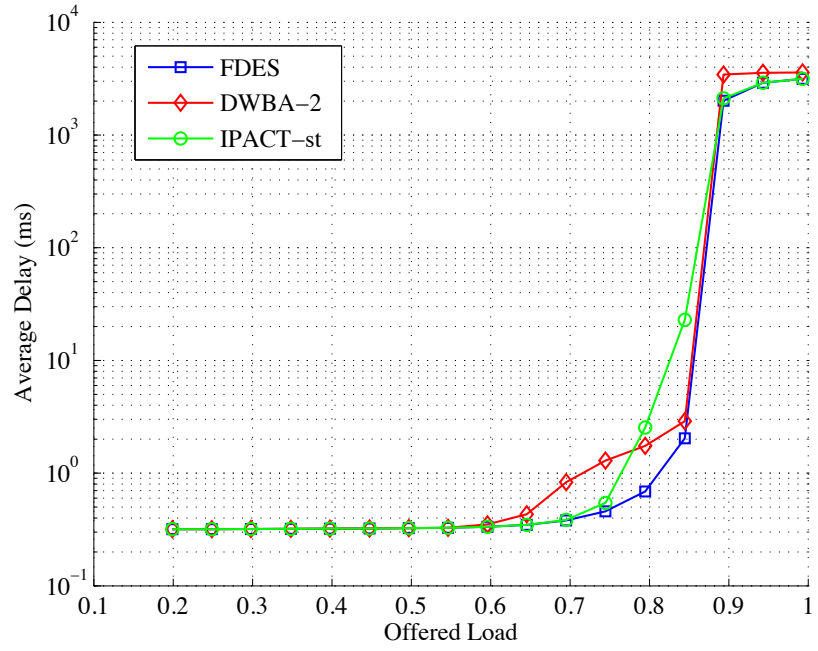


Figure 3.6: BE average delay jitter.

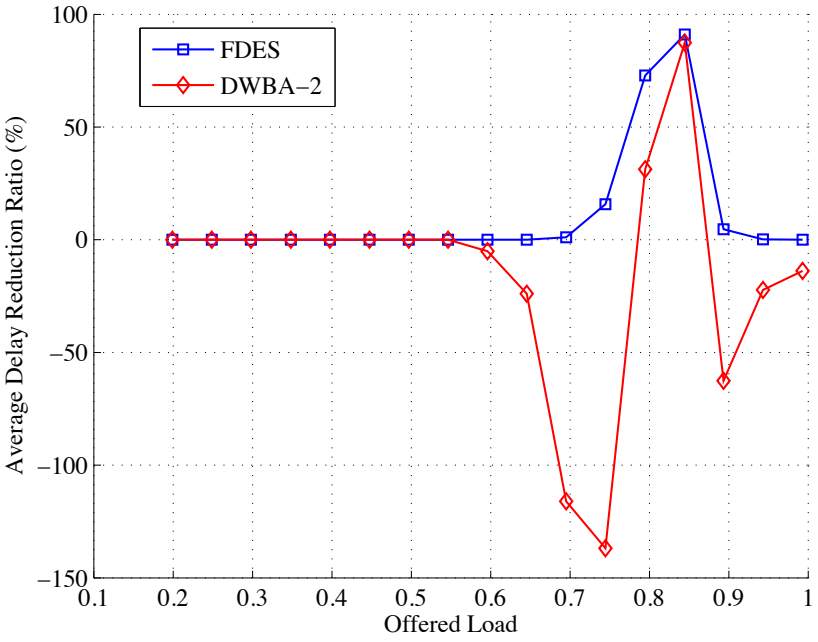
and W-DBA2 are close with advantage to DES. For AF traffic, both DES and W-DBA2 Jitter are lower than E-DBA2 at 0.5-0.8 load. At load 0.8-full load, DES and E-DBA2 are better than W-DBA2. The performance of the three schemes is close in case of BE with relative advantage to DES.

3.4.2 Hybrid TDM/WDM EPON

In this section, we show the simulation results of TWDM-EPONs. We consider a TWDM EPON with single OLT, 128 ONUs and four upstream WDM channels. Each ONU is equipped with fast tunable laser (negligible tuning time) that can be tuned to any of the upstream channels. Incoming traffic is self-similar with long range dependence and Hurst parameter of 0.8. Guard bandwidth is set to 1 μs (125 B). We consider two distance spans scenarios: 20 Km and 100 Km (long-reach). For 20 km span, the maximum cycle time is set to 2 ms. This corresponds to G_i^{max} of 7187 B. For 100 Km span, the maximum cycle time is set to 4 ms. This corresponds to G_i^{max} of 15500 B. FDES is tested against

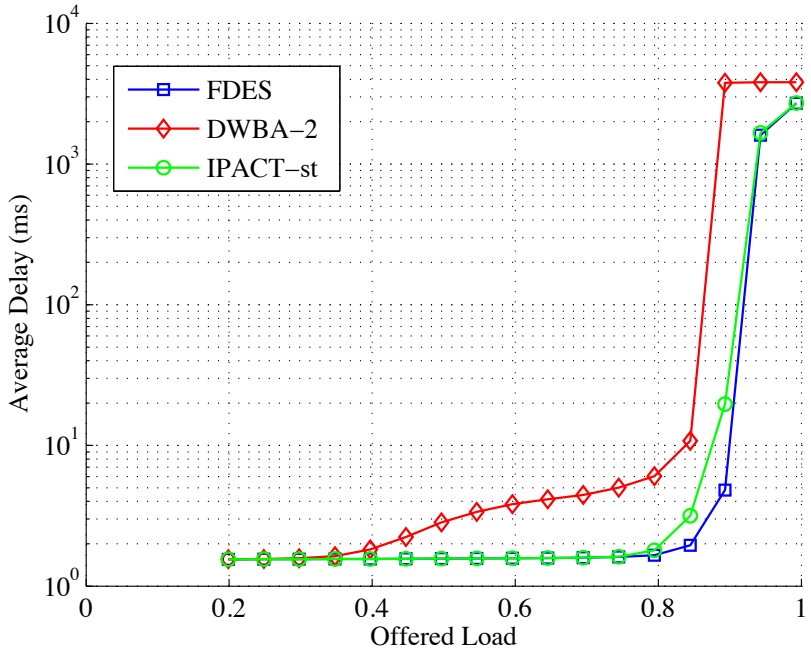


(a)

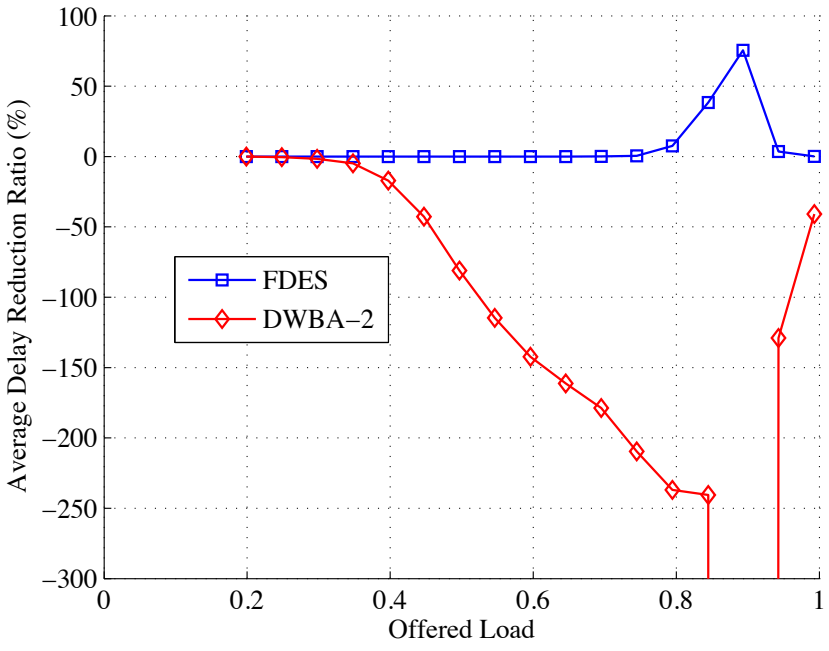


(b)

Figure 3.7: FDES, DWBA-2, and IPACT-st performance comparison for 20 Km span (a) Average delay (b) Delay reduction ratio. (%)



(a)



(b)

Figure 3.8: FDES, DWBA-2, and IPACT-st performance comparison for 100 Km span (a) Average delay (b) Delay reduction ratio. (%)

IPACT-st [25] (WDM version of IPACT) and DWBA-2 [21] (WDM version of W-BDA2).

Figure 3.7 shows the average delay comparison among the tested schemes for 20 Km span. For offered load below 0.6, all schemes have the same average delay. At load above 0.6, DWBA-2 average delay quickly increase with offered load. IPACT-st average delay is less than DWBA-2 till load of 0.78. Between 0.78 and 0.9 load, IPACT-st delay is higher than both DWBA-2 and FDES. FDES Outperforms DWBA-2 at load range between 0.6 to 0.85 and from 0.9 to 1. It also outperforms IPACT-st at load range between 0.7 to 0.9. FDES achieves delay reduction up to 90% compared to IPACT-st. On the other hand, DWBA-2 performance fluctuates between achieving positive and negative delay reduction.

Figure 3.8 shows the average delay comparison between the tested schemes for 100 Km span (long-reach). DWBA-2 has the worst performance. Its delay quickly increases at load of 0.4. This is because ONUs are overloaded most of the time in long-reach case. This makes DWBA-2 works almost as an offline scheme rather than early allocation scheme. FDES is stable compared to both DWBA-2 and IPACT-st. FDES achieves delay reduction up to 75% compared to IPACT-st.

3.5 Summary

In this chapter, we introduced a novel scheme, DES, to effectively distribute upstream excess bandwidth with overloaded ONUs using controlled allocation and without incorporating idle time. DES uses delayed allocation rather than early allocation. DES preserves ONUs polling order and hence reduces delay Jitter. DES achieves delay reduction for EF, AF, and BE traffic. Our proposed scheme can fit easily with industrial ONU deployment due to its simplicity and its small DBA run time.

We also investigated applicability of DES to TWDM EPONs. We presented FDES as a WDM extension to DES. FDES allocates excess bandwidth allocation upon the start of new segment. Numerical results show that FDES achieves less delay compared to IPACT-st and

DWBA-2. It is important to note that both DES and FDES performance converges to IPACT and IPACT-st, respectively at high load. This is because all ONUs are overloaded and excess bandwidth becomes scarce. Next chapter discusses the performance enhancement at high load through resolving frame delineation challenge in TWDM-EPONs.

Chapter 4

Bandwidth Compensation of Frame Delineation Wasted Bandwidth

4.1 Introduction

(TWDM)-EPON is considered one of the candidates of next generation PONs (NG-PONs). TWDM employs K upstream channels to carry the upstream data of N ONUs. Wavelength assignment (WA) can be either static (SWA) or dynamic (DWA) [21]. In this chapter, we present a novel DBA scheme called Efficient Grant Sizing Interleaved Polling (EGSIP). We introduce a novel concept in DBA process called bandwidth compensation. Bandwidth compensation allows ONUs to reuse the wasted bandwidth due to frame delineation [29] in their next cycle(s) and hence achieve the maximum bandwidth efficiency. Bandwidth compensation reduces the average packet delay and packet dropping while increases network throughput.

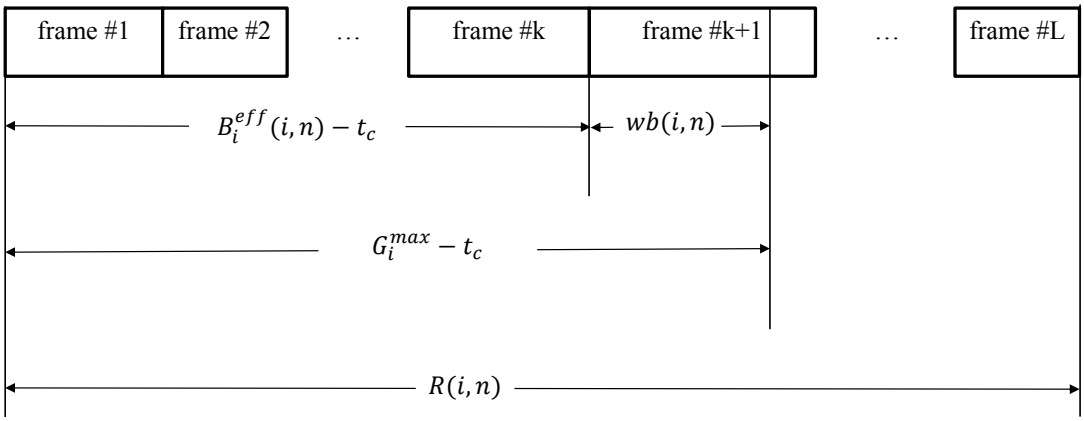


Figure 4.1: Wasted bandwidth.

4.2 Efficient Grant Sizing Interleaved Polling (EGSIP)

4.2.1 EGSIP

In limited time polling disciplines, some of the granted bandwidth is wasted due to frame delineation as shown in Figure 4.1, where $wb(i,n)$ is the wasted bandwidth and t_c is the report frame size including Inter Frame Gap (IFG) and preamble. Considering Ethernet frames, $wb(i,n)$ ranges from 1 to 1537 B. In order to show the effect of $wb(i,n)$ on performance, we define the bandwidth efficiency, η_{eff}, given by:

$$\eta_{eff} = \frac{\sum_{i=1}^{N}(\overline{G_i} - wb(i,n))}{\sum_{i=1}^{N}\overline{G_i} + \frac{N \times t_g \times C}{8}}, \tag{4.1}$$

where $\overline{G_i}$ is the average grant size, C is the transmission rate, and t_g is the guard time between time slots. The maximum bandwidth efficiency can be achieved is:

$$\eta_{eff}^{max} = \frac{T_{cycle}^{max} - N \times t_g}{T_{cycle}^{max}}, \tag{4.2}$$

Only overloaded ONUs suffer from wasted bandwidth, which we refer to as the over-

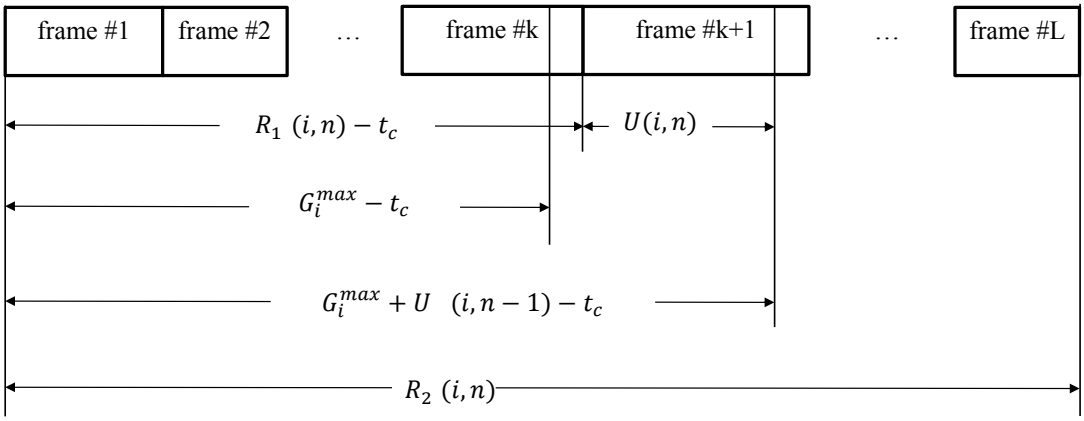


Figure 4.2: Bandwidth compensation.

load penalty. Removing wasted bandwidth reduces packet delay and reduces the polling cycle length, hence, reduces the bandwidth utilization. In [15], the authors developed a mathematical model to calculate the average wasted bandwidth given that the ONU is overloaded to be:

$$E[w/overloaded] = \frac{1}{\overline{X}} \sum_{r=1}^{S^{max}-1} r \times (1 - F_X(r)), \tag{4.3}$$

where $\overline{X}$ and $F_X(x)$ is the average value and cumulative distribution of packet size (bytes), respectively, and S^{max} is the maximum packet size. For the distribution in [11], $\overline{w} \simeq$ equals 598 B. This waste represents 4% of the bandwidth for 16 ONUs with maximum cycle time, T_{cycle}^{max}, of 2 ms and t_g = 5 μs. In general, the average wasted bandwidth in seconds, $\overline{w}$, is:

$$\overline{w} = \frac{p_{overloaded} \times 8}{C \times \overline{X}} \sum_{r=1}^{S^{max}-1} r \times (1 - F_X(r)), \tag{4.4}$$

where $p_{overloaded}$ is the probability that the ONU is overloaded.

EGSIP uses threshold reporting but in more smart way. EGSIP adaptively changes the maximum transmission window, $G_i^{max}(n)$ according to frame delineation existence. The idea of EGSIP is shown in Figure 4.2. EGSIP algorithm uses two thresholds in report frame, $R_1(i,n)$ and $R_2(i,n)$, where n is the granting cycle index. $R_1(i,n)$ cannot exceed $G_i^{max}(n)$, where $U_i(n-1)$ is the reusable bandwidth credit left form cycle $n-1$. $R_2(i,n)$

is used to report the whole ONU buffer size. The threshold values include the size of the report frame. $G_i^{max}(n+1)$ is updated as:

$$G_i^{max}(n+1) = \begin{cases} G_i^{max}, & R_2(i,n) \leq G_i^{max} \\ G_i^{max} + G_i^{max}(n) - R_1(i,n). & otherwise \end{cases} \tag{4.5}$$

A similar process is done on the OLT side to update $G_i^{max}(n+1)$. The OLT allocates the bandwidth according to:

$$G(i,n) = \begin{cases} R_1(i,n), & R_1(i,n) \leq G_i^{max}(n) \\ G_i^{max}(n). & otherwise \end{cases} \tag{4.6}$$

It is worth to note that no malicious/malfunction ONU can be granted more than its basic maximum transmission window. If the ONU is working properly, the allocated bandwidth will be always $R_1(i,n)$. The compensated bandwidth, $U(i,n) = G_i^{max}(n) - R_1(i,n)$, cannot exceed the length of the largest frame including IFG and preamble, which corresponds to $U(i,n) \leq 1537$ B in case of Ethernet traffic.

In order to estimate $R_1(i,n)$, ONUs need to know the minimum guaranteed bandwidth. This can be done via two methods. The first method is through the extended versions of register or gate frames [52] using part of the padding. The other one is what we call it the exploring report method. The idea is to use the first report message after ONU registration to know the basic G_i^{max}. The ONU will send a report message containing with R_{1i} and R_{2i} equal to the maximum ONU buffer capacity. As a result, the ONU will reply with a gate message containing the maximum permissible grant. The ONU with use this value as G_i^{max} and start applying EGSIP.

The computation complexity of $R_1(i,n)$ depends on the ONU state. If ONU is underloaded, the computation complexity is $O(1)$. This is because in that case $R_1(i,n)$, $R_2(i,n)$, and buffer size are identical. If ONU is overloaded, The ONU begins sequentially

to calculate $R_1(i,n)$ by summing the buffered packet sizes as shown in Figure 4.2. The computation complexity in that case will be $O(K_{max})$, where K_{max} is

$$K_{max} = \frac{G_i^{max} + 1537}{t_c}. \tag{4.7}$$

K_{max} represents the maximum number of frames to be transmitting during G_i^{max}, while t_c is the time needed to transmit the smallest Ethernet frame (control frame).

4.2.2 Stability Conditions

Stability is an important factor in EPONs. Stability is measured as the maximum offered load the network can handle. Without loss of generality, we consider TWDM-EPONs with SWA. In general, the maximum offered load, ρ^{max}, can be measured as:

$$\rho^{max} = \frac{T_{cycle}^{max} - T_{idle}}{T_{cycle}^{max}}, \tag{4.8}$$

where T_{idle} is the average cycle idle time. We can apply this concept to derive the stability conditions for IPACT and EGSIP as follows:

$$\rho_{IPACT}^{max} = \frac{T_{cycle}^{max} - N \times t_g - N \times \overline{w}_{IPACT}}{T_{cycle}^{max}}, \tag{4.9}$$

$$\rho_{EGSIP}^{max} = \frac{T_{cycle}^{max} - N \times t_g}{T_{cycle}^{max}}. \tag{4.10}$$

From (4.9) and (4.10), it can be noticed that:

$$\rho_{EGSIP}^{max} > \rho_{IPACT}^{max}. \tag{4.11}$$

The last inequality proves that bandwidth compensation is more stable than IPACT. Furthermore, EGSIP is the only one achieves the maximum permissible offered load (*e.g.,* the

maximum achievable bandwidth).

To the best of our knowledge, all of EPON mathematical modeling studies ignores the effect of frame delineation [53, 54, 55, 56, 57]. In [53], an approximate model for IPACT is proposed. The authors did not consider the effect of the wasted bandwidth on the delay. We modify their model and include the effect of $\overline{w}$. The modified formula is:

$$\overline{T}_{IPACT} \approx \frac{\lambda\overline{X^2}}{2(1-\rho(T_{cycle}^{max}-Nt_g-N\overline{w}_{IPACT})/T_{cycle}^{max})} + \frac{(3N-\rho-\frac{2\rho(N-\rho)(Nt_g+N\overline{w}_{IPACT})}{(1-\rho)(T_{cycle}^{max}-Nt_g-N\overline{w}_{IPACT})})}{2(1-\rho(T_{cycle}^{max}-Nt_g-N\overline{w}_{IPACT})/T_{cycle}^{max})}(t_g+\overline{w}_{IPACT}), \tag{4.12}$$

where $\overline{T}_{IPACT}$ is the average packet queuing delay for IPACT. The average packet queuing delay, $\overline{T}_{EGSIP}$, is given by:

$$\overline{T}_{EGSIP} \approx \frac{\lambda\overline{X^2}}{2(1-\rho(T_{cycle}^{max}-Nt_g)/T_{cycle}^{max})} + \frac{(3N-\rho-\frac{2\rho(N-\rho)Nt_g}{(1-\rho)(T_{cycle}^{max}-Nt_g)})}{2(1-\rho(T_{cycle}^{max}-Nt_g)/T_{cycle}^{max})}t_g. \tag{4.13}$$

Comparing (4.12) and (4.13), we observe that $\overline{T}_{EGSIP} \leq \overline{T}_{IPACT}$ for all values of ρ. The delay reduction ratio mainly depends on the offered load. It also depends on the packet size distribution, guard time, maximum cycle time, and the number of ONUs.

4.3 Simulation and Numerical Results

4.3.1 Simulation Model

We consider an EPON with a single OLT and 128 ONUs with 10 MB buffer size. Ethernet frames size is between 64 to 1518 B, we use the packet size distribution reported in [11], with a minimum IFG of 12 B and preamble of 8 B. The distance between OLT and all ONUs is set to 20 Km, which corresponds to 200 μs RTT. The offered traffic is self-similar with

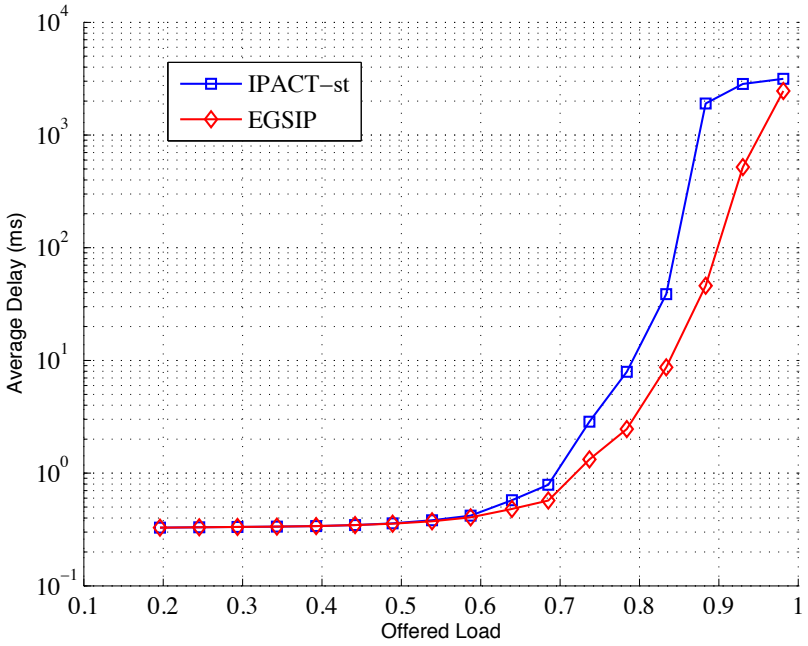
Hurst parameter 0.8 and long range dependence (LRD). The offered traffic is distributed uniformly over all ONUs. We use guard period of 125 B between consecutive slots. There are K upstream channels with transmission rate 1 Gbps, where K is set to 4. G_i^{max} is set to 7687 B. We use both SWA and DWA. For DWA, we assume that all ONUs are equipped with fast tunable lasers in order of few microseconds. EGSIP is evaluated against IPACT-st. It worth to note that in case of SWA, IPACT-st acts as IPACT. The main performance measures are average delay, blocking probability, and bandwidth efficiency.

4.3.2 Numerical Results

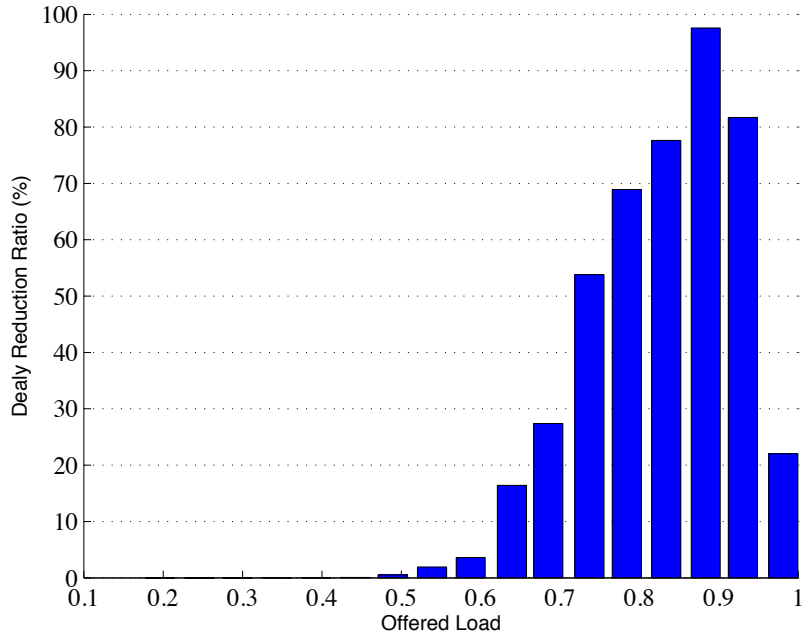
Figure 4.3 shows performance comparison between IPACT-st and EGSIP in case of SWA. Figure 4.3(a) shows the average delay in ms. At load below 0.5, both IPACT-st and EGSIP are almost identical. At load above 0.5, EGSIP outperforms IPACT-st. In terms of delay reduction, EGSIP reduces IPACT-st delay by 16% at 0.65 load, 70% at 0.75 load, and 98% at 0.9 load. This reduction proves the EGSIP robustness against IPACT-st derived in (4.10).

Packet dropping probability is shown in Figure 4.4(a). EGSIP shows substantial low dropping rate compared to IPACT-st at high load. This is mainly because EGSIP ability to eliminate idle time and compensate it in the next grant(s). This would help faster packet transmission, reducing buffer size, and reducing dropping rate. The last performance measure is bandwidth efficiency shown in Figure 4.4(b). EGSIP is more efficient than IPACT-st at high load. It also achieve bandwidth efficiency close to η_{eff}^{max}. It is very important to clarify that EGSIP does not outperform IPACT-st at low load because the problem tackled by EGSIP (frame delineation) becomes more apparent at high load.

Figure 4.5 shows performance comparison between IPACT-st and EGSIP in case of DWA. EGSIP reduces IPACT-st delay by 10% at 0.73, 60% at 0.8, and 98% at 0.9 load. It can be noticed that IPACT-st delay in DWA is less than its peer in SWA. This is because the wavelength assignment gain in DWA. ONUs are polled faster in DWA rather than in SWA. EGSIP still outperforms IPACT-st in terms of dropping rate and bandwidth efficiency (see



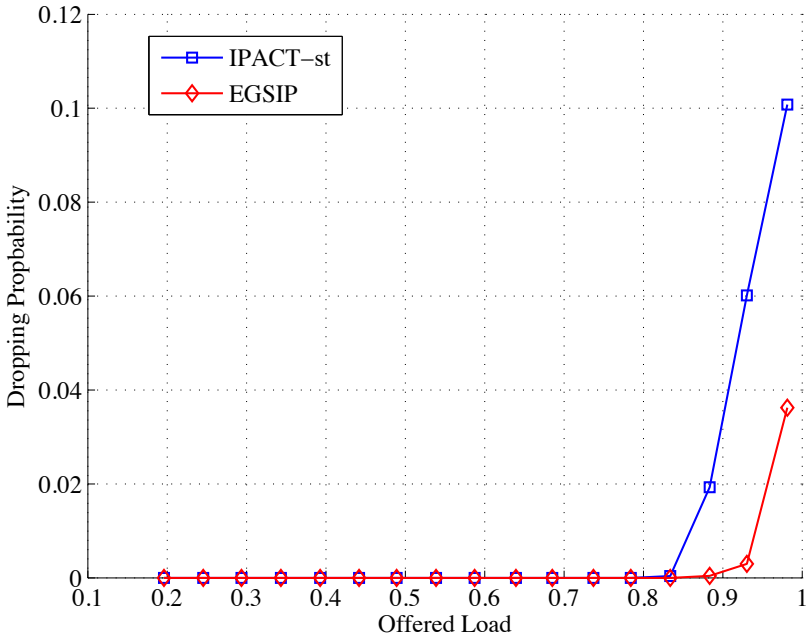
(a)



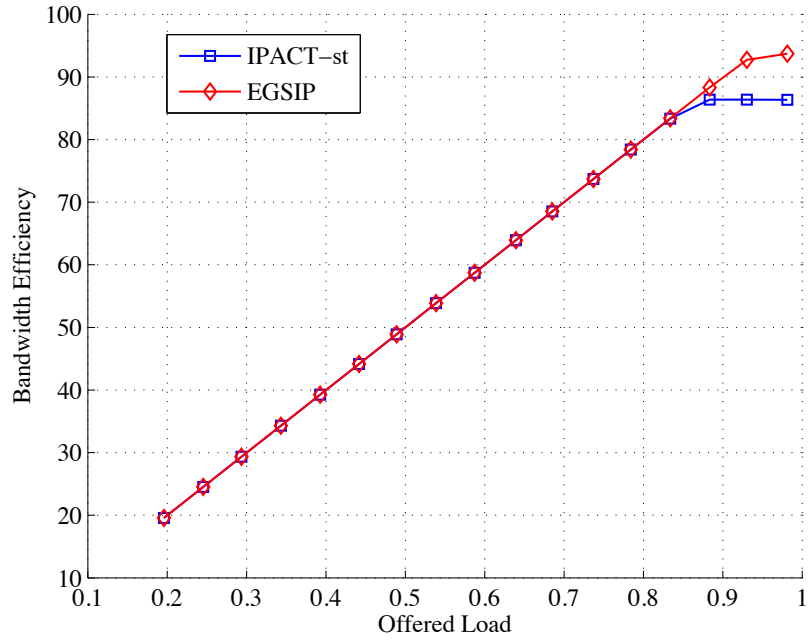
(b)

Figure 4.3: SWA, K=4: (a) Average delay (b) Average delay reduction.

Figure 4.6).

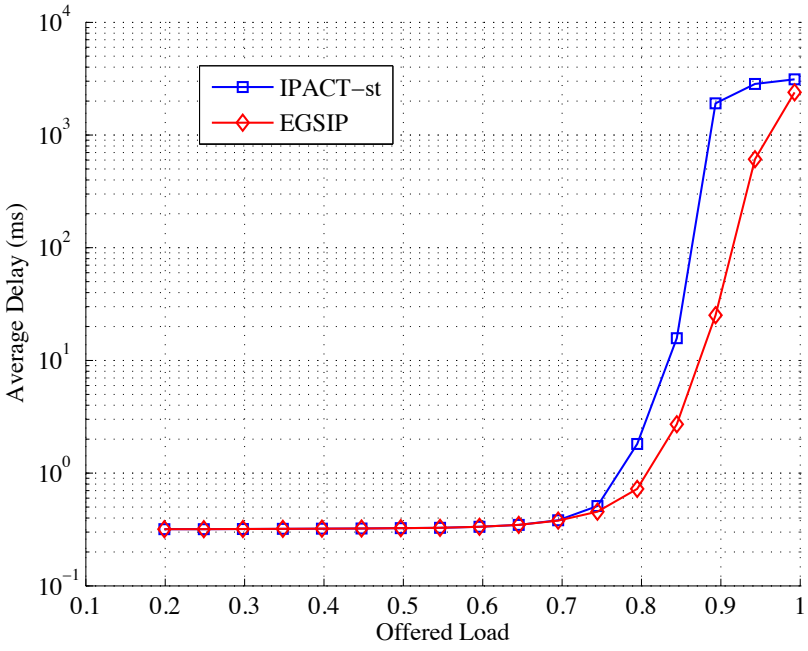


(a)

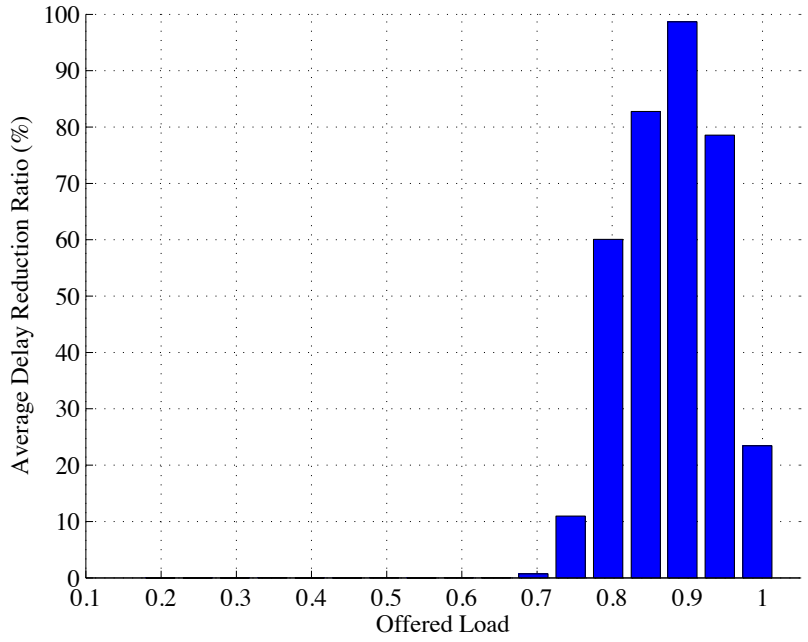


(b)

Figure 4.4: SWA, K=4: (a) Dropping probability (b) Bandwidth efficiency.



(a)

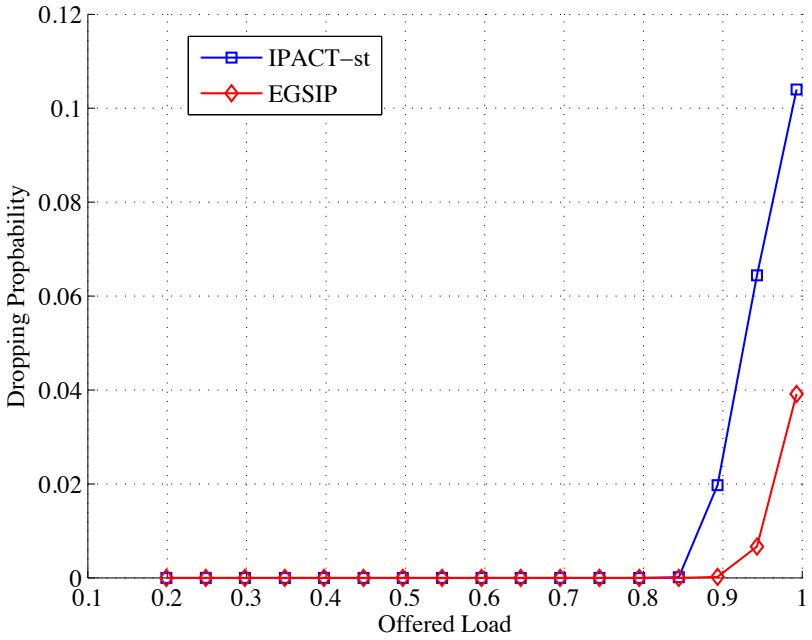


(b)

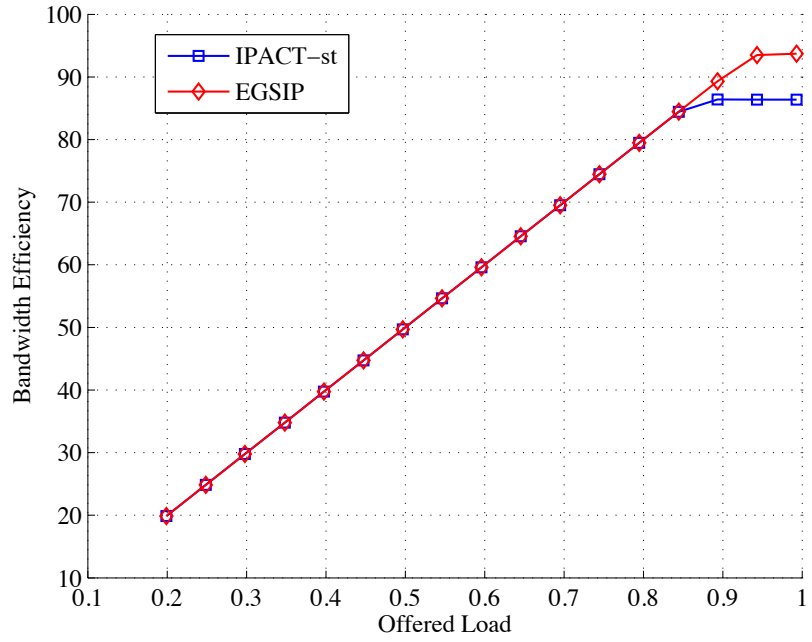
Figure 4.5: DWA, K=4: (a) Average delay (b) Average delay reduction.

4.4 Summary

In this chapter, we introduced a novel concept in DBA which is bandwidth compensation. We also presented a new DBA scheme called EGSIP which improves the TWDM EOPN



(a)



(b)

Figure 4.6: DWA, K=4: (a) Dropping probability (b) Bandwidth efficiency.

performance. EGSIP resolves the frame delineation problem and guarantees that each overloaded ONUs can obtain their maximum grant size. We proved via mathematical analysis

that EGSIP is more stable than IPACT-st. This stability has a large impact on delay and buffer size reduction. The main reason behind EGSIP superior performance is its capability of utilizing the available bandwidth without loss as in IPACT-st and without reduction as in threshold reporting. EGSIP outperforms IPACT-st at load higher than 0.5 and 0.6 for SWA and DWA respectively. EGSIP computation complexity is $O(1)$ in case of underloaded ONU. In case of overloaded ONU, the computation complexity is $O(K_{max})$, where K_{max} is the maximum number of packets can be transmitted during a bandwidth grant. EGSIP is more stable than IPACT-st and can be used at load peaks or sudden network congestion.

Chapter 5

Congestion Detection and Resolution

5.1 Introduction

In the two previous chapters, we investigated both excess bandwidth allocation and frame delineation in EPONs. However, in both chapters, the maximum cycle time is assumed to be set to fixed value defined by network operators. Most of the proposed DBA schemes so far apply time-limited polling [10]. Time-limited polling sets a maximum value for cycle time and ONU grant size to avoid bandwidth monopoly from high loaded ONUs. Increasing the maximum cycle time improves bandwidth utilization especially in peak-hours time but some packets will experience longer delays due to longer cycle times [10]. On the other hand, decreasing the maximum cycle time helps to reduce the delay sensitive traffic (real time) average delay because it allows ONUs to transmit more frequently. Many of the proposed DBA schemes choose the maximum cycle value to be 2 ms [10, 18, 20, 19] in their simulation experiments. There is not much research on varying the value of the maximum cycle time.

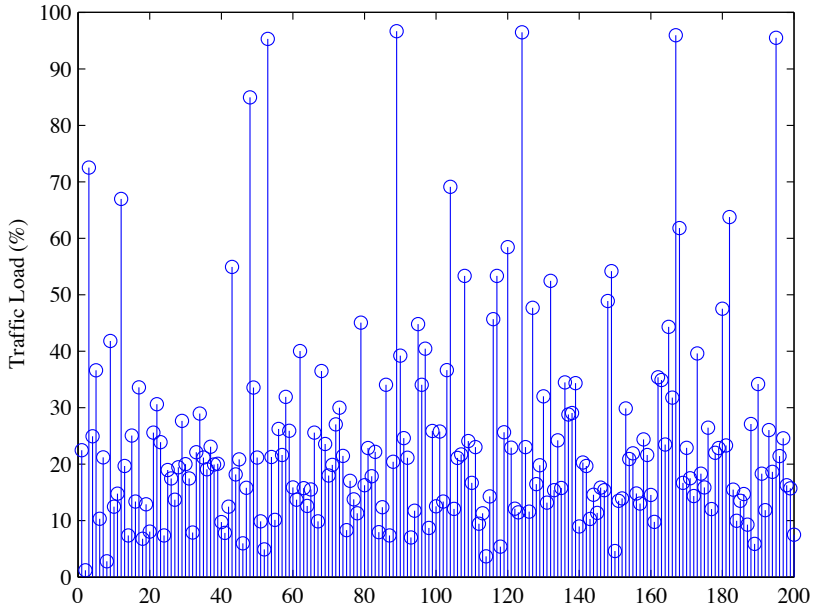
In this chapter, we analyze in-depth the effect of dynamically changing the maximum cycle time value with network load status on the main performance metrics: average delay, throughput, dropping, and fairness. Moreover, we present our new DBA algorithm called Congestion Aware Limited Time (CALT). CALT is a smart framework that adapts the maximum cycle time duration with the incoming traffic load to achieve a compromise between

increasing bandwidth utilization and reducing packet delay from one side and keeping accepted fairness level form the other side.

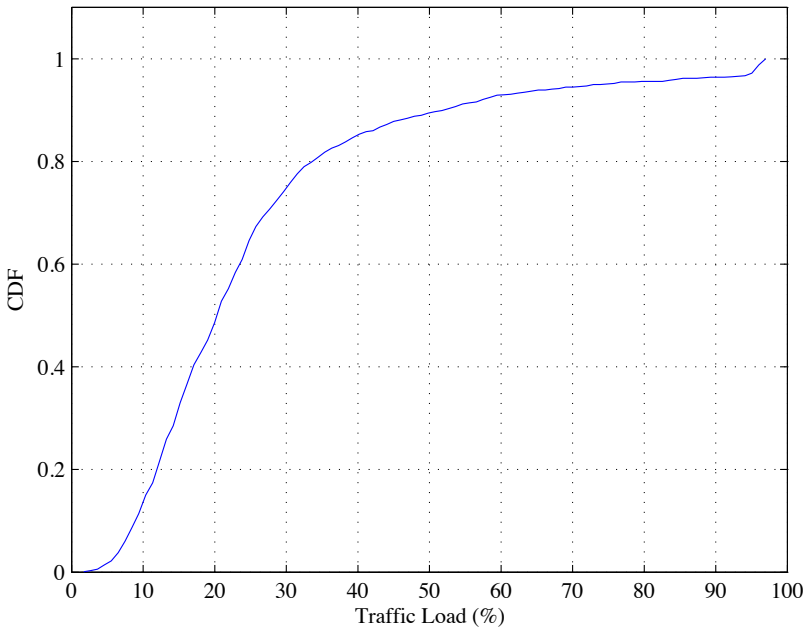
5.2 Congestion Aware Limited Time

In polling systems, the polling disciplines with longer cycle time have the least average queueing delay [58]. Exhaustive service and gated service are the best known disciplines in terms of queueing delay and they are the biggest in terms of polling cycle time. Both of them are not suitable for bandwidth allocation as they suffer from bandwidth monopoly. In [53], the authors presented a mathematical model for IPACT. They derived an approximate formula for the queueing delay for limited service under incoming Poisson traffic. The model shows that increasing T_{cycle}^{max} reduces the queueing delay.

Another reason to consider T_{cycle}^{max} is the bursty and self-similarity nature of access network traffic [59]. Figure 5.1 shows the measured traffic load of generated self-similar traffic with long range dependence (LRD) of 0.25 load sampled over a scale of 1 ms. Figure 5.1(a) shows that the measured traffic load values range from zero to full load. Although the generated traffic load is 0.25, the CDF distribution shows that the traffic load is more than 0.25 almost 40% of time and greater than 0.75 almost 10% of time. It also shows that 10% of the time, the traffic load is less than 0.08. The reason behind such behavior is that frames arrive in longer bursts with off periods between them. This behavior highlights the importance of dynamically changing T_{cycle}^{max} to cope with the long data bursts and also the off periods to ensure better utilization and accepted fairness level. In [60], the authors evaluated the effect of changing the maximum cycle time on EPON performance. Their results show good improvement in both delay and bandwidth utilization especially in medium and higher load which makes increasing T_{max} is a good option during peak hours. We conclude from the above discussion that DBA should dynamically tune its parameters with the short term load variations (Figure 5.1 and long term variations such as peak hours load. Our



(a)



(b)

Figure 5.1: Self-similar traffic load over a scale of 1 ms.

proposed scheme, CALT. is a smart dynamic framework that adaptively controls the maximum cycle time duration based on the detected offered load condition. CALT goal is to

decrease average packet delay and increase the bandwidth allocation fairness and utilization. In CALT, the OLT maintains a set of L levels (values)$\{T_1, T_2, ..., T_L\}$ of T_{cycle}^{max}, where $T_i < T_j \; \forall \; i < j$. CALT uses the number of overloaded ONUs, N_O, as a heuristic to move between the different levels. N_O is considered an indicator to network congestion. The OLT keeps a matrix of threshold values of moving upward and downward, **Q**, as:

$$\mathbf{Q} = \begin{bmatrix} 0 & Q_U^1 \\ Q_D^2 & Q_U^2 \\ . & . \\ . & . \\ . & . \\ Q_D^L & N \end{bmatrix}. \tag{5.1}$$

Q_D^l and Q_U^l represents the downward and upward threshold values of level l respectively. Increasing Q_U^l reduces the probability of incrementing the maximum cycle time. The difference between Q_U^l and Q_D^l along with the traffic load determines how often the OLT uses this level. Network operators determines the best value of **Q** according to the acceptable congestion level. CALT works as follows:

step 1: the OLT receives bandwidth requests and assigns bandwidth to each ONU according to (2.1) until all ONUs report frames are received.

step 2: the OLT counts the number of overloaded ONUs, N_O.

step 3: if $N_O > Q_U^l$ then the OLT increases the current maximum cycle level by 1 to be $l{+}1$. If $N_O < Q_D^l$ then the OLT decreases the current maximum cycle level to $l{-}1$.

step 4: the OLT recalculates G_i^{max} according to new T_{cycle}^{max}.

step 5: go to step 1.

CALT operates in either waste (W) or threshold reporting (T) modes. In W mode, some of the granted bandwidth to overloaded ONUs will be wasted due to frame delineation. In T mode, ONU takes advantage of reporting multiple thresholds to OLT. The OLT sends

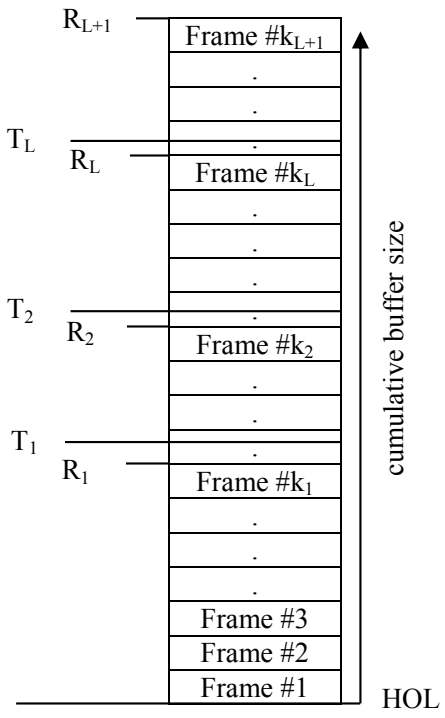


Figure 5.2: CALT-T.

the threshold values, representing G_i^{max} in all L levels, to ONUs using the pad segment of the Register frame [5, 6]. It is worth to note that T mode cannot be used for more than 12 levels because of the size limitations of report frame. The idea of threshold reporting is shown in Figure 5.2, where T_i and $R(i, n)$ are the threshold and reported values of level l respectively. The last report value, R_{l+1} represents the whole ONU buffer size.

5.3 Numerical Results

5.3.1 System Model

We consider an EPON with single OLT and 16 ONUs with 10 MB upstream buffer size. The upstream and downstream transmission rates are symmetric with 1Gb/s. In IEEE 802.3 standard, Ethernet frames size is between 64 to 1518 B, we use the packet size distribution reported in [11], with minimum inter-frame-gab (IFG) of 12 B and preamble of 8 B. The

distance between OLT and all ONUs is set to 20 Km, which corresponds to 200 μs round trip time (RTT). The generated traffic is self-similar traffic with Hurst parameter 0.8 and long range dependence (LRD). t_g is set to 5 μs. T_{cycle}^{max} possible values are {1,2,3,4} ms. The threshold matrix, **Q**, is set to be:

$$\mathbf{Q} = \begin{bmatrix} 0 & 3 \\ 3 & 6 \\ 3 & 6 \\ 3 & 6 \end{bmatrix}. \tag{5.2}$$

We assume that 1 ms is a suitable value for T_{cycle}^{max} in case of light load, while 4 ms is suitable for T_{cycle}^{max} in case of severe load. The lowest level, 1 ms, upward threshold is set to 3, as this level is meant to be used during off periods. The upper levels thresholds are set to 3 and 6 respectively. We assume that the up to 6 overloaded ONUs out of 16 ONUs is an accepted congestion level. We also think that the OLT should move to the lower level if there are less than 3 overloaded ONUs. The offered load is distributed uniformly over ONUs. Table 5.1 summarize the simulation parameters.

Table 5.1: Simulation Parameters.

N	16
C	1 Gbps
ONU upstream buffer size	10 MB
T_{cycle}^{max}	{1,2,3,4} ms
t_g	5 μs
IFG	12 B
RTT	200 μs

CALT is evaluated against IPACT [10] and a combination of DBA2 [18] and W-DBA [19]. In our results, we label the two later schemes as W-DBA2, which has the capability of on-line granting to underloaded ONUs and hence reduce upstream idle time. The maximum cycle time for both IPACT and W-DBA2 is set to 2 ms.

5.3.2 Numerical Results

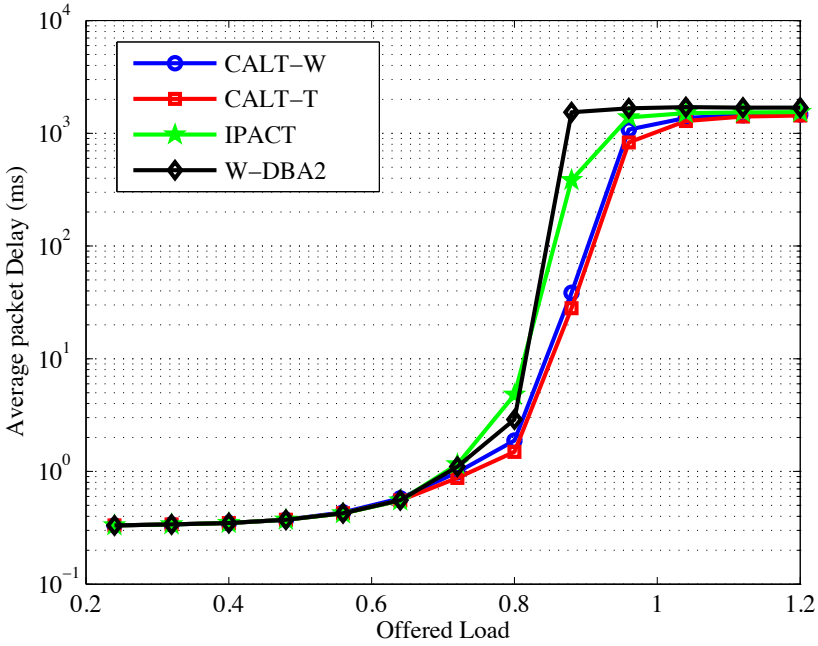


Figure 5.3: Average packet delay.

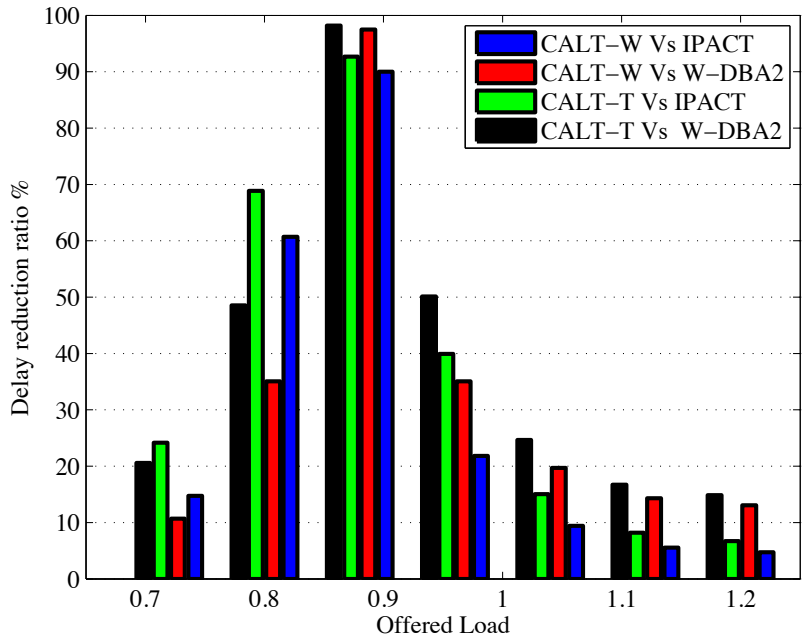


Figure 5.4: Delay reduction ratio.

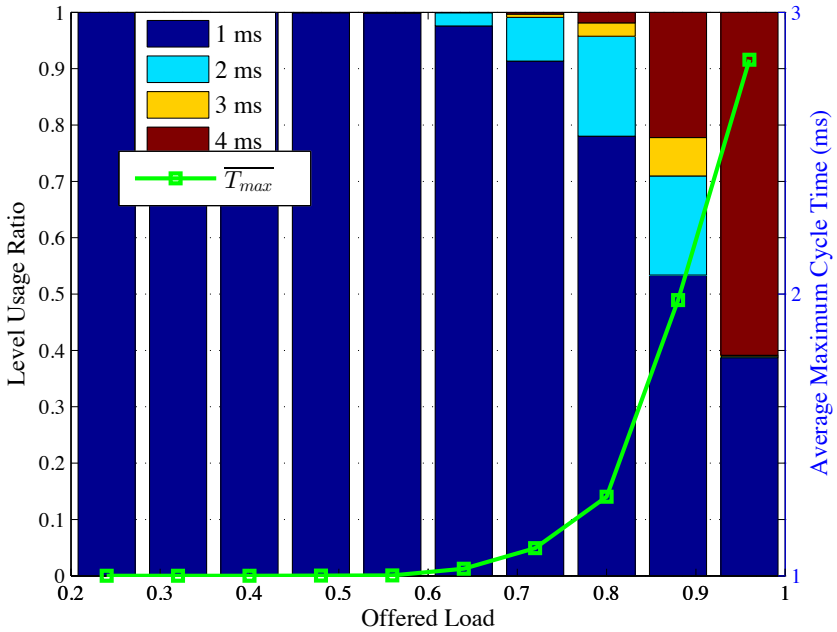
One of the most important performance metrics in EPON is the average packet delay. Figure 5.3 shows the average packet delay of CALT, IPACT, and W-DBA2 respectively. It is clear that CALT has a great impact on reducing delay at medium and higher load (0.64-1.2). The Delay reduction ratio is shown in Figure 5.4. At 0.88 offered load CALT-T shows delay reduction up to 97% compared to W-DBA2 and 92% compared to IPACT, while CALT-W shows also delay reduction up to 95% compared to W-DBA2 and 90% compared to IPACT. At offered load beyond 1 (sever congestion), both modes of CALT reduces delay by ranges from 5-17%.

Figure 5.5 shows level usage ratio and the corresponding average maximum cycle time, $\overline{T_{cycle}^{max}}$, for both CALT-W and CALT-T. It can be seen that CALT uses the lowest level 78-100% of the time at load below 0.9. The most important observation about this figure is that CALT-W reduces the average delay by 60% at 0.80 offered load using 1 ms as T_{cycle}^{max} almost 78% of time. It can be seen also that $\overline{T_{cycle}^{max}}$ is greater than 2 ms only at almost full load condition. CALT-T tends to use the low value levels more than CALT-W due to its increased stability.

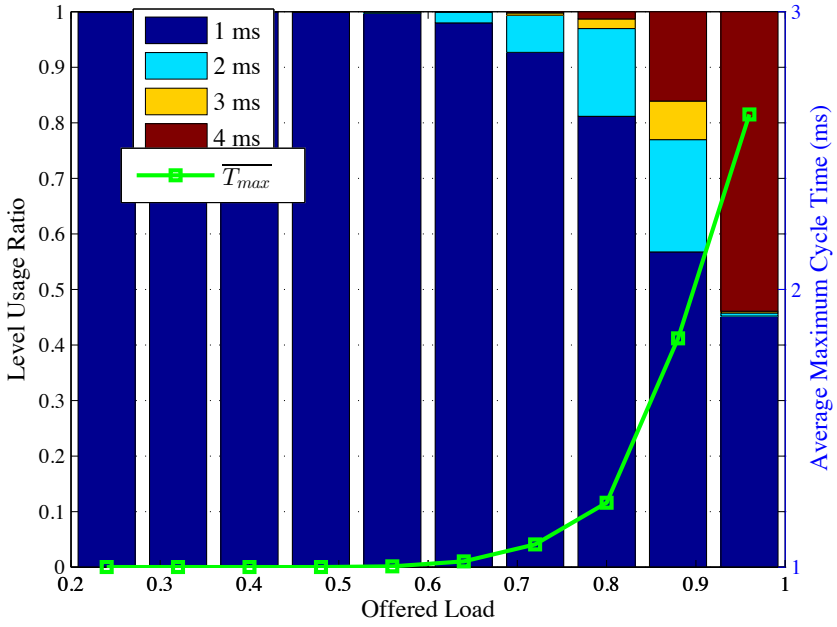
Figure 5.6 shows a comparison between the average cycle time between the different four schemes. At load range from 0.8-0.9, CALT average cycle time is less than both IPACT and W-DBA2. On the other hand, the situation is reversed at load greater than 0.9. The reason behind that is that CALT uses the higher level more frequently at higher load as shown previously in Figure 5.5.

The average ONU buffer size is shown in Figure 5.7. It turns out that CALT stability is much higher than both W-DBA2 and IPACT. CALT is more stable because it adapts instantaneously to traffic load conditions and can serve more packets within the cycle time. CALT buffer size saturates at 1.04 offered load, while W-DBA2 saturates at 0.88.

Delay Jitter is shown in Figure 5.8. Although CALT cycle time is more dynamic due to multiple T_{cycle}^{max} usage, CALT is able to reduce delay jitter compared to IPACT and W-DBA2 at medium and high load. This shows that CALT is a better candidate for delay sensitive



(a)



(b)

Figure 5.5: Level Usage Ratio And Average Maximum Cycle: (a) W (b) T.

applications and differentiated services EPONs. CALT delay jitter is more than IPACT and W-DBA2 at load higher than 0.9. The reason of this behavior is that both IPACT and

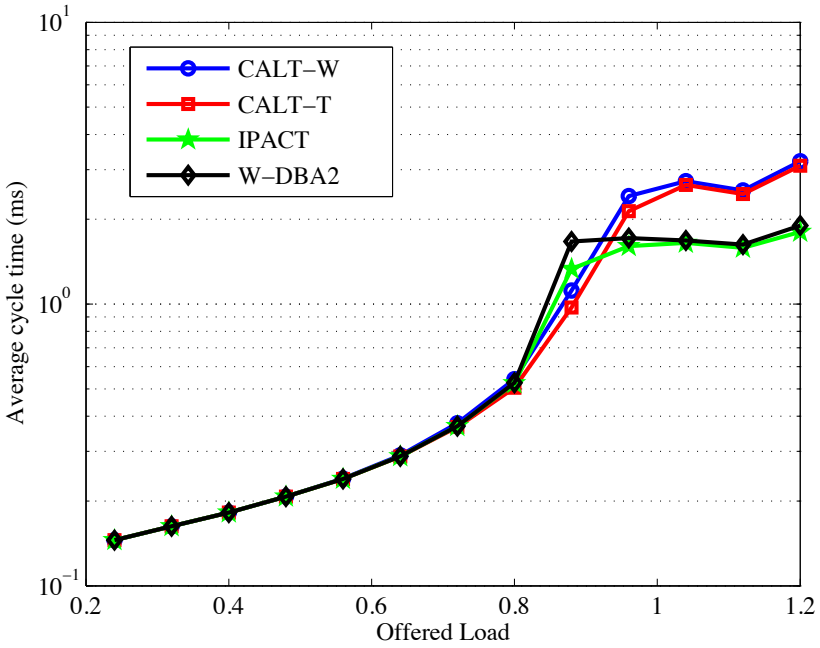


Figure 5.6: Average cycle time.

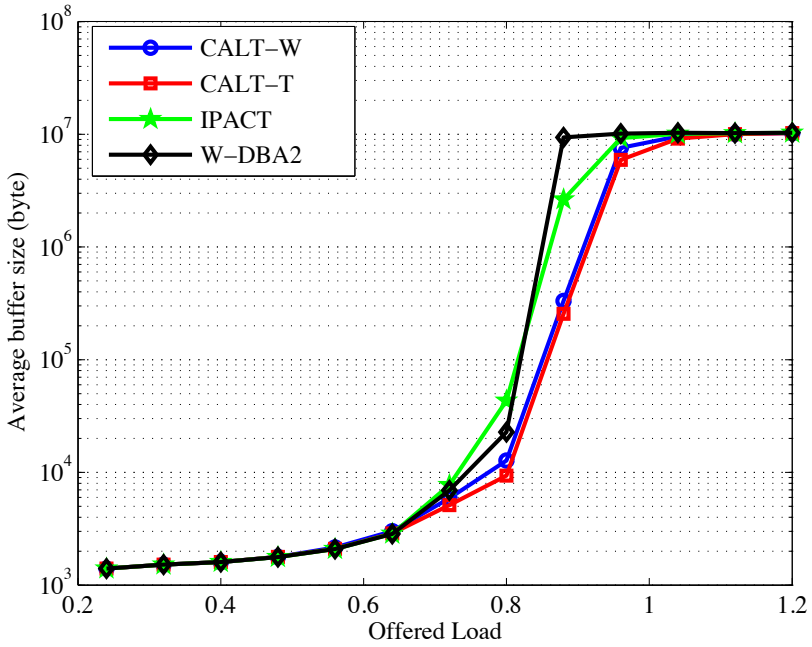


Figure 5.7: Average buffer size.

W-DBA2 saturate before CALT. The almost fixed buffer size causes delay jitter reduction. CALT jitter tends to decrease again at 1.1 load. Figure 5.9 shows another aspect of CALT

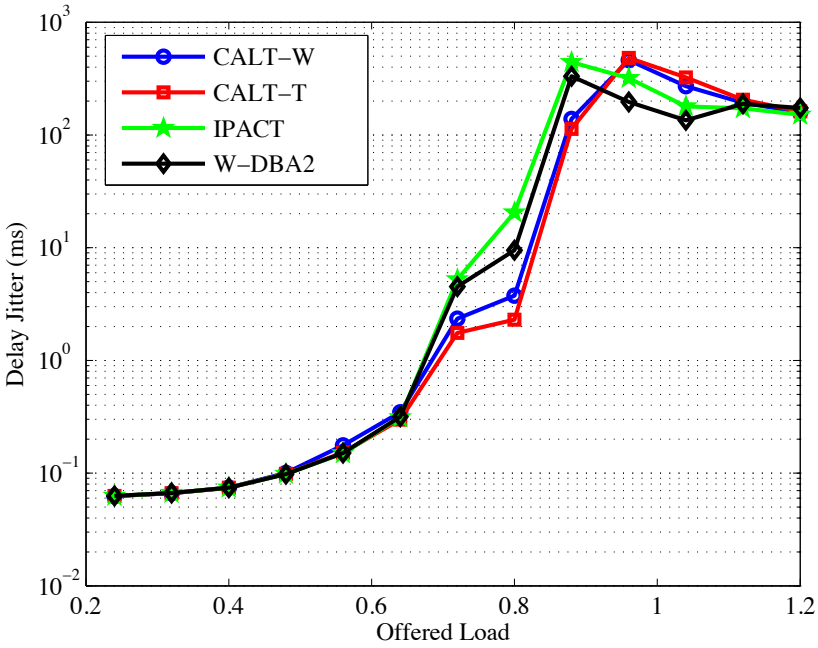


Figure 5.8: Delay jitter

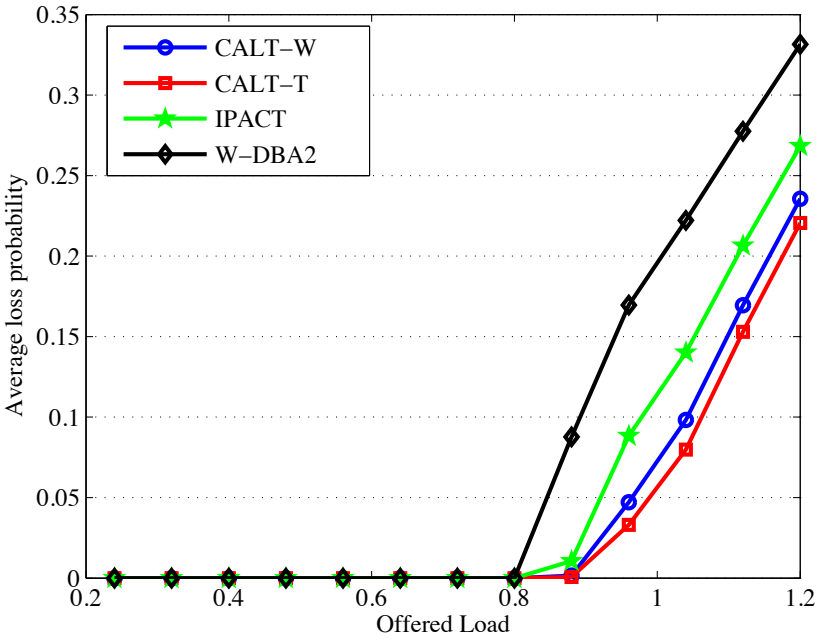


Figure 5.9: Packet loss probability.

excel performance. CALT improved the loss probability by 50% compared to IPACT and 70% compared to W-DBA2 at nearly full offered load. This improvement is a result of

CALT delay, cycle time, and buffer size reduction compared to other schemes.

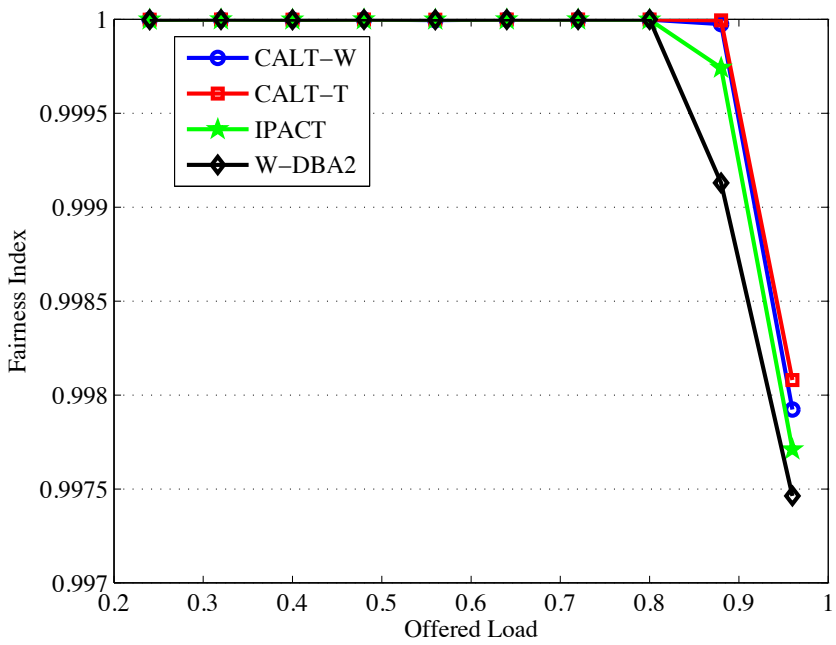


Figure 5.10: Fairness index.

One of the important bandwidth allocation performance metrics is the fairness index. We use the Raj Jain fairness index [61]. Figure 5.10 shows enhanced fairness index for CALT. We observe that CALT allocates bandwidth in a more fair way than IPACT and W-DBA2. This also proves that using N_O as a heuristic to increase/decrease maximum cycle time does not contradict with the fairness principal.

5.4 Summary

This chapter presented a new DBA scheme, called CALT, for EPONs. CALT incorporates interleaved polling along with offered load controlled maximum cycle time. CALT uses the number of overloaded ONUs, N_O as a heuristic to increase/decrease the maximum cycle time value. CALT uses a set of levels (values) the maximum cycle time and both upper and lower threshold for N_O associated with each level. Numerical results show superior improvement in packet delay, dropping probability, bandwidth utilization and buffer size.

CALT also reduces the delay jitter which is one of the important metrics in Diffserv EPONs. CALT is able to fairly allocate bandwidth more than IPACT and W-DBA2.

Chapter 6

Stable and Adaptive Multi-Thread Polling for Long-Reach EPONs

6.1 Introduction

The presented DBA schemes so far discuss challenges related to high offered load. Starting from this chapter, we focus on resolving challenges at low and medium load for Long-Reach Ethernet Passive Optical Networks (LR-EPONs). DBA in LR-EPONs is considered an interesting challenge due to the long propagation delay between both the OLT and the ONUs. The next PON generation should consider extending the reach of Passive Optical Networks in order to reduce the number of needed Central Offices (COs) [62]. Longer reach implies longer round trip time (RTT) delays. This leads to imposing a relatively large minimum average delay bound as we will explain shortly. This chapter and Chapter 7 investigate two different approaches for delay reduction in LR-EPONs, namely Multi-Thread Polling (MTP) and on-the-fly void filling.

The average delay in EPON, $\overline{W}$, is,

$$\overline{W} = \overline{W_{poll}} + \overline{W_{grant}} + \overline{W_{queue}}. \tag{6.1}$$

where $\overline{W_{poll}}$ is the time between packet arrival and the next report message, $\overline{W_{grant}}$ is the

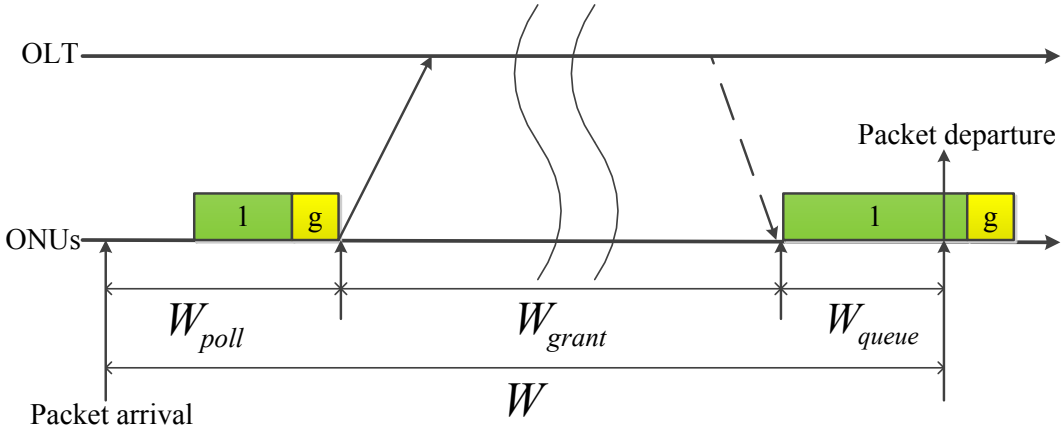


Figure 6.1: Delay composition in EPON.

time between sending report message and receiving corresponding allocated grant via gate message, and $\overline{W_{queue}}$ is the waiting time before transmission during the allocated grant (see Figure 6.1). W_{grant} cannot be less than RTT_{max} (the maximum round-trip-time for all ONUs). The relation between W_{poll} and W_{grant} is

$$\overline{W_{poll}} = \frac{\overline{W_{grant}}}{2}. \tag{6.2}$$

Consequently, the average delay $\overline{W}$ cannot be less than 1.5 RTT_{max}.

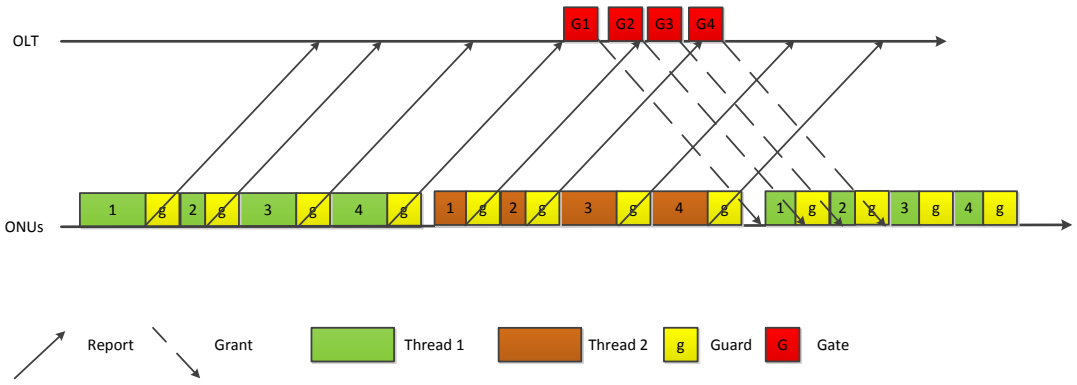


Figure 6.2: Offline MTP.

Offline Multi-Thread Polling (referred to as MTP-off) [8] and online Multi-Thread Polling (referred to as MTP-on) [4] are proposed to reduce W_{poll}. The basic idea of both

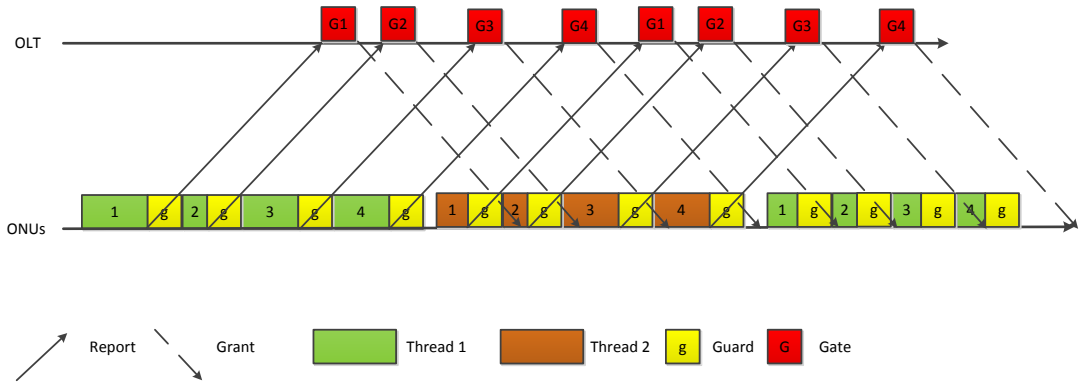


Figure 6.3: Online MTP.

MTP-off (see Figure 6.2) and MTP-on (see Figure 6.3) is to run multiple instances (threads) of certain DBA scheme. The relationship between W_{poll}, W_{grant}, and number of threads (θ) is,

$$\overline{W_{poll}} = \frac{\overline{W_{grant}}}{(2\Theta)}. \tag{6.3}$$

Combining both (6.1) and (6.3), we obtain the minimum boundary for the average delay as,

$$\overline{W} \geq (1 + \frac{1}{(2\Theta)})RTT_{max}. \tag{6.4}$$

This chapter investigates two major associated problems with Multi-Thread Polling, namely bandwidth over-granting and performance degradation at high load. Bandwidth over-granting happens due to that ONUs report their buffered data before they receive bandwidth allocation for the previous thread request. This leads to the fact that some buffered data might be reported multiple times. Consequently, large bandwidth grants without real associated buffered packets are allocated. This effect is useful at low load as the new incoming packets can be transmitted immediately. However, at higher load, this leads to quick performance degradation as reported in [38]. The other problem is that MTP performance quickly degrades at higher load due to the increased report messages and guard bandwidth overhead [3, 4].

6.2 Bandwidth Over-Granting Related Work

Bandwidth over-granting problem happens due to the lack of inter-thread coordination. The consecutive threads have partial overlap in their buffered data report. This happens because of the ONU has to send report message in thread θ prior receiving the grant message of the report sent for thread θ-1.

In [63, 64], authors proposed an approach called newly arrived frames plus (NA+). In their approach, the OLT keeps a record for the last two received bandwidth requests. The OLT subtracts the newer request from the old one to extract the newly arrived frames size ($I(i, \theta, n)$). Upon knowing $I(i, \theta, n)$, the OLT calculates the actual bandwidth request of the reporting ONU, $R(i, n)$, as

$$R(i, n) = I(i, \theta, n) + C(i), \tag{6.5}$$

where $C(i)$ is the compensated bandwidth for backlog traffic. The backlog traffic happens when the actual bandwidth request is greater than the maximum ONU transmission window per thread and due to the fact that Ethernet frames cannot be fragmented. NA+ performance is highly affected by the term $C(i)$ [63]. Choosing large values for $C(i)$ leads to over-granting. On the other hand, small value of $C(i)$ leads to new arriving frames blocking [63]. In order to determine $C(i)$, the authors proposed that the OLT keeps monitoring the upstream channel for measuring the unused slot reminder (USR) of each ONU and use it as a value of $C(i)$ [63]. In [4], authors proposed an online MTP (MTP-on) which leverages the advantage of online over offline polling. Their results show that MTP-on with NA+ cannot achieve the minimum bound derived in (7.2). Similar results are also reported in [3].

In [65], the authors proposed another solution similar to NA+ called Bandwidth Reporting (BR). In BR, ONUs report only the size of the new arrivals and also calculate the compensation term. Upon receiving bandwidth request, $R(i, \theta, n)$, the OLT increments the

total requested bandwidth, $RV(i)$ as

$$RV(i) = RV(i) + R(i, \theta, n), \tag{6.6}$$

where $R(i, \theta, n)$ equals

$$R(i, \theta, n) = I(i, \theta, n) + C(i). \tag{6.7}$$

ONU calculates $C(i)$ as the difference between the granted bandwidth and the actual transmitted bandwidth. BR is quite similar to NA+. However, it moves the burden of $C(i)$ and $I(i, \theta, n)$ calculation to ONU. Numerical results in [65], show that BR outperforms NA+.

6.3 Interactive Bandwidth Allocation (IBA)

In order to resolve in bandwidth over-granting challenge, we present a novel technique called Interactive Bandwidth Allocation (IBA). IBA can be integrated with both MTP-off and MTP-on. Without loss of generality, we focus on integration of IBA with MTP-on because MTP-on outperforms MTP-off in terms of average packet delay [4].

IBA relies on interaction between the OLT and the ONUs. In other words, if the ONU knows its maximum transmission window during particular cycle and thread, it can adjust its reported buffer size to that value and marks these packets as reported in the buffer. This is because these packets are guaranteed to be transmitted during the next grant. In CO side, the OLT transmits two threshold values to each ONU through the padding part in the gate control frame. The first threshold, $Th_1(i, \theta, n)$ represents the maximum transmission window for that ONU during that thread and cycle. The second one, $Th_2(i, \theta, n)$ is always 0XFFFF and represents the maximum value can a report frame carry for buffer size reporting. The second threshold is an indicator of overloaded ONUs and is used for excess bandwidth allocation purpose. It can be noted that IBA uses EGSIP (see Chapter 4)

with the feature of excess bandwidth allocation. For IBA, we choose to employ DES (see Chapter 3) as the excess bandwidth allocation scheme, however, it can support any other scheme. The buffer size reporting process works as follows:

- Upon receiving gate message, the ONU saves the value of both $Th_1(i, \theta, n)$, and $Th_2(i, \theta, n)$ to be used when the bandwidth grant actually starts.

- When it is time for ONU to issue report frame, the ONU uses the saved thresholds to issue two bandwidth report values:$R_1(i, \theta, n)$ and $R_2(i, \theta, n)$ within the same report frame. $R_1(i, \theta, n)$ is less than $Th_1(i, \theta, n)$. ONUs report only packets that were not reported before with $R_1(i, \theta, n)$ value. $R_2(i, \theta, n)$ represents the accumulated buffer size starting from the first packet reported within $R_1(i, \theta, n)$ and is used for excess bandwidth allocation.

Upon receiving a report frame, the OLT acts as follows:

- It issues a bandwidth grant of size $G(i, \theta, n + 1) = R_1(i, \theta, n)$.

- If $R_1(i, \theta, n) = R_2(i, \theta, n)$ and both of them is less than G_i^{max} then the ONU is underloaded (U).

- If $R_1(i, \theta, n) = R_2(i, \theta, n)$ and both of them is less than $G_i^{max}(\theta, n)$ then the ONU is satisfied (S).

- If $R_1(i, \theta, n) < R_2(i, \theta, n)$ then the ONU is overloaded (O). The compensated bandwidth in that case is:

$$U(i, \theta, n) = Th_1(i, \theta, n) - R_1(i, \theta, n). \tag{6.8}$$

- The OLT issues a gate frame with $Th_1(i, \theta, n+1)$ (see next step) and $Th_2(i, \theta, n+1)$.

Upon receiving all report frames from a particular thread, the OLT invokes DES as follows:

- The OLT calculates the excess bandwidth share of each overloaded ONU as $E_i(\theta, n) = \frac{w_i}{\sum_{k \in O} w_j} \sum_{j \in U} G_i^{max} - R_1(i, \theta, n)$, where w_i is the weight assigned to each ONU according to service level agreement.

- The OLT updates ONUs maximum transmission window, $G_i^{max}(\theta, n+1)$, as

$$G_i^{max}(\theta, n+1) = \begin{cases} G_i^{max} + E_i(\theta, n) + U(i, \theta, n) & i \in O \\ G_i^{max} & i \in S \cup U. \end{cases} \tag{6.9}$$

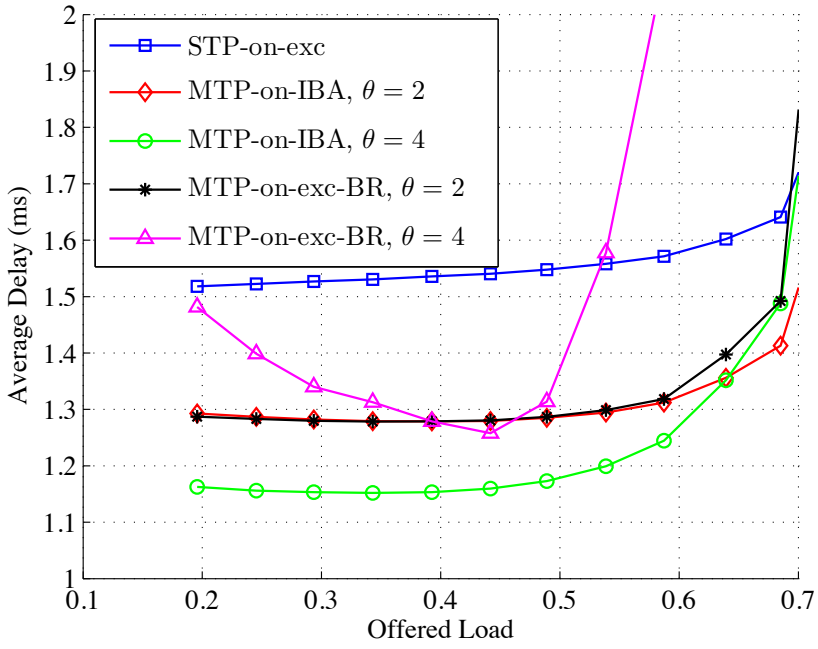
- The OLT updates $Th_1(i, \theta, n+1)$ value as:

$$Th_1(i, \theta, n+1) = G_i^{max}(\theta, n+1). \tag{6.10}$$

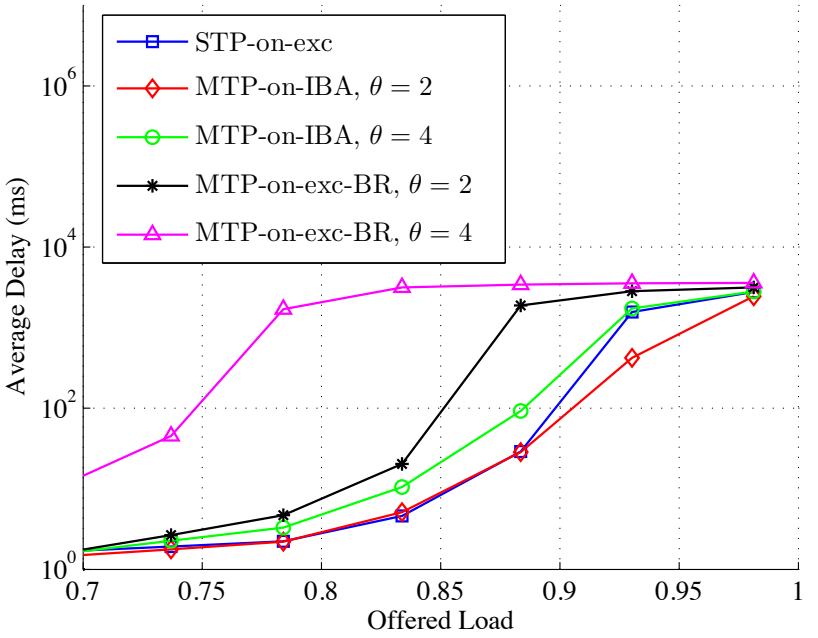
6.4 IBA Performance Evaluation

We consider an EPON with single OLT and 32 ONUs with 10 MB buffer size. The upstream and the downstream transmission rates are symmetric with 1 Gbps. Ethernet frames size ranges from 64 to 1518 B, (we use the packet size distribution reported in [11], with minimum inter-frame-gap (IFG) of 12 B and preamble of 8 B). For incoming traffic, we use self-similar traffic with long range dependence (LRD) and Hurst parameter 0.8 and packets are served in first-come-first-served (FCFS) order. Guard time is set to 1 μs and $G_i^{max} = \frac{15500}{\theta}$ B (single-thread case). MTP-on with IBA (MTP-on-IBA) is tested against MTP-on with BR and with online excess allocation [4, 22] (MTP-on-exc-BR). Single-Thread Polling with online excess (STP-on-exc) [4] is used as benchmark to evaluate the performance.

Average delay performance of tested schemes is shown in Figure 6.4. For offered load below 0.7 and θ = 2, both MTP-on-IBA and MTP-on-exc-BR are close to each other till load of 0.6. They almost achieve the minimum delay bound for MTP derived in (7.2). For



(a)



(b)

Figure 6.4: Average delay comparison of MTP-on-IBA, MTP-on-exc-BR, and STP-on-exc (a) low load (b) high load.

$\theta = 4$, MTP-on-IBA is stable till load of 0.65 while MTP-on-exc-BR is unstable and quickly degrades at load of 0.55. Figure 6.4 (b) show the performance at high load. The shown results indicate that MTP-on-exc-BR is less stable compared to STP-on-exc and MTP-on-IBR. Moreover, increasing the number of running threads leads to quick performance degradation. MTP-on-IBR, utilizing four threads, delay is more than STP-on-exc. On the other hand, using only two threads improves MTP-on-exc performance at load above 0.85.

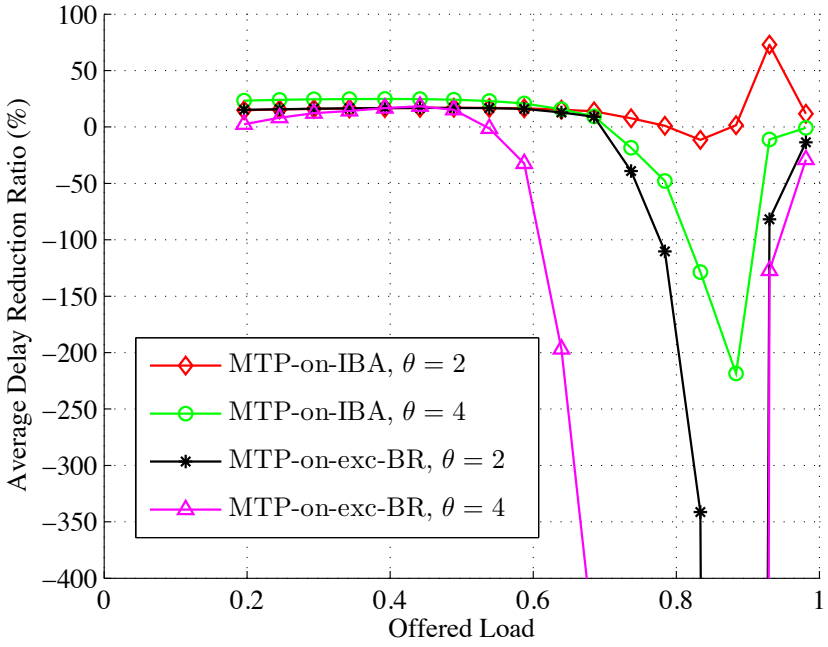


Figure 6.5: Average delay reduction ratio of MTP-on-IBA and MTP-on-exc-BR.

Figure 6.5 shows the average delay reduction of each scheme compared to STP-on-exc. It is clear that increasing the number of threads is beneficial at lower and medium load while degrading the performance at high load. MTP-on-IBA with two threads achieves delay reduction ratio of 15% at low load and 72% at high load. MTP-on-IBA with four threads achieves higher reduction (24%) at low load while achieving negative reduction up to -200% at high load. The reason behind IBA excel against BR is that IBA completely suppresses bandwidth over-granting and in the same time eliminates and compensates for

the unused slot reminders. On the other hand, BR suppresses bandwidth over-granting but it cannot remove nor compensates unused slot reminders. It is important to note two issues. First, BR compensation term is used to update the total demand from the OLT side to resolve the backlog traffic issue but the actual grants never exceed the maximum transmission window plus the excess bandwidth share. Second, as θ increases, the ONU maximum transmission window decreases and the effect of the unused slot reminders becomes more significant. This explains why MTP-on-BR (to some extent) is capable of maintaining good performance with increasing load.

6.5 Adaptive online Multi-Thread-Polling with IBA (AMTP-on-IBA)

In order to resolve MTP performance degradation at high load and maintain the good performance at low load, we propose an adaptive MTP-on (AMTP-on) version to smartly tune the number of threads (θ) between preset bounds:Θ_{max} and Θ_{min}. Θ_{min} should be set to one in order to adapt with high load case, while Θ_{max} is determined by network operator based on how much downstream bandwidth can be allocated to gate messages. In [65], the authors presented an adaptive Multi-Thread Polling scheme called Synergized-Adaptive Multi-GATE Polling With Void Filling (SAMGAV). Their idea is to map the number of running threads with the estimated offered load. SAMGAV has no limits on the maximum number of applied threads and hence it might cause uncontrolled gate message overhead in downstream direction. Moreover, their estimation of offered load is based on single sampling every relatively long interval ($\approxeq$ 2 ms) and hence it is not much accurate.

AMTP-on relies on the average attendance time ($\overline{W_{att}}$) to tune the number of active running threads. $\overline{W_{att}}$ is the average time between two consecutive bandwidth grants for the same ONU. Delay decomposition in MTP is shown in Figure 6.6. It can be seen that

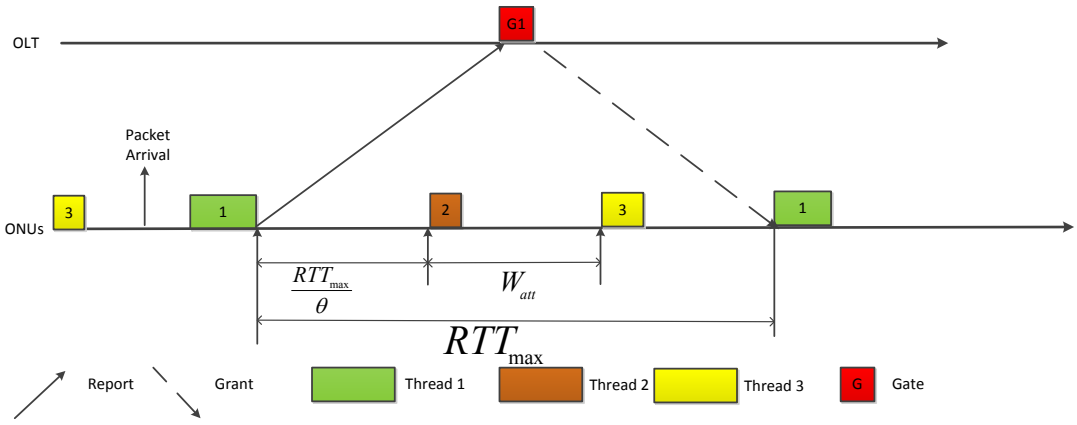


Figure 6.6: Delay decomposition of MTP-on.

the minimum delay bound, $\overline{W_{min}(\theta)}$, can be estimated via two methods. First,

$$W_{min}(\theta) = \overline{W_{att}}(\theta + 0.5), \tag{6.11}$$

or second,

$$W_{min}(\theta) = (1 + \frac{1}{(2\theta)})RTT_{max}. \tag{6.12}$$

AMTP-on works in two modes, namely collection mode and transient mode. In collection mode, the OLT runs MTP-on as usual and accumulates the values of W_{att} in order to calculate $\overline{W_{att}}$. Collection mode duration should be long enough to collect sufficient samples to get accurate $\overline{W_{att}}$. In the same time, collection mode should be short to quickly adapt to load fluctuations. Our experiments show that having 100-200 samples of W_{att} for each ONU is enough. Upon end of collection mode, the OLT decides either to keep the number of running threads unchanged, increase it by one, or decrease it by one. In the two later cases, AMTP-on switches to transient mode. In transient mode, ATMP-on does not collect W_{att} samples. In that mode, AMTP-on either initiates new thread or kills the last thread according to the decision in the last collection mode. AMTP-on enters transient mode to make sure that the action has been completely applied before entering collection mode again. In our simulation, we set transient mode length to be two full cycles.

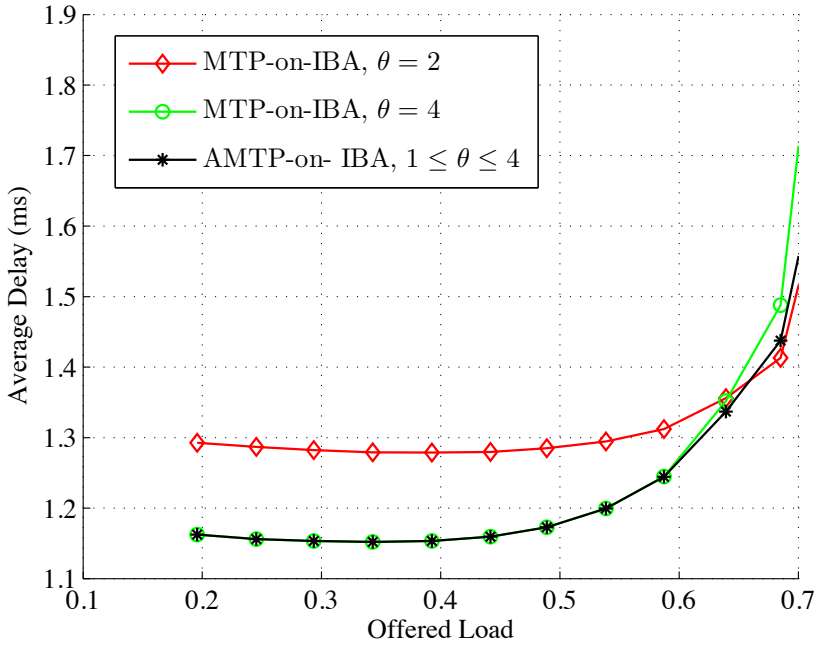
Upon end of collection mode and calculation of $\overline{W_{att}}$, tuning the number of active threads process works as follows:

- The OLT estimates the average accumulated delay bound ($W^{acc}_{min}(\theta)$) as in (6.11)
- The OLT estimates the average delay in case of the next adjacent (if applicable) threads: θ+1 and θ-1 as in (6.12)
- If $\overline{W_{att}} < (1+\epsilon)\overline{W^{exp}_{att}}$, then the OLT considers increasing the number of threads if $W^{acc}_{min}(\theta) > W_{min}(\theta+1)$. $\overline{W^{exp}_{att}}$ is the expected attendance time and is equal to $\frac{RTT_{max}}{\theta+1}$. ϵ is the accepted margin in attendance time. We use ϵ to avoid employing less threads unless the attendance time is deviated from the expected one. In our simulation experiments, we set ϵ to 0.1.
- If $\overline{W_{att}} > (1+\epsilon)\overline{W^{exp}_{att}}$, then the OLT considers decreasing the number of threads if $W^{acc}_{min}(\theta) > W_{min}(\theta-1)$.

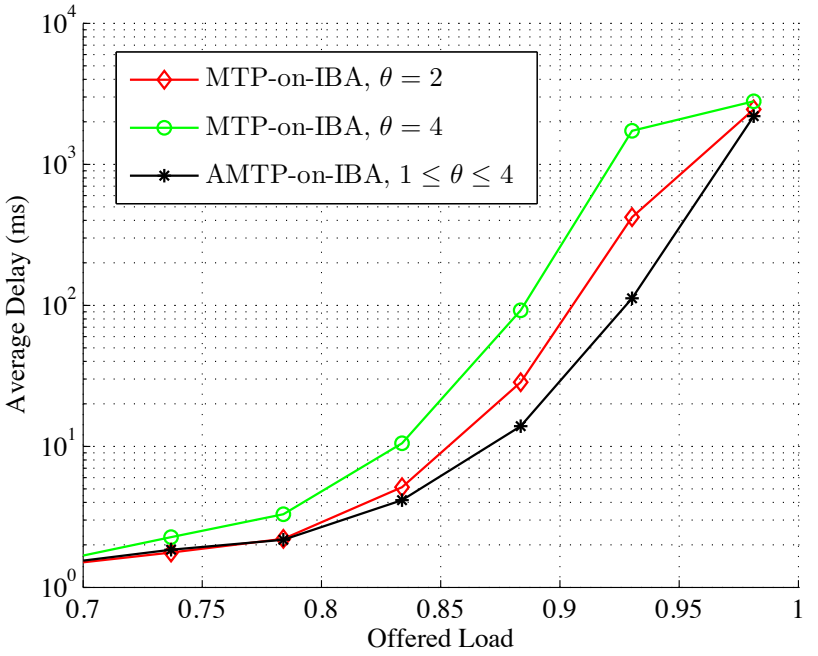
6.6 AMTP-on-IBA Performance Evaluation

We use the same simulation setup described in Section 6.4. We set Θ_{max} to 4 and Θ_{min} to 1. Figure 6.7 compares the performance of MTP-on-IBA and AMTP-on-IBA. At load below 0.7, AMTP-on-IBA can smartly choose the best performing thread count and consequently achieves almost optimal delay. At load above 0.7 (high load), AMTP-on-IBA is still stable compared to MTP-on-IBA for θ = 2 and 4.

In terms of delay reduction, Figure 6.8 shows that AMTP-on-IBA follows the best performing thread count till load of 0.8. After that, AMTP-on-IBA is still stable achieving delay reduction ratio up to 92%. The reason behind that is introduced in Figure 6.9. It shows the average number of active threads during simulation. One can see that the number of active threads is decreasing with offered load increase. At high load, AMTP-on-IBA is



(a)



(b)

Figure 6.7: Average packet delay comparison of MTP-on-IBA and AMTP-on-IBA (a) low load (b) high load.

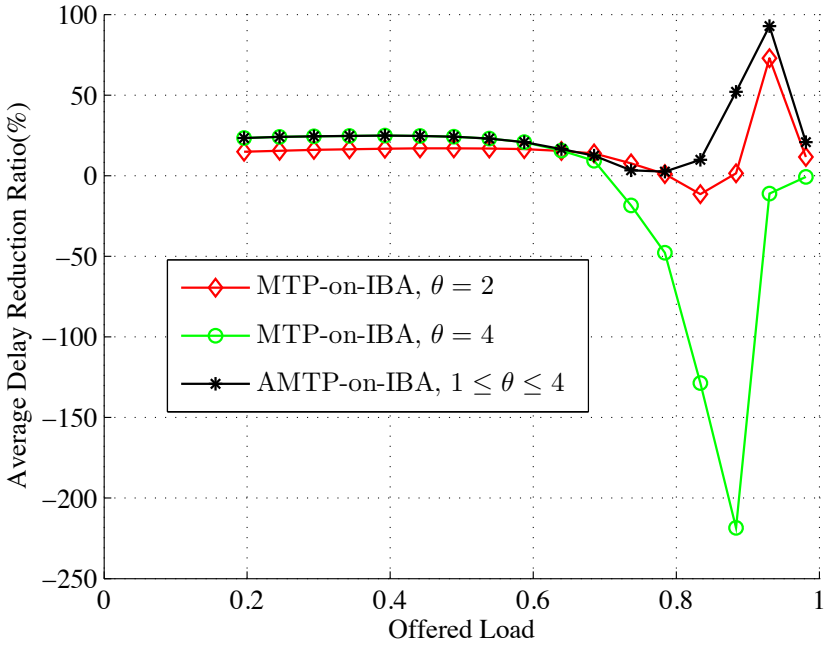


Figure 6.8: Average delay reduction ratio (%) of MTP-on-IBA and AMTP-on-IBA.

almost converged to Single-Thread Polling (θ = 1) and hence reduce both the control and guard bandwidth overhead.

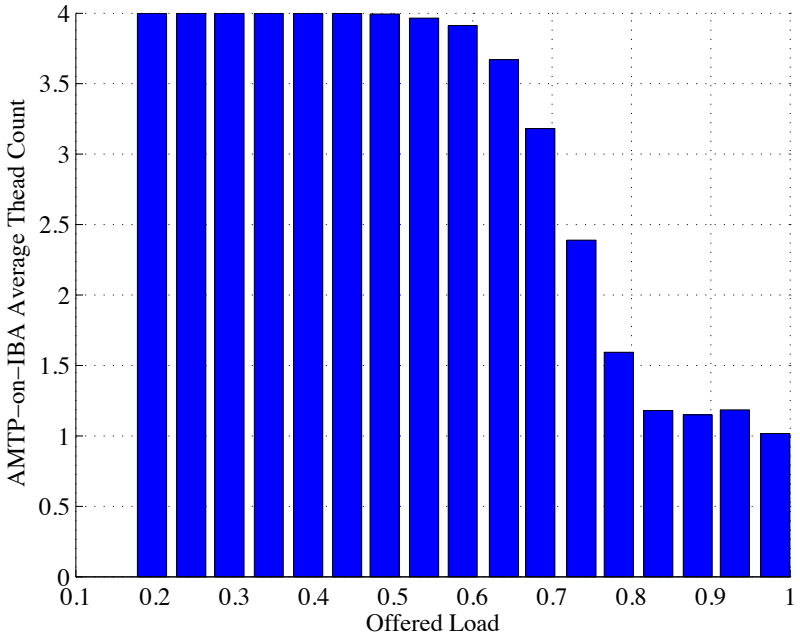


Figure 6.9: AMTP-on-IBA average thread count.

6.7 Summary

This chapter introduced a novel scheme, Interactive Bandwidth Allocation (IBA), to suppress bandwidth over-granting in MTP and in the same time pushes MTP performance to the max bound at low and moderate load while achieving significant performance at higher load compared to previous reported schemes. MTP-on-IBA achieves average delay reduction up to 25% at low load and 75% at high load. We also introduced AMTP which is an adaptive version of MTP to smartly tune the number of active threads along with the attendance delay. Numerical results show that AMTP-on-IBA is stable at high load achieving delay reduction of 92%. In the next chapter, we investigate an alternative approach to reduce the minimum delay bound of LR-EPONs through void filling.

Chapter 7

Parallel Void Thread in Long-Reach Ethernet Passive Optical Networks

7.1 Introduction

In Chapter 6, we explain packet delay decomposition of LR-EPONs. We also derive the minimum achievable average delay bound is 1.5 RTT_{max} for Single-Thread Polling (STP). In this chapter, we investigate another alternative of delay reduction through void filling. We also investigate how far void filling can contribute to delay reduction.

As a recall, the packet delay components W_{poll}, W_{grant}, and W_{queue} are shown in Figure 6.1. W_{poll} is the delay between packet arrival and bandwidth request transmission. The notations used hereafter are shown in Table 7.1. On average, W_{poll} equals half the duration of the cycle time. W_{grant} is the delay between grant request and grant assignment for certain packet. W_{grant} can span over multiple cycles but it cannot be less than $RTT_i + t_c$. Hence the minimum grant start time is

$$t_s^{min}(i,n) = t_e(i,n-1) + RTT_i + t_c. \tag{7.1}$$

W_{queue} is the delay between grant start and packet transmission and it depends on both the buffer size seen by the packet, the grant size, and the packet transmission discipline. In

Table 7.1: Definition of Variables.

N	Number of ONUs, indexed $i = 1, 2, ..., N$
C	Transmission capacity in bps
T_{cycle}^{max}	Maximum cycle time
G_i^{max}	Maximum allocated grant per cycle
$R(i,n)$	ith ONU bandwidth request for cycle n
$G(i,n)$	ith ONU bandwidth grant for cycle n
w_i	ith ONU weight according to service level agreement (SLA)
t_g	Guard time between consecutive grants
t_c	time needed to transmit Gate or report message
D_i	Distance between OLT and ith ONU
RTT_i	Round-trip-time delay between OLT and ith ONU
$\overline{RTT}$	Average round-trip-time delay
RTT^{max}	Maximum round-trip-time delay
RTT^{min}	Minimum round-trip-time delay
$t_s(i,n)$	ith ONU request based grant start time in cycle n
$t_e(i,n)$	ith ONU request based grant end time in cycle n
$t_s^{min}(i,n)$	ith ONU request based grant minimum start time in cycle n
$t_h(i,n)$	The horizon time after ith ONU grant allocation in cycle n
V_i	Void succeeding ith ONU request based grant
$V_s(i,n)$	Start time of void succeeding ith ONU request based grant
$V_e(i,n)$	End time of void succeeding ith ONU request based grant
W	Packet delay
$\overline{W}$	Average packet delay
W_{poll}	Polling delay
W_{grant}	Grant delay
W_{queue}	Queuing delay
K_B	The batch size in CCBVF
V_i^{max}	The maximum allocated void based grant for SCBVF

general we state that

$$\overline{W} \geq 1.5 RTT^{max}. \tag{7.2}$$

A DBA consists of two main functions, namely grant sizing and grant scheduling. Grant sizing determines the amount of bandwidth grant assigned to each ONU during cycle. This decision is based on the bandwidth requests sent from ONU via a report message during the previous grant. This kind of grant will be referred to as request based grant (RBG). Most of the proposed DBA schemes apply a limited grant sizing policy in which the grant assigned cannot exceed a certain threshold G_i^{max} [10]. Grant scheduling determines both start and end time of bandwidth grant. The most common scheduling approach is to use the horizon (first free) time or non-void filling (NVF) as shown in Figure 7.1. Assuming

cyclic polling, grant start time is given by,

$$t_s(i,n) = \begin{cases} max(t_h(N,n-1), t_s^{min}(i,n)) & i = 1, \\ max(t_h(i-1,n), t_s^{min}(i,n)) & 1 < i \leq N, \end{cases} \tag{7.3}$$

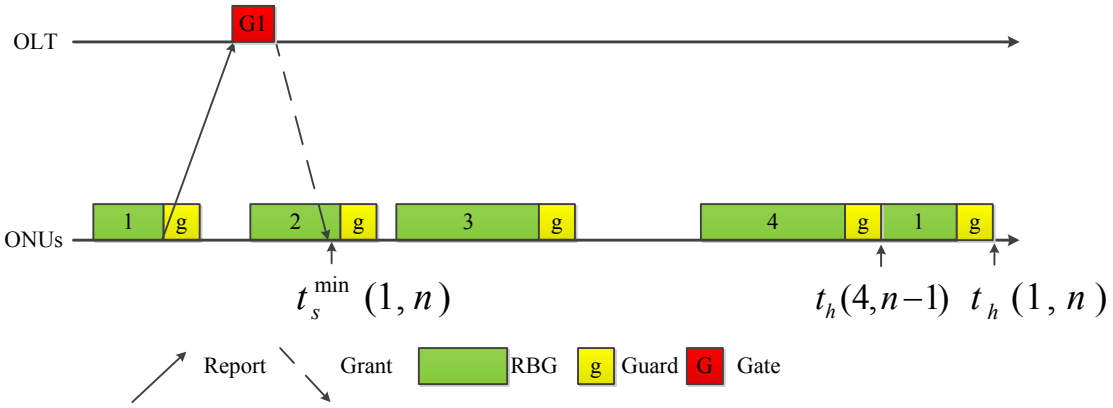


Figure 7.1: Non-void filling bandwidth allocation.

In long-reach Ethernet passive optical networks (LR-EPONs), the distance between OLT and ONUs spans over a longer distance (up to 100 Km) and hence longer RTT_i. This is reflected on both W_{grant} and void time periods between bandwidth grants [4]. In such a case, RTT_i will be the dominant factor controlling the packet delay at low and medium load [3] such that

$$\overline{W} \approx 1.5 RTT^{max}. \tag{7.4}$$

The negative effect of such voids motivates many researchers to propose new DBA schemes to further reduce the delay [4, 35, 38, 8, 66, 67, 68]. The proposed ideas fall into two categories. The first one aims at reducing the void size while maintaining the cyclic bandwidth allocation order [4, 8]. The second one aims at filling the void with request based grants [35, 38, 67]. The numerical results presented in [3, 35, 38] show that the proposed schemes could not achieve delay below 1.5 $\overline{RTT}$ (see section IV for more details). This is

very crucial for delay sensitive traffic.

In this chapter, we propose a Parallel Void Thread (PVT) approach to allocate bandwidth grants to ONUs during voids. Upon void detection, the OLT fills the detected void with bandwidth grants baseless of ONUs bandwidth requests. This would help transmitting more packets during unused bandwidth voids and reducing W_{grant}, as well as overall average delay. In order to be distinguished from RBGs, we will refer to grants allocated during voids baseless of grant requests as void based grants (VBGs). PVT partially falls in the second DBA category as it fills the void but independently from the original running thread. This would make PVT as a plus function that can be combined with any proposed DBA scheme from the first category (see subsection V-D for more details).

7.2 Parallel Void Thread

Parallel Void Thread (PVT) [9] relies on void detection after an RBG is assigned by the OLT as shown in Figure 7.2. Upon receiving grant request from ith ONU, the OLT schedules the RBG grant start time as in (7.3) and updates the horizon time, $t_h(i,n)$. In order to detect if there is a void succeeding ith ONU, the OLT compares $t_h(i,n)$ with $t_s^{min}(i+1,n)$. If

$$t_s^{min}(i+1,n) - t_h(i,n) > t_c + t_g, \tag{7.5}$$

then the OLT detects a void, V_i, with start and end times as,

$$V_s(i,n) = t_h(i,n) \tag{7.6}$$

and

$$V_e(i,n) = t_s^{min}(i+1,n). \tag{7.7}$$

If $i = N$, then the above equations are modified to,

$$t_s^{min}(1, n+1) - t_h(N, n) > t_c + t_g, \tag{7.8}$$

$$V_s(N, n) = t_h(N, n) \tag{7.9}$$

and

$$V_e(N, n) = t_s^{min}(1, n+1). \tag{7.10}$$

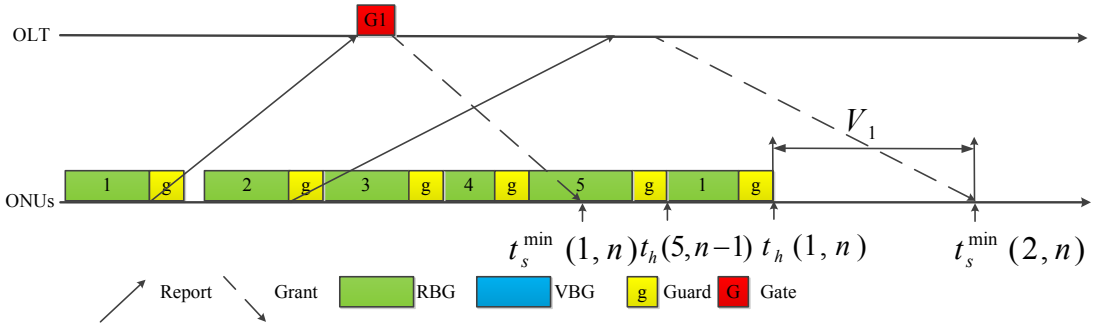


Figure 7.2: Void detection.

Upon void detection, the OLT invokes PVT immediately to fill the detected voids with void based grant(s) VBG(s). Since ONUs are polled in cyclic order, the void detection computation complexity is of $O(1)$. The size of VBGs only depends on the detected void duration. It is worth noting that during VBG, the ONU will not send bandwidth requests. In fact, bandwidth requests are only sent during RBG. We propose three different VBG sizing schemes to fill the detected voids, namely Void Extension (VE), count control batch void filling (CCBVF), and Size Controlled Batch Void Filling (SCBVF).

7.2.1 Void Extension

Void Extension (VE) assigns ith ONU a VBG with duration of the detected void V_i as shown in Figure 7.3. It is worth noting that VE is different from LOHEDA [67] in two

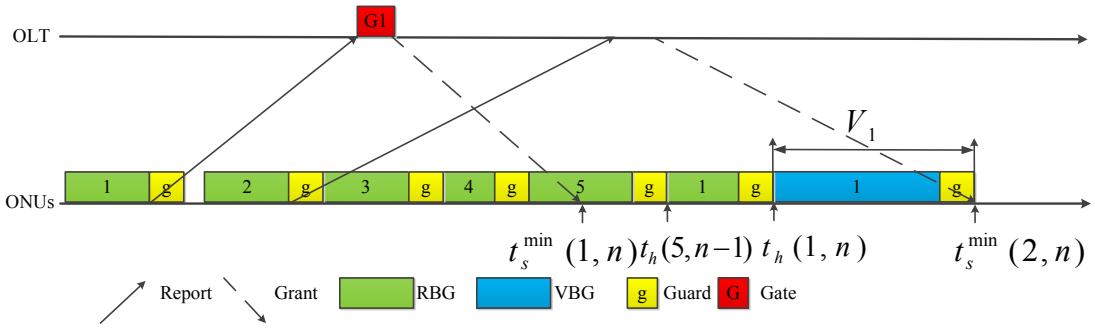


Figure 7.3: Void extension (VE).

ways. First, VE fills the void with a separate grant rather than merging both the RBG and VBG into single grant. This is because enlarging the grant will further delay the bandwidth request (report message) and creates another void in the next cycle [4]. Second, VE does not delay the grant of ONU $(i-1)$ until it receives the bandwidth request of ith ONU as LOHEDA does.

7.2.2 Count Controlled Batch Void Filling

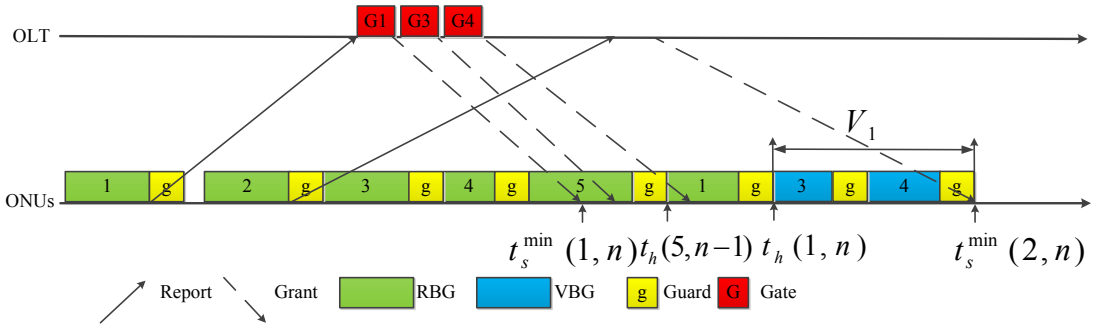


Figure 7.4: Count Controlled Batch Void Filling (CCBVF), (K_B = 2).

In CCBVF, a batch of K_B ONUs is assigned VBGs during the detected void. Figure 7.4 demonstrates CCBVF for $K_B = 2$. Regarding VBG assignment, ONUs are polled according to cyclic order that is independent of the cyclic order used for RBG. For CCBVF, an

eligible void must satisfy the following condition:

$$V_e(i,n) - V_s(i,n) \geq K_B \times (t_c + t_g). \tag{7.11}$$

This condition is necessary to ensure that the allocated VBGs are sufficient to send the shortest packet but not sufficient to ensure that all VBGs will be utilized. As K_B increases, VBG becomes shorter due to increased guard periods. The amount of VBG assigned to each ONU is

$$VBG_i = \frac{w_i}{\sum_{j=z}^{j=z+K_B-1} w_j}(V_e(k,n) - V_s(k,n) - K_B \times t_g). \tag{7.12}$$

7.2.3 Size Controlled Batch Void Filling

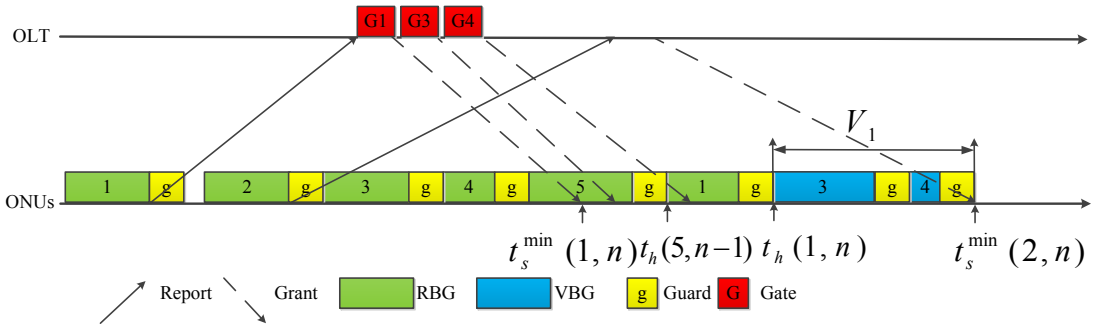


Figure 7.5: Size Controlled Batch Void Filling (SCBVF).

SCBVF [66, 68] assigns VBGs based on size rather on the batch count (K_B) as CCBVF. In SCBVF, each VBG does not exceed a certain threshold V_i^{max} which is set based on each ONU relative weight w_i such that:

$$\frac{V_i^{max}}{V_j^{max}} = \frac{w_i}{w_j}, \tag{7.13}$$

where i and $j \in \{0, 1, 2, ..., N-1\}$. SCBVF continues to allocate VBGs with the maximum void duration until a polled ONU cannot be granted its maximum void duration. In order

to maximize the benefit of void filling, this ONU will be granted the remaining grant rather that leaving it unscheduled. The operation of SCBVF is shown in Figure 7.5. In this figure, we observe that ONU 3 is allocated its maximum VBG size while ONU 4 is granted the remaining part.

7.3 Void Filling Schemes Delay Analysis

In this section, we present delay analysis for the proposed void filling techniques. We highlight the following:

- The delay analysis is presented for low to moderate load to be conforming with the assumptions presented in Section 7.3.1.
- The delay analysis purpose is to analytically explain the delay reduction mechanism of PVT and to provide a comparative analysis to NVF, RBPVF, and RBPVF.
- The outcome of the delay analysis is considered sufficient but not necessary condition for PVT to achieve delay lower than its competitors.

7.3.1 Preliminaries

In order to ensure that the existence of voids, we assume that,

$$\sum_{i=0}^{N-1} G(i,n) < RTT^{max} - Nt_g, \tag{7.14}$$

and

$$\sum_{i=0}^{N-1} G_i^{max} > RTT^{max} - Nt_g. \tag{7.15}$$

We also assume packets arrive according to Poisson process. For LR EPONs, we neglect W_{queue} as it is very small compared to W_{poll} and W_{grant}. We also assume that W_{grant}

only spans over one cycle as in the case of low load. The offered load is uniformly distributed among ONUs. This leads to uniformly distributed voids along W_{grant}.

It is clear from Figure 6.1 that W_{poll} depends on W_{grant}. For the best case, $W_{poll} = 0$ and for the worst case, $W_{poll} = W_{grant}$. On average, we can state that

$$\overline{W_{poll}} = 0.5\overline{W_{grant}}, \tag{7.16}$$

and

$$\overline{W} = 1.5\overline{W_{grant}}, \tag{7.17}$$

7.3.2 Non Void Filling (NVF)

For NVF, $W_{grant} = RTT^{max}$. Based on (7.17), the average delay of NVF is

$$\overline{W} = 1.5RTT^{max}. \tag{7.18}$$

7.3.3 Request Based Void Filling and Request Based Partial Void Filling

Both RBVF and RBPVF tries to fit the bandwidth grants as close to RTT_i as possible. For ith ONU, the grant delay is bounded by RTT_i and RTT^{max} as lower and upper bounds respectively. On average, It can be stated that:

$$\overline{RTT} \leq W_{grant} \leq RTT^{max}. \tag{7.19}$$

Based on (7.17) and (7.19), the average delay of RBPVF at low load is bounded by

$$1.5\overline{RTT} \leq \overline{W} \leq 1.5RTT^{max}. \tag{7.20}$$

7.3.4 Count Controlled Batch Void Filling (CCBVF) and Size Controlled Batch Void Filling (SCBVF)

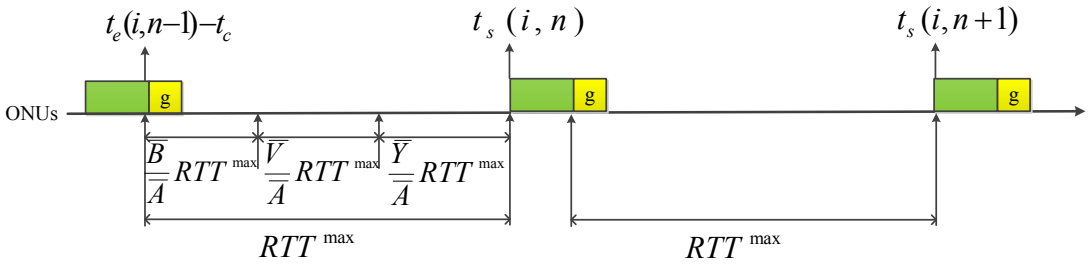


Figure 7.6: PVT packet decomposition.

Let $A(i,n)$ be the number of packets arrive between $t_e(i,n-1)-t_c$ and $t_e(i,n)-t_c$. Those packets are decomposed according to their transmission time as,

- $B(i,n)$: packets transmitted at the end of RBG slot during cycle n.
- $V(i,n)$: packets transmitted during VBG slot(s) allocated uniformly between $t_e(i,n)$ and $t_s(i,n+1)$.
- $Y(i,n)$: packets transmitted at the start of RBG slot during cycle $(n+1)$.

The timing diagram of PVT packets is shown in Figure 7.6. $B(i,n)$ mainly depends on both $V(i,n-1)$ and $A(i,n)$. $B(i,n)$ packets are transmitted during the unutilized bandwidth at the end of RBG slot of cycle n resulting from the transmission of $V(i,n-1)$ during the VBG slot(s) before the RBG slot. In order to calculate $\overline{W}$, the delay of each type should be calculated.

In order to make the analysis tractable, we consider the case where there are always remaining packets to be sent during RBG slot, (i.e., $Y(i,n) > 0$). This implies that the calculated average delay is an upper bound of the real average delay.

Let $\overline{B}$, $\overline{V}$, $\overline{Y}$, and $\overline{A}$ be the average values of $B(i,n)$, $V(i,n)$, $Y(i,n)$, and $A(i,n)$ respectively. It is also worth noting that for any two consecutive RBG slots, the following

relation is valid

$$t_s(i,n) - t_e(i,n-1) \approx RTT^{max}. \tag{7.21}$$

t_c and report message processing time are neglected as they are very small compared to round-trip-time in LR-EPONs.

$B(i,n)$ packets arrival starts at $t_e(i,n-1) - t_c$ and is on average $\frac{\overline{B}}{\overline{A}} RTT^{max}$ long. They are transmitted at the end of RBG slot of cycle n. This makes their average arrival time is $t_e(i,n-1) - t_c + \frac{\overline{B}}{2\overline{A}} RTT^{max}$ and their average transmission time is $t_e(i,n) - t_c$. Hence, the average delay of these packets ($\overline{W_B}$) is

$$\overline{W_B} = (1 - \frac{\overline{B}}{2\overline{A}}) RTT^{max}. \tag{7.22}$$

$V(i,n)$ packets arrival on average starts at $t_e(i,n-1) - t_c + \frac{\overline{B}}{\overline{A}} RTT^{max}$ and lasts for $\frac{\overline{V}}{\overline{A}} RTT^{max}$ long. $V(i,n)$ packets are transmitted between $t_e(i,n)$ and $t_s(i,n+1)$ The average delay of $V(i,n)$ packets is

$$\overline{W_V} = (\frac{3}{2} - \frac{\overline{B}}{\overline{A}} - \frac{\overline{V}}{2\overline{A}}) RTT^{max}. \tag{7.23}$$

$Y(i,n)$ are the packets yet to be sent during RBG slot of cycle $(n+1)$. Their transmission starts at $t_s(i,n+1)$, while their arrival on average starts after the $V(i,n)$ packets arrival and lasts for $\frac{\overline{Y}}{\overline{A}} RTT^{max}$ long. The average delay of such packets ($\overline{W_V}$) is

$$\overline{W_Y} = (\frac{3}{2} - \frac{\overline{B}}{2\overline{A}} - \frac{\overline{V}}{2\overline{A}}) RTT^{max}. \tag{7.24}$$

The average delay of PVT is given by

$$\overline{W} = \frac{\overline{B}}{\overline{A}} \overline{W_B} + \frac{\overline{V}}{\overline{A}} \overline{W_V} + \frac{\overline{Y}}{\overline{A}} \overline{W_Y}. \tag{7.25}$$

(7.25) is simplified to

$$\overline{W} = (\frac{3}{2} - \frac{\overline{B}}{\overline{A}} - \frac{\overline{V}}{2\overline{A}})RTT^{max}. \tag{7.26}$$

The remaining challenge is how to estimate both $\overline{B}$ and $\overline{V}$. In fact, $\overline{V}$ depends on the used grant sizing approach. Both CCBVF and SCBVF have different values for $\overline{V}$ and $\overline{B}$. In order to know estimate them, we should know the unfinished work distribution for the polling system under time limited service discipline. In fact, this is still not solved problem yet [69]. For Poisson arrivals, the void size to grant size ratio (VGR) can be computed as,

$$VGR = \frac{1}{\rho}(1 - \rho - \frac{Nt_g}{RTT^{max}}), \tag{7.27}$$

where ρ is the offered load. For the offered load below 0.5, VGR is greater than 1, which means that there is "theoretically" available bandwidth for all reported data to be sent during VBG(s).

7.3.5 Void Extension (VE)

The analysis of VE is similar to both CCBVF and SCBVF except for $V(i,n)$ packets. $V(i,n)$ packets are transmitted immediately after $t_e(i,n)$. The average delay of $V(i,n)$ packets is

$$\overline{W_V} = (1 - \frac{\overline{B}}{\overline{A}} - \frac{\overline{V}}{2\overline{A}})RTT^{max}. \tag{7.28}$$

Substitute by (7.28) in (7.25), we get

$$\overline{W} = (\frac{3}{2} - \frac{\overline{B}}{\overline{A}} - \frac{\overline{V}}{\overline{A}})RTT^{max}. \tag{7.29}$$

7.4 Discussion

In this section, we discuss the relevant delay, fairness, and overhead factors that is associated with void filling schemes.

7.4.1 Delay Discussion

The purpose of this section is to compare between the average delay of the proposed void filling schemes. Based on the average delay driven in (7.18), (7.26), and (7.29), It is clear that PVT delay is always less than NVF delay. CCBVF and SCBVF are guaranteed to achieve less delay than RBVF and RBPVF if

$$\frac{\overline{B}}{\overline{A}} + \frac{\overline{V}}{2\overline{A}} > \frac{3}{2}(1 - \frac{\overline{RTT}}{RTT^{max}}). \tag{7.30}$$

VE is guaranteed to achieve delay lower than 1.5 $\overline{RTT}$ bound if

$$\frac{\overline{B}}{\overline{A}} + \frac{\overline{V}}{\overline{A}} > \frac{3}{2}(1 - \frac{\overline{RTT}}{RTT^{max}}). \tag{7.31}$$

We will refer to $\frac{\overline{B}}{\overline{A}} + \frac{\overline{V}}{2\overline{A}}$ and $\frac{\overline{B}}{\overline{A}} + \frac{\overline{V}}{\overline{A}}$ as Batch Void Filling (BVF) bound (R_{BVF}) and VE bound (R_{VE}) respectively.

The derived lower bounds in (7.30) and (7.31) are sufficient but not necessary as the derived average delay in (7.26) and (7.29) is upper delay bound. For uniformly distributed ONUs between 80-100 Km, CCBVF and SCBVF are required to achieve R_{BVF} less than 15% to ensure average delay lower than the 1.5 $\overline{RTT}$ bound. However, VE can achieve the same figure with R_{VE} less than 15%.

7.4.2 Control Messages Proliferation

Control message proliferation is considered as a side effect of void filling techniques in both upstream and downstream directions. RBVF slightly increases both report and gate messages load. On the other hand, RBPVF increases the report message load more than RBVF as each partial void grant should end with a report message. PVT proposed approaches do not increase the report message load as VBGs do not conclude with a report message. In downstream direction, the situation is different based on what kind of PVT is

applied. For VE, there is no increase in gate message load as both RBG and VBG can be sent together in a single gate message. For CCBVF and SCBVF, the amount of increase depends on both K_B and V_i^{max}. It is also affected by the offered load which controls the average void duration.

7.4.3 Bandwidth Assignment Fairness

Both RBVF and RBPVF suffer from unfair bandwidth assignment. This is because their void filling techniques helps near ONUs to flush their data during voids rather than far ones. From (7.19), it is clear that both RBVF and RBPVF have higher grant delays for far nodes.

Regarding PVT VBG sizing approaches, VE is less fair compared to CCBVF and SCBVF. In VE, the VBG size assigned to each node depends on the ONU RBG size, the next node previous RBG size, and the round-trip-time. In case of non-uniform load, ONUs VBG assignment is not strictly fair. CCBVF is more fair compared to SCBVF as it allocates VBGs based on ONUs relative weights but this comes on the expense of VBG size control. SCBVF is less fair as the last scheduled ONU during a void should be assigned a VBG less than its maximum VBG value compared to the other ONU scheduled in the same void. Although voids are not similar in the size. However, on average, both CCBVF and SCBVF assign VBG more fair than VE.

7.4.4 Dynamic Bandwidth Allocation

PVT can be integrated with almost all proposed DBA schemes for LR-EPONs. For offline (interleaved-polling-with-stop) DBA schemes, PVT can be used with offline Multi-Thread polling (MTP-offline) [8], offline single-thread (STP-offline) [8], and Double Phase polling (DPP) [23]. It also can be used with online (interleaved) polling such as online Multi-Thread polling (MTP-online) [4] or online Single-Thread Polling with online excess allocation (STP-on-exc) [4]. The results in [4] show that STP-on-exc outperforms MTP-offline and MTP-online at T_{cycle}^{max} up to 4 ms. Moreover, STP-on-exc is not as complex as Multi-

Thread polling in terms of reporting process and thread tuning. Although MTP-online has less void size since there is ΘN RBGs per cycle for Θ threads. This implies that the cycle contains almost the same idle time amount but divided into more slots.

For offline schemes, SCBVF efficiently utilizes more voids compared to VE and CCBVF. Since in offline polling, there is a single large void at the end of cycle, VE will assign this void to the last node in the cycle, while CBBVF will divide it among K_B VBGs. SCBVF seems to be a reasonable option in that case as it will utilize the large void with the maximum possible VBGs. On the other hand, all PVT approaches efficiently work with online polling. For more information about the DBA schemes for LR-PONs, the reader might refer to [70].

7.4.5 PVT Computation Complexity

As we mentioned before, PVT consists of two main functions: void detection and VBG sizing. Void detection computation complexity is $O(1)$, while VBG sizing complexity mainly depends on the grant sizing approach. In general, the complexity of allocating single VBG is $O(1)$ as PVT is independent from the running DBA thread. For VE, the computation complexity is $O(1)$ as it allocates a single VBG per void. CCBVF has computation complexity of $O(K_B)$ as it performs K_B VBG allocations per void. Regarding SCBVF, the computation complexity depends on the number of VBG allocations per void. For the worst case scenario, the computation complexity is $O(\lceil \frac{RTT^{max}-RTT^{min}}{V_i^{max}} \rceil)$ (assuming all ONUs have the same SLA). This case is very rare as it requires that all ONUs have no data to send and the nearest ONU RBG is followed by the most far ONU. This case at most happens once per cycle. In realistic scenario, the average computation complexity is $O(\lceil \frac{average\,void\,size}{V_i^{max}} \rceil)$.

7.5 Simulation Setup

We develop an event driven EPON simulator using C++. Our simulator consists of three modules. The first module generates synthetic traffic, the second module simulates the underlying DBA scheme, and the third module simulates the PVT process.

We consider an EPON with single OLT and 32 ONUs with 10 MB buffer size. Otherwise mentioned, the upstream and the downstream transmission rates are symmetric with 1 Gbps. Ethernet frames size ranges from 64 to 1518 B, (we use the packet size distribution reported in [11], with minimum inter-frame-gap (IFG) of 12 B and preamble of 8 B). We use two scenarios for the incoming traffic. The first scenario is single service self-similar traffic with long range dependence (LRD) and Hurst parameter 0.8 and packets are served in first-come-first-served (FCFS) order. The second scenario is differentiated services (DiffServ) traffic with 3 classes of service: Expedited Forward (EF), Assured Forward (AF), and Best Effort (BE). EF ONU offered load share is 20%, while the rest is divided equally between AF and BE. EF is constant-bit-rate (CBR) traffic with Poisson arrivals and fixed packet size of 70 B, while AF and BE traffic are self-similar with long range dependence and Hurst parameter 0.8. In order to maintain a fair comparison, we impose a strict priority queuing [18] for Intra-ONU scheduling in the case of DiffServ traffic. T_{cycle}^{max} is set to 4 ms, t_g is set to 1 μs and G_i^{max} = 15500 B. The offered load is distributed uniformly over ONUs. We choose STP-on-exc [4] as the underlying non-void DBA scheme with excess pool bound of NG_i^{max} bytes. STP-on-exc is the main running thread in case of PVT, RBVF, and RBPVF simulation. We will refer to STP-on-exc as NVF. PVT is compared against NVF, RBVF, and RBPVF.

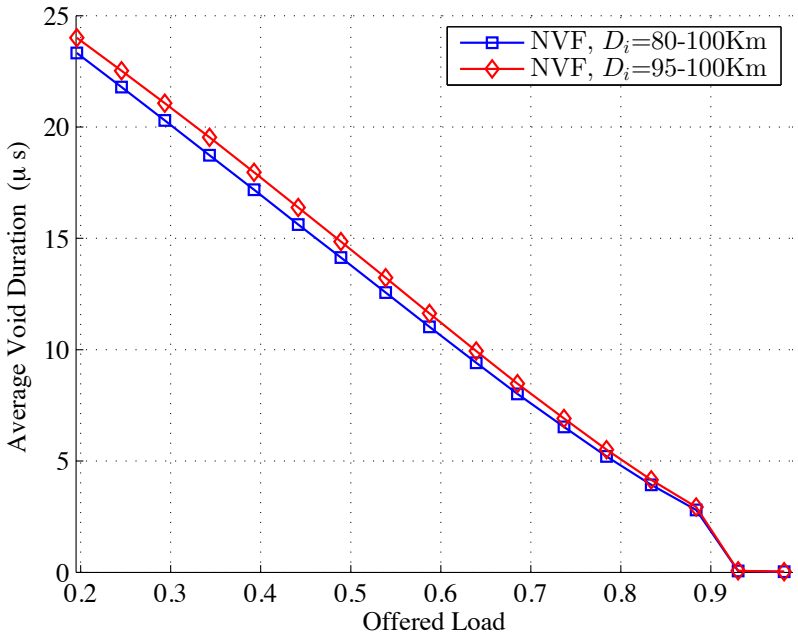


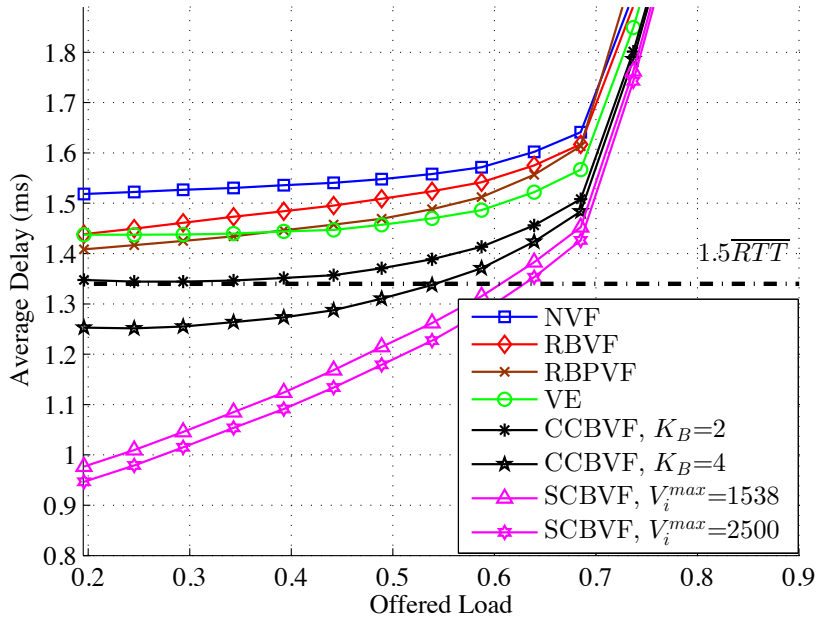
Figure 7.7: Void duration between consecutive RBGs.

7.6 Performance Evaluation

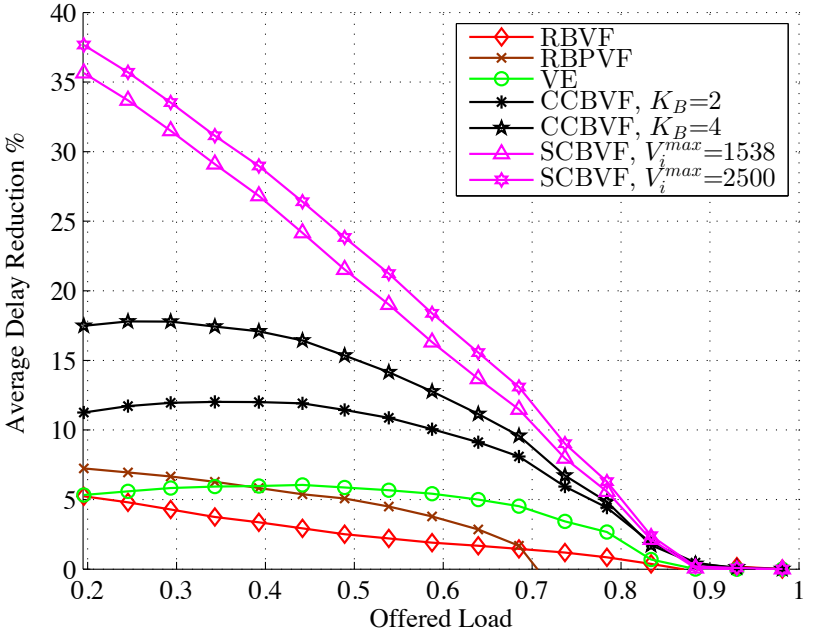
7.6.1 Single Service Traffic

The average void duration of STP-on-exc is shown in Figure 7.7 for two different distance spans. The results show the void duration decreases with offered load increase. The void duration approximately ranges from 15 to 24 μs at load below 0.5. If we consider the total voids per cycle duration ($\approx$ 1 ms), the idle time duration represents 50-75% of the cycle length.

Figure 7.8 shows the delay reduction comparison among NVF, RBVF, RBPVF, VE, CCBVF, and SCBVF for D_i = 80-100 Km. At 0.7 load and above, all schemes have approximately similar delay. CCBVF and SCBVF are better than NVF, VE, RBVF, and RBPVF for all load range below 0.7. The delay reduction ratio is calculated based on NVF delay. RBVF achieves delay reduction ratio is up to 5%. VE achieves a delay reduction up to 6%. RBPVF outperforms VE for load range below 0.4. Regarding CCBVF, it achieves



(a)



(b)

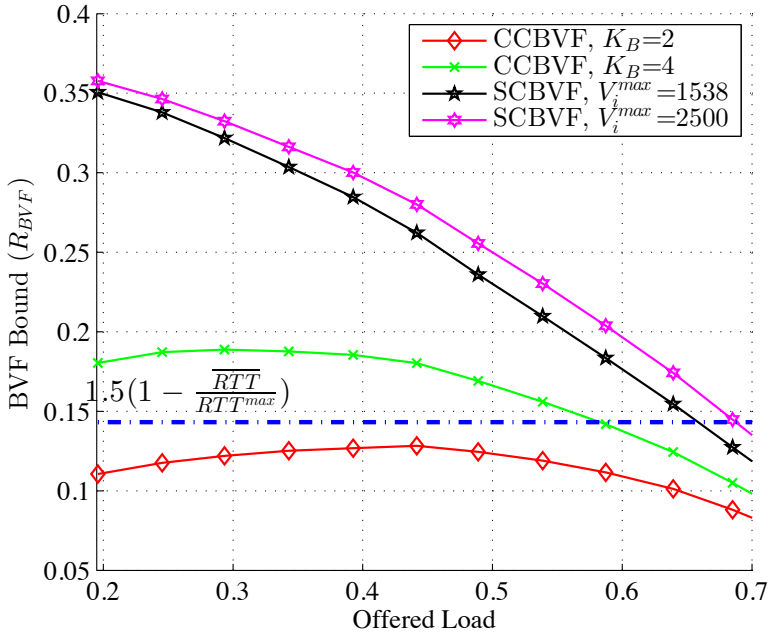
Figure 7.8: (a) Average delay, D_i = 80-100 Km (b) Average delay reduction, D_i = 80-100 Km.

delay reduction up to 12% for K_B = 2 and 19% for K_B = 4. SCBVF achieves the largest delay reduction compared to the other schemes. Its delay reduction is up to 35% and 37%

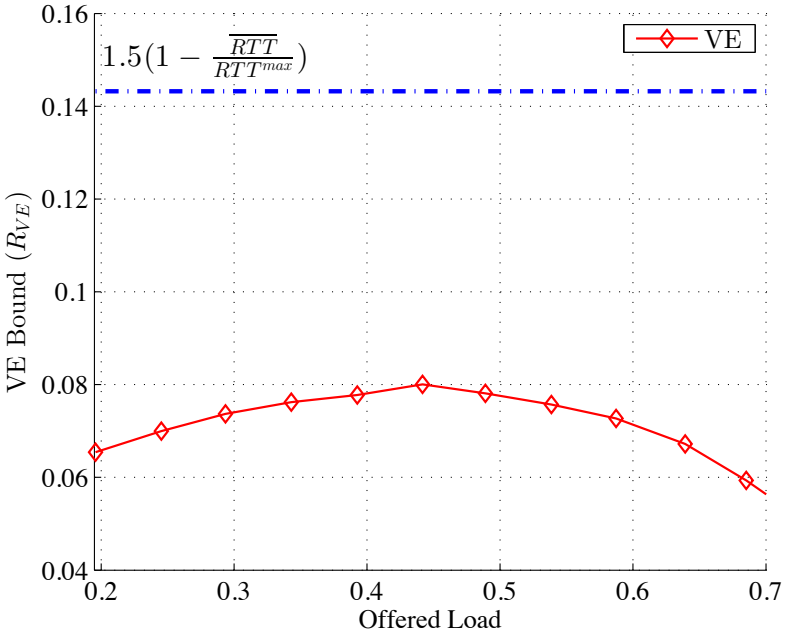
for v_i^{max} = 1538 B and v_i^{max} = 2500 B respectively. It is also worth mentioning that both CCBVF and SCBVF achieves an average of a delay lower than 1.5 $\overline{RTT}$ bound. Both CCBVF and SCBVF delay begins to quickly increase starting from 0.6 offered load. This is because that the time between consecutive grants begins to increase beyond RTT^{max} due to increased burst arrivals.

In Figure 7.9, we show the bounds derived in (7.30) and (7.31). We observe that although these bounds are for Poisson arrivals and are sufficient but not necessary, they approximately hold for LRD traffic. Figure 7.9 (a) shows the bound for CCBVF and SCBVF. It can be seen that CCBVF did not achieve the minimum bound for $K_B = 2$ and hence it did not achieve delay lower than 1.5 $\overline{RTT}$ (see Figure 7.8). However, CCBVF achieves R_{BVF} more than bound up to load $\approx$ 0.55. SCBVF also achieves R_{BVF} more than the bound up to load $\approx$ 0.65. VE did not achieve R_{VE} more than the bound in order to achieve average delay lower than 1.5 $\overline{RTT}$. The reason behind VE, CCBVF, and SCBVF excel is shown in Figure 7.10 and Figure 7.11. They show four performance measures, namely traffic ratio transmitted during VBG ($\frac{\overline{V}}{\overline{A}}$), average delay of traffic transmitted during VBG ($\overline{W_V}$), average delay of traffic transmitted during RBG, and VBG utilization.

Figure 7.10(a) shows that the traffic transmitted during VBGs decreases with the offered load as the available voids becomes shorter. VE achieves $\frac{\overline{V}}{\overline{A}}$ of 6%, CCBVF achieves up to 28% for K_B = 4, and SCBVF achieves up to 52% for V_i^{max} = 2500 B. The reason behind VE performance lag compared to CCBVF and SCBVF can be explained as follows: Although VE archives the minimum delay in VBG, this is not enough to achieve large overall delay reduction as the traffic percent transmitted during VBG is less than the other competitors. In fact, the void duration is to great extent correlated with RTT_i and $G(i, n)$. When ONU is lightly loaded, it is more likely to have a larger void succeeding it. In that case, VE will not be of much good as there is no much traffic to be sent during VBG. On the other hand, batch void filling enables more packets to be sent during VBG. This observation is emphasized by VBG utilization shown in Figure 7.11(b).



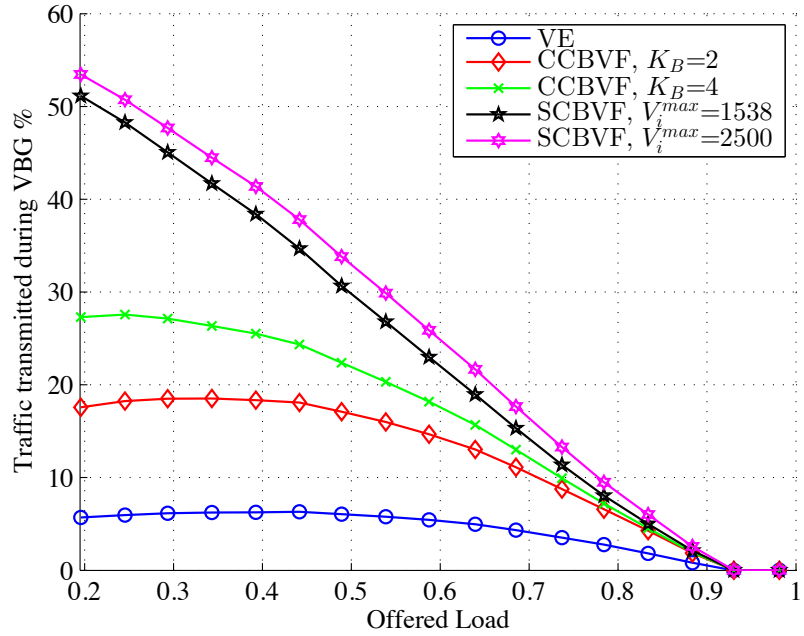
(a)



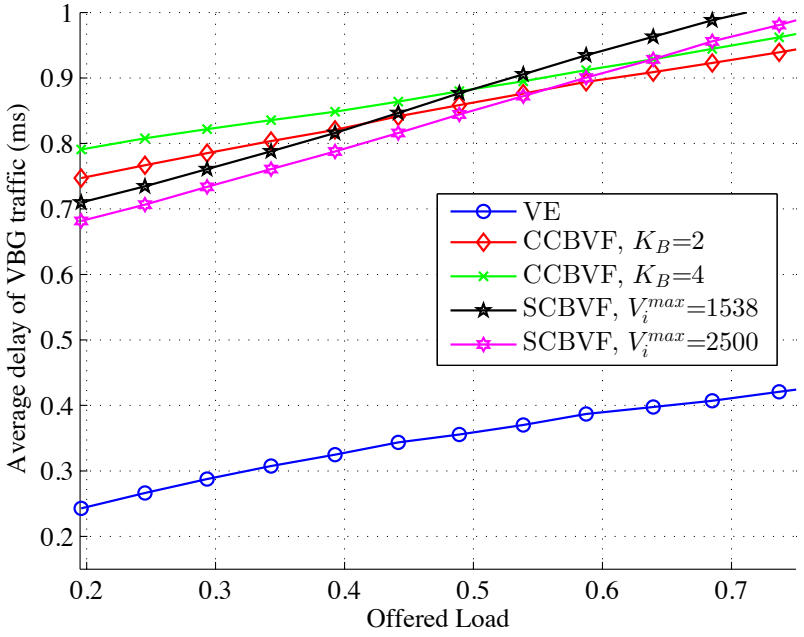
(b)

Figure 7.9: (a) BVF bound (b) VE bound, D_i = 80-100 Km

Although SCBVF has comparable VBG traffic delay to CCBVF, SCBVF can transmit more packets during VBG and hence reduce the overall traffic. Figure 7.11(a) highlights

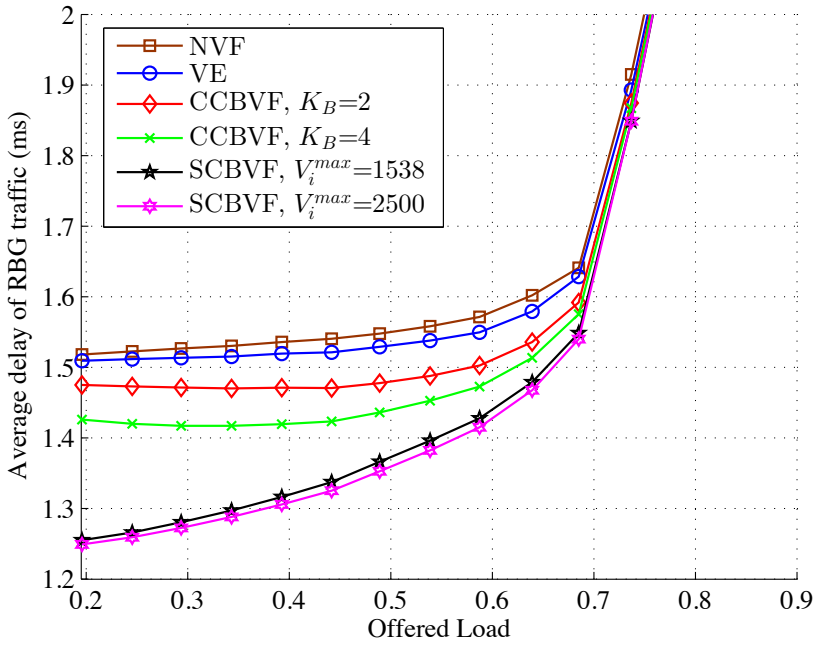


(a)

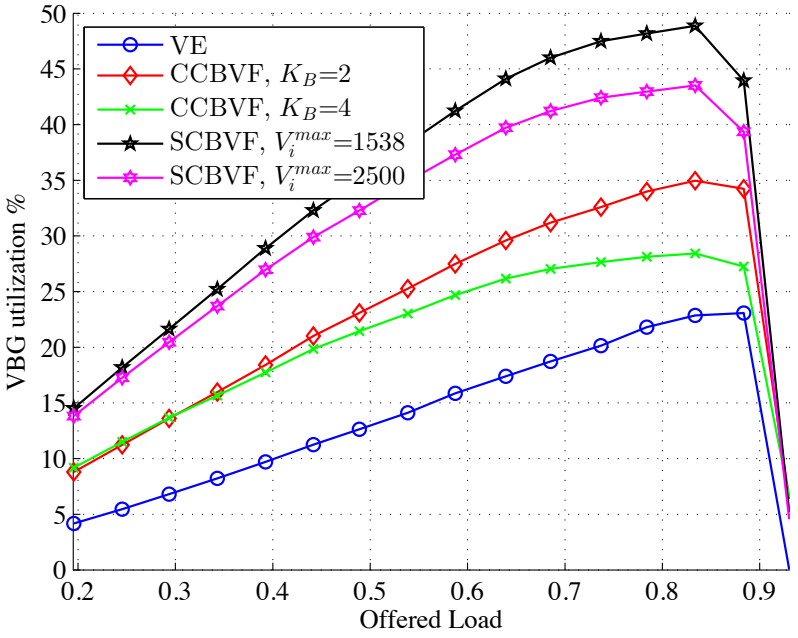


(b)

Figure 7.10: Comparison between VE, CCBVF, and SCBVF (D_i = 80-100 Km) (a) Traffic % transmitted during VBG (b) Delay of transmitted packets during VBG.



(a)



(b)

Figure 7.11: Comparison between VE, CCBVF, and SCBVF (D_i = 80-100 Km) (a) Delay of transmitted packets during RBG (b) VBG utilization.

that VE, CCBVF, and SCBVF have less RBG delay compared to NVF. SCBVF has the least delay of RBG traffic because it is capable of sending more packets during VBGs (high $\frac{\overline{V}}{\overline{A}}$). This would provide available bandwidth for some of recent arriving packets to be sent at the end of the next RBG (more $\frac{\overline{B}}{\overline{A}}$). It can be noticed that the traffic ratio sent during VBGs and the RBG traffic delay are correlated. In other words, the scheme that manages to send more packets during VBGs is the one able to achieve the lowest delay during RBGs.

VBG utilization performance has two phases. The first phase, at low and medium load, VBG utilization increases with load increase making use of longer available voids and increased buffered packets. In the second phase (at high load), VBG utilization begins to decrease with load increase as the voids becomes shorter and more large frames cannot be fitted within such voids (frame delineation) [29].

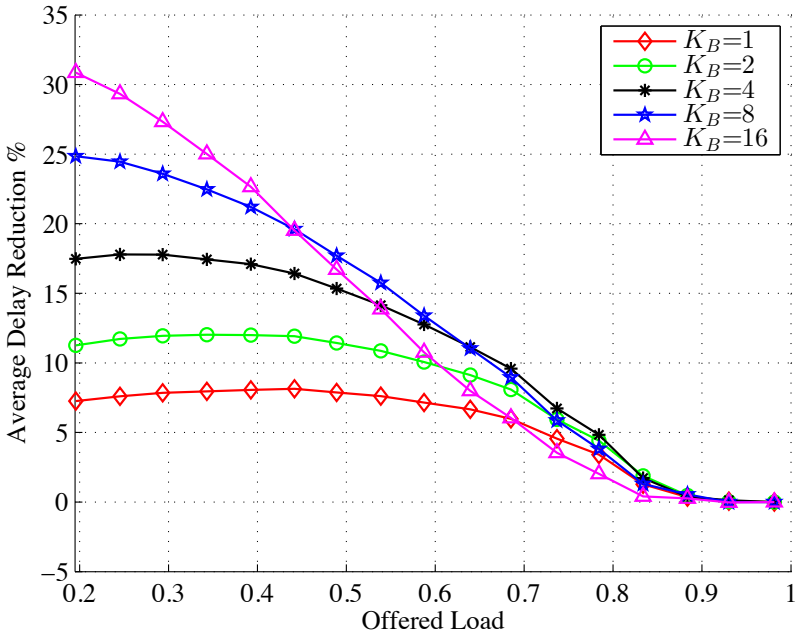


Figure 7.12: CCBVF delay reduction ratio (D_i = 80-100 Km).

Figure 7.12 shows the batch size, K_B, on CCBVF average delay. It is clear that larger batch size leads to increased delay reduction at lower load. In fact, choosing large K_B is more beneficial at lower load rather than at higher load. Employing large K_B quickly degrades performance with load increase. This is because the void size decreases with

offered load increase and more bandwidth will be wasted as guard band and VBG size becomes shorter to fit large packets. In Figure 7.12, using batch size of 16 ONUs is superior at low load and quickly degrades with load increase.

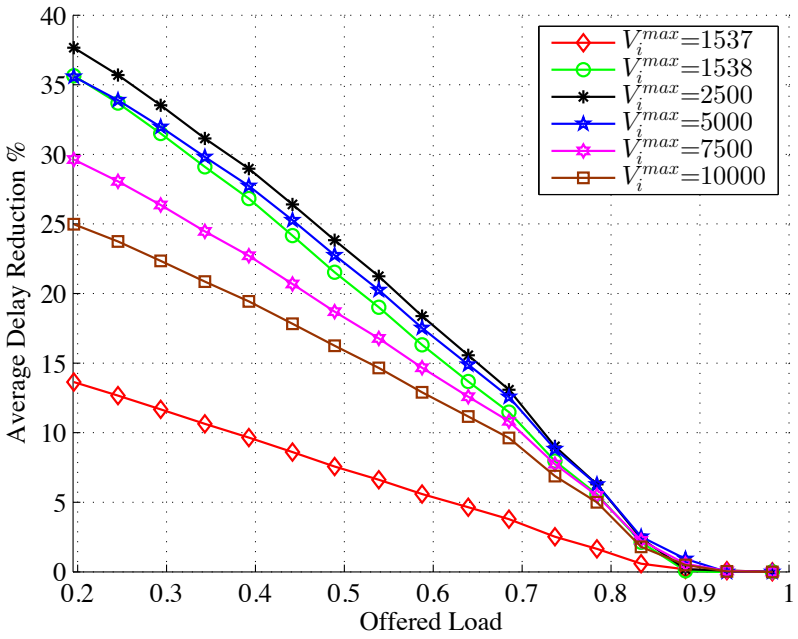


Figure 7.13: SCBVF delay reduction ratio (D_i = 80-100 Km).

The effect of V_i^{max} on SCBVF is shown in Figure 7.13. One interesting note to highlight is the significant difference in performance for V_i^{max} equals 1537 and 1538 B. This is due to the head of line blocking (HOLB) phenomena. In HOLB, a large frame can block ONU from utilizing all assigned VGBs during one cycle and hence causes performance degradation. Since the largest Ethernet frame requires 1538 B to be transmitted including inter-frame-gap and preamble, it is recommended that V_i^{max} should not be less than that value. In Figure 7.13, SCBVF performance is improving with V_i^{max} increase but this behavior is reversed when V_i^{max} is close to or greater than G_i^{max}. One interesting point for future research investigation is how to adapt K_B or V_i^{max} with void utilization and VBG traffic ratio to achieve a better performance with daily traffic variations through the access network.

Gate message proliferation is shown in Figure 7.14. VE and NVF have identical gate

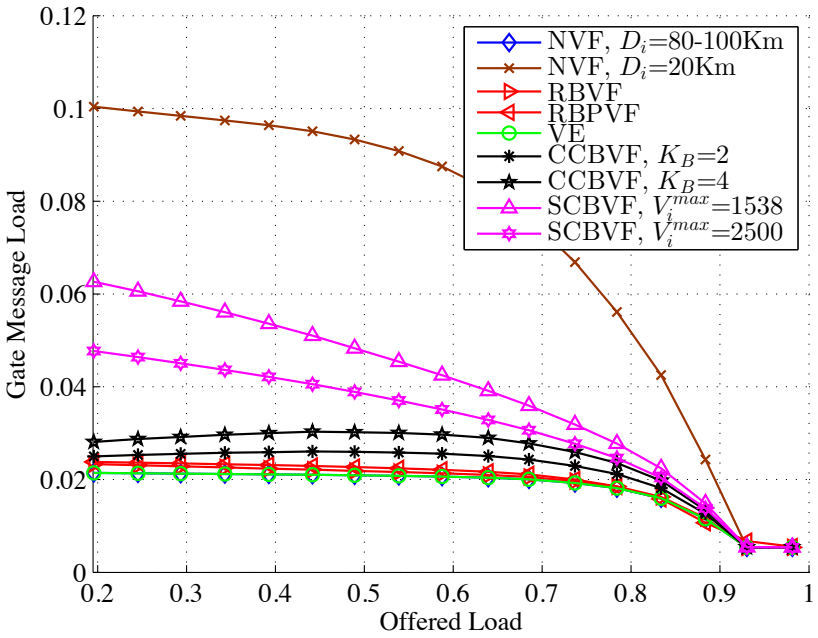


Figure 7.14: Gate message proliferation, D_i = 80-100 Km.

message load. RBVF is much close to both VE and NVF. Both CCBVF and SCBVF have larger gate message load compared to the other schemes. The gate message load increases when more ONUs are assigned VBGs during voids. This explains why the gate message load increases with larger K_B while decreases with larger V_i^{max}. In order to achieve a fair comparison, we compare both CCBVF and SCBVF with NVF at short range (D_i =20 Km) EPONs. It can be noticed that both CCBVF and SCBVF gate message load is far less than NVF in short range EPONs. Based on this note, we observe that the control message proliferation caused by CCBVF and SCBVF is considered acceptable price for the high average delay reduction ratio they achieve.

Figure 7.15 shows the average delay reduction compared to NVF for D_i = 95-100 Km. It shows that both RBVF and RBPVF are much affected by the reduced distance span compared to 20 km distance span presented in Figure 7.8 (b). However all PVT VBG sizing schemes did not face any performance degradation in terms of delay reduction ratio. Both VE and CCBVF achieves better delay reduction than the 20 km distance span. SCBVF is less sensitive to distance span as it is almost achieves the same reduction ratio. It can be

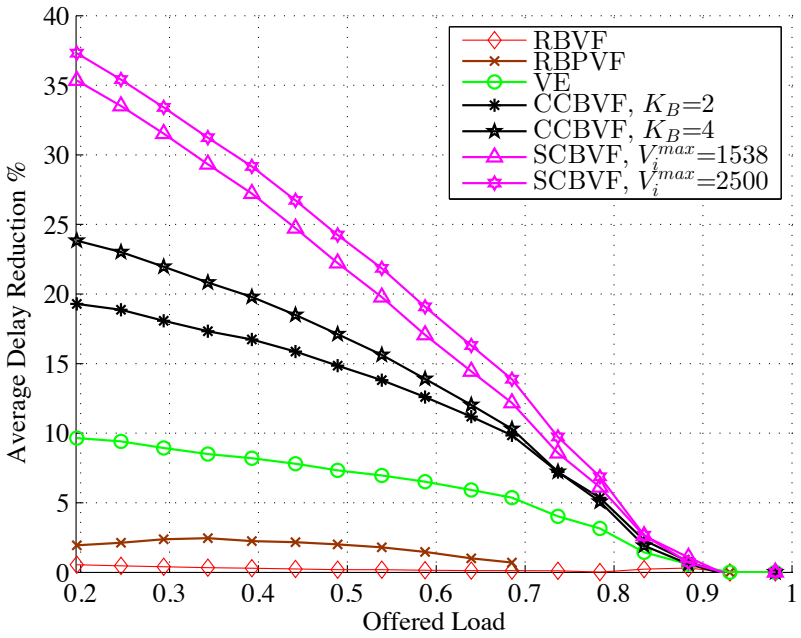


Figure 7.15: Delay reduction ratio, D_i = 95-100 Km.

inferred that PVT is strong competitor for both RBVF and RBPVF in both cases of long and short differential span distance.

7.6.2 DiffServ Traffic

Figure 7.16 shows the time between consecutive bandwidth grants for NVF, RBVF, RBPVF, VE, CCBVF, and SCBVF. Both CCBVF and SCBVF have significant lower time period between consecutive bandwidth grants (between 0.2-0.5 ms), while it is between (0.9-1 ms) for the other schemes. This means that both CCBVF and SCBVF are able to utilize the voids more efficiently. The short time between grants helps ONU to flush the higher priority packets and delay sensitive traffic more frequently and thus reduces their average delay. It also helps the lower priority packets (BE) to be transmitted mainly in RBG and small portion of them during VBG.

Figure 7.17 shows EF traffic average delay performance. Both CCBVF and SCBVF achieves delay reduction up to 63% and 81% respectively. VE performance quickly de-

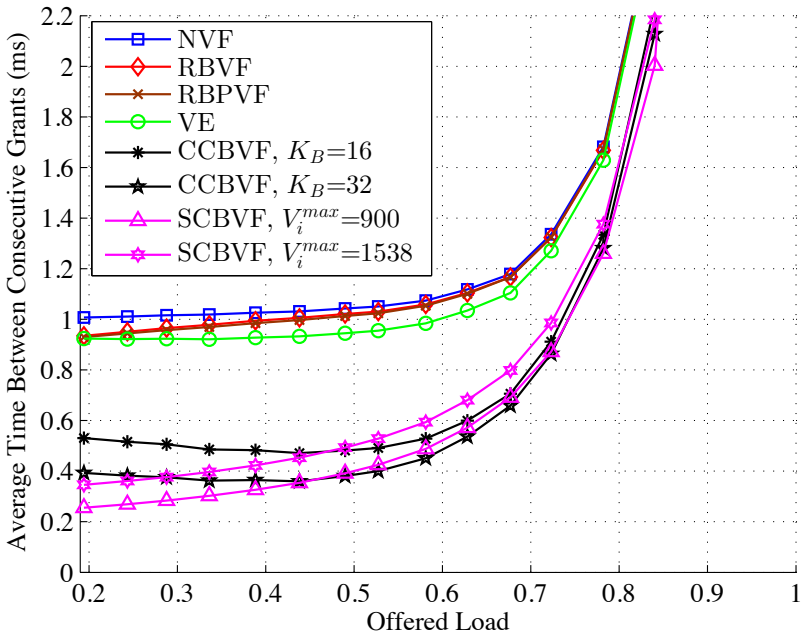
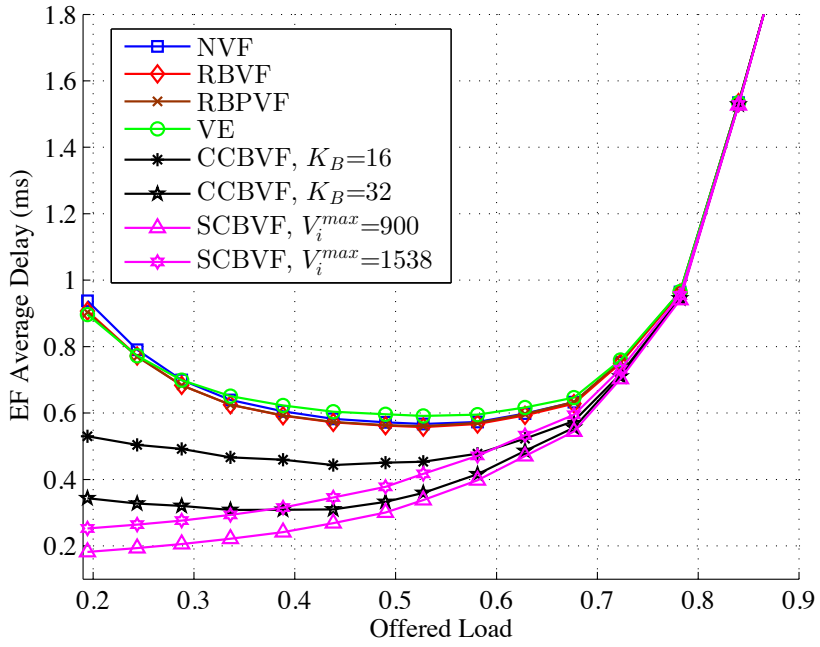


Figure 7.16: Average time between two consecutive grants.

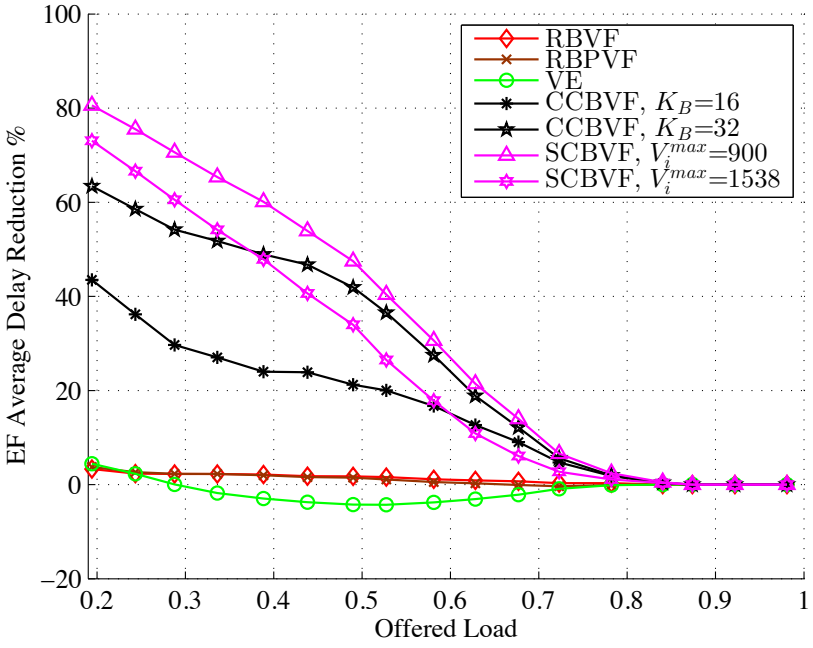
grades with load increase and shows negative delay reduction. This is because the uncontrolled VBG is more beneficial for lower priority traffic classes. It is also shown that reducing VBG size is more beneficial to EF traffic. While SCBVF EF performance improves with decreasing V_i^{max}, CCBVF performance improves with increasing K_B. It is also notable that SCBVF is less sensitive to HOLB as EF traffic packets are much smaller than V_i^{max}.

Both CCBVF and SCBVF switch their positions for the AF traffic performance as shown in Figure 7.18. CCBVF outperforms SCBVF because its relatively higher VBG size allows more AF traffic packets to be transmitted during VBG compared to SCBVF. CCBVF achieves delay reduction up to 52% compared to 45% achieved by SCBVF.

Regarding BE traffic (Figure 7.19), VE shows moderate improvement with delay reduction up to 10%. Both SCBVF and CCBVF continue their excel with delay reduction up to 52% and 56% respectively. It is important to emphasize that choosing large K_B value might be beneficial to EF traffic but the performance quickly degrades in case of AF and BE classes.

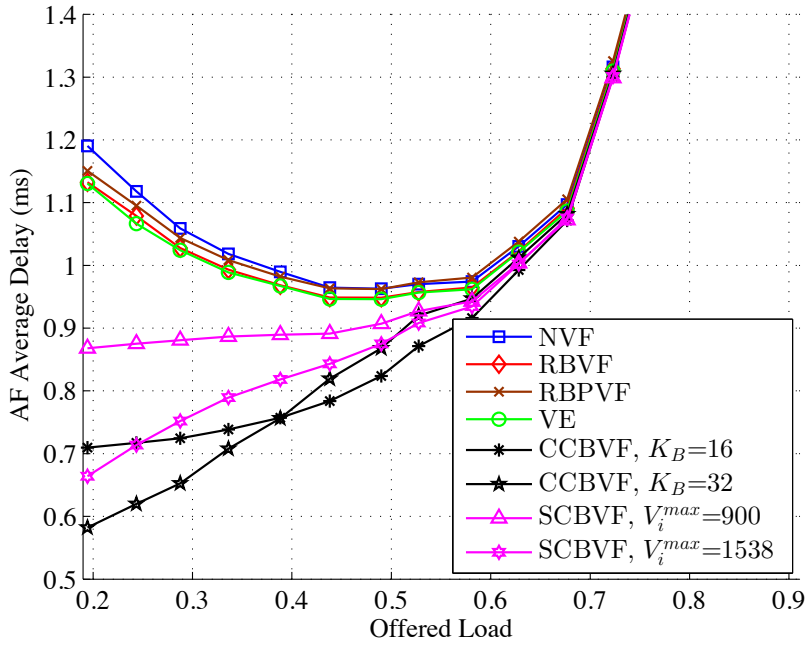


(a)

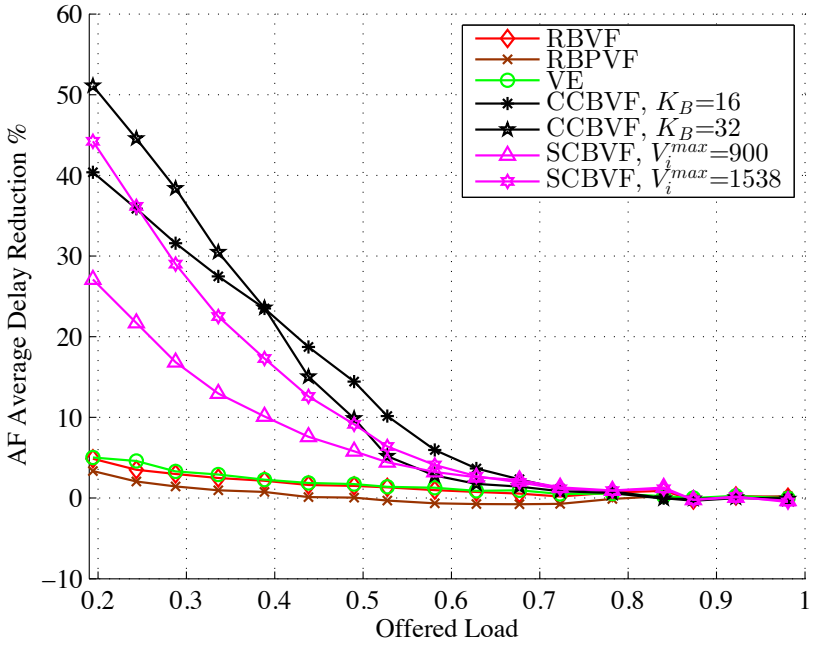


(b)

Figure 7.17: EF delay performance (D_i = 80-100 Km) (a) Average delay (b) Average delay reduction.

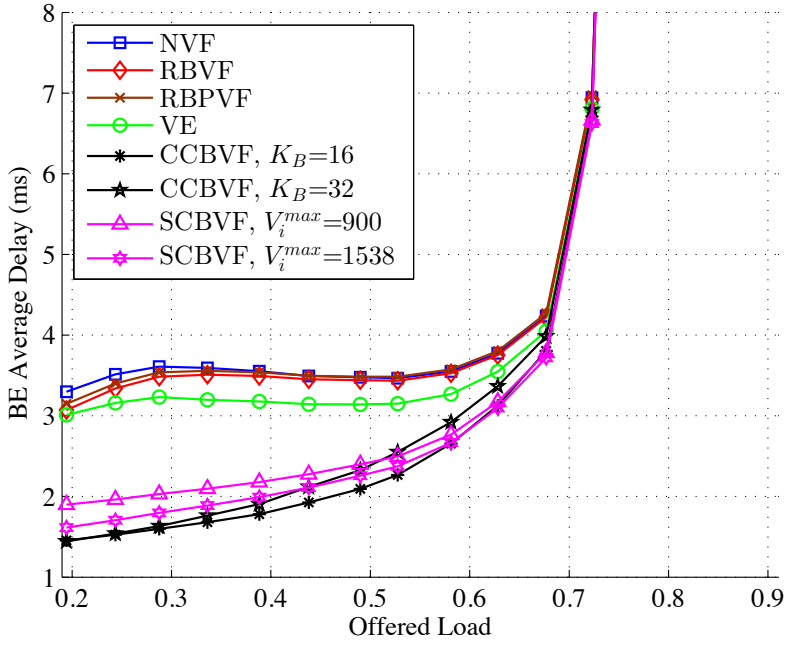


(a)

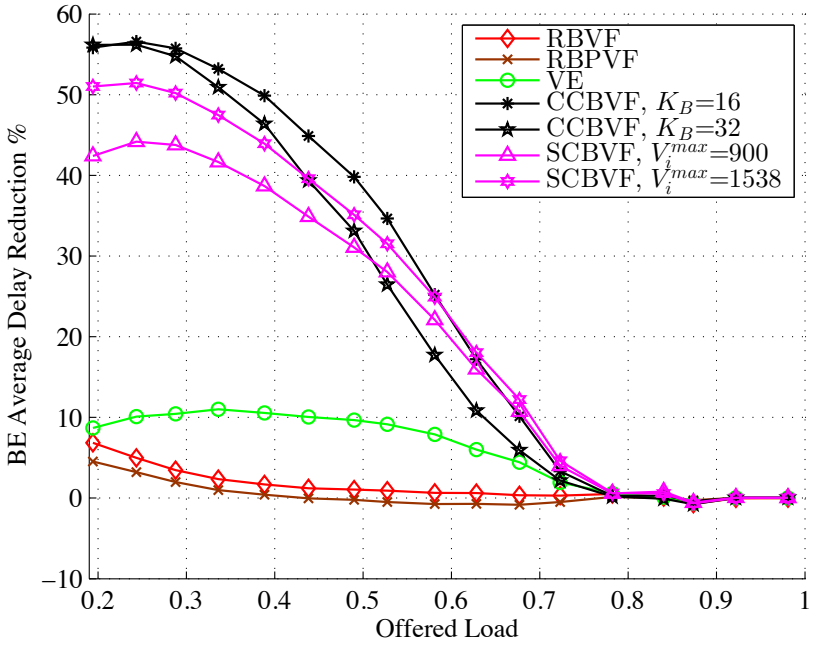


(b)

Figure 7.18: AF delay performance (D_i = 80-100 Km) (a) Average delay (b) Average delay reduction.



(a)



(b)

Figure 7.19: BE delay performance (D_i = 80-100 Km) (a) Average delay (b) Average delay reduction.

It is also important to highlight that NVF, RBVF, RBPVF, and VE suffer from light load penalty phenomena [18, 29] for all traffic classes. The light load penalty phenomena is that the average delay decreases with the load increase at lower load range. As the load becomes moderate, this phenomena disappears. Both CCBVF and SCBVF are less affected by this phenomena for relatively lower VBG size.

7.7 Summary

This chapter presented a new blind void filling approach, called Parallel Void Thread (PVT) for Long-Reach Ethernet Passive Optical Networks (LR-EPONs). PVT is a polling thread independent from the running DBA thread(s). When the OLT detects a void succeeding a bandwidth grants, PVT is invoked to allocate void based grant(s) VBG(s) to fill that void. PVT delay reduction mechanism is to enable ONUs to early transmit part of their upstream traffic during VBGs. This technique reduces the grant delays and enables bandwidth grants more frequently than both RBVF and RBPVF. There are three proposed VBG sizing schemes for PVT, namely VE, CCBVF, and SCBVF.

For single class LRD traffic, numerical results show that all of VE, CCBVF, and SCBVF outperform NVF, RBVF, and RBPVF and also achieve average delay below the 1.5 $\overline{RTT}$ bound. Regarding PVT, SCBVF is more efficient than CCBVF and VE but this comes with the expense of higher gate message load proliferation. Compared to SCBVF, CCBVF achieves less delay reduction but has very low gate message proliferation. VE has the least gate message proliferation as well as the least delay reduction. Regarding upstream report message proliferation, PVT does not add any additional control messages to the underlying DBA scheme. PVT performance mainly depends on how much traffic is transmitted during voids, the average delay of such traffic, and VBG utilization. Numerical results show that SCBVF has the highest VBG traffic ratio, while VE has the least one. For future research direction, K_B and V_i^{max} can be adapted to those parameters to achieve optimum

performance with continuous load variations.

For DiffServ traffic, Both CCBVF and SCBVF are able to reduce the time between consecutive grants to $\approx$ 0.2 ms compared to $\approx$ 0.9 ms for the other schemes. This has a great impact on delay reduction for EF, AF, and BE traffic classes respectively. The numerical results show that the smaller the VBG size is, the lower EF delay SCBVF and CCBVF achieve. On the other hand, reducing VBG size increases both AF and BE traffic delay. Numerical results show that SCBVF achieves delay reduction up to 80%, 45%, and 52% for EF, AF, and BE traffic classes. CCBVF achieves delay reduction up to 62%, 52%, and 57% for EF, AF, and BE traffic classes. VE achieves delay reduction up to 3%, 5%, and 11% for EF, AF, and BE traffic classes respectively.

PVT can be applied without any modification to hybrid time/wavelength division multiplexing PONs (TWDM-PONs) for static wavelength allocation. In our point of view, combining PVT with STP scheme is much better for three reasons. First, reducing overhead cost as MTP has its fixed overhead cost (see Chapter 6). Second, mathematical analysis and simulation results show that PVT has less average delay performance. Third, STP is more robust that MTP at high load due to its less overhead wasted bandwidth. Applying PVT for TWDM-PONs with dynamic wavelength allocation must be tied with the wavelength allocation algorithm and is discussed in the next chapter.

Chapter 8

An Integrated Dynamic Bandwidth Allocation Scheme for Long-Reach Hybrid TDM/WDM EPONs

8.1 Introduction

In this chapter, we investigate the integrability of our presented solutions in the previous chapters within the context of Long-Reach Hybrid TDM/WDM EPONs (LR-TWDM-EPONs) with dynamic wavelength assignment. Our goal is to design a well performing DBA scheme on all offered load range. The design space problems are frame delineation, excess bandwidth allocation, and void filling. The first two problems can be solved by employing Integrated bandwidth Allocation (IBA) presented in Chapter 6. IBA can be applied in TWDM-EPONs. The only difference is that IBA will employ FDES (see Chapter 3) instead of DES for the excess bandwidth allocation function. Regarding void filling, we present a Wavelength based Parallel Void Thread (WPVT). WPVT extends the idea of PVT (see Chapter 7) to enable blind void filling in TWDM-EPONs.

8.2 On-The-Fly Void Filling Challenges in LR-TWDM-EPONs

This section investigates the challenges of applying PVT in LR-TWDM-EPONS. Challenges are divided into two categories, namely void detection and void filling.

8.2.1 Void Detection

As we explained earlier in Chapter 7, on-the-fly void filling relies on detecting a void succeeding RBG grant. There are two main problems. First, what is the next node to be allocated an RBG? This problem happens because in TWDM-EPONs ONUs are not polled in cyclic order anymore. The second problem is how to detect a void in such WDM system? The ambiguity of void detection is because the next RBG associated channel is not known yet.

8.2.2 Void Filling

The second challenge is void filling process. This is because voids on different upstream wavelength channels might overlap in time as shown in Figure 8.1. Voids overlap leads to both RBG and VBG allocation complexities. We identified three different types of overlapping, namely VBG-RBG overlap, VBG-VBG overlap, and RBG-VBG overlap.

VBG-RBG overlap, shown in Figure 8.2, represents the case when VBG allocation overlaps with previously allocated RBG slot for the same ONU. The OLT cannot allow overlapped allocated grants for the same ONU because each ONU has only single tunable laser. Moreover, this leads to poor bandwidth utilization as the ONU will not be able to utilize the overlapping period on all channels.

Figure 8.3 shows the VBG-VBG overlap case. In this case, a new VBG is overlapping with a previous one. This is very common case especially at low load because of voids

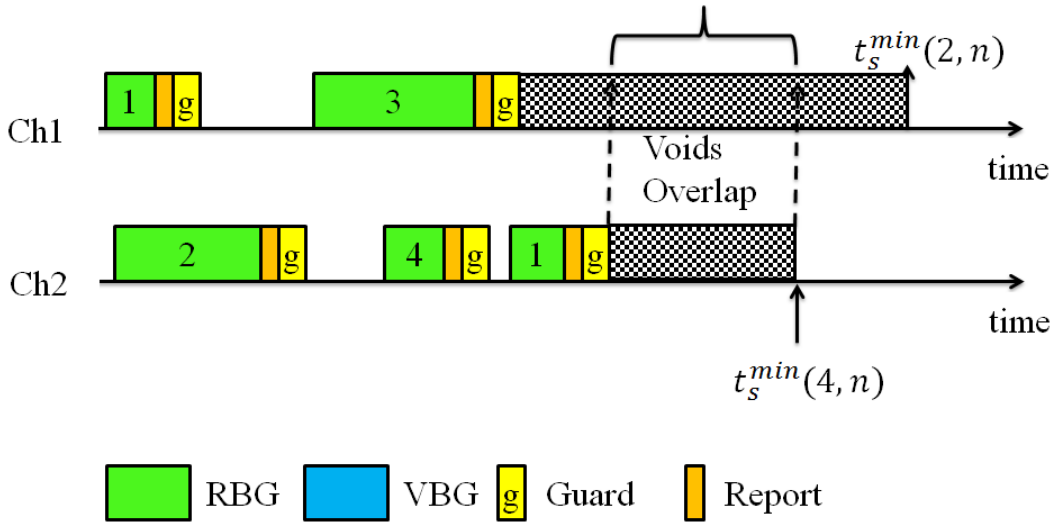


Figure 8.1: Voids overlap.

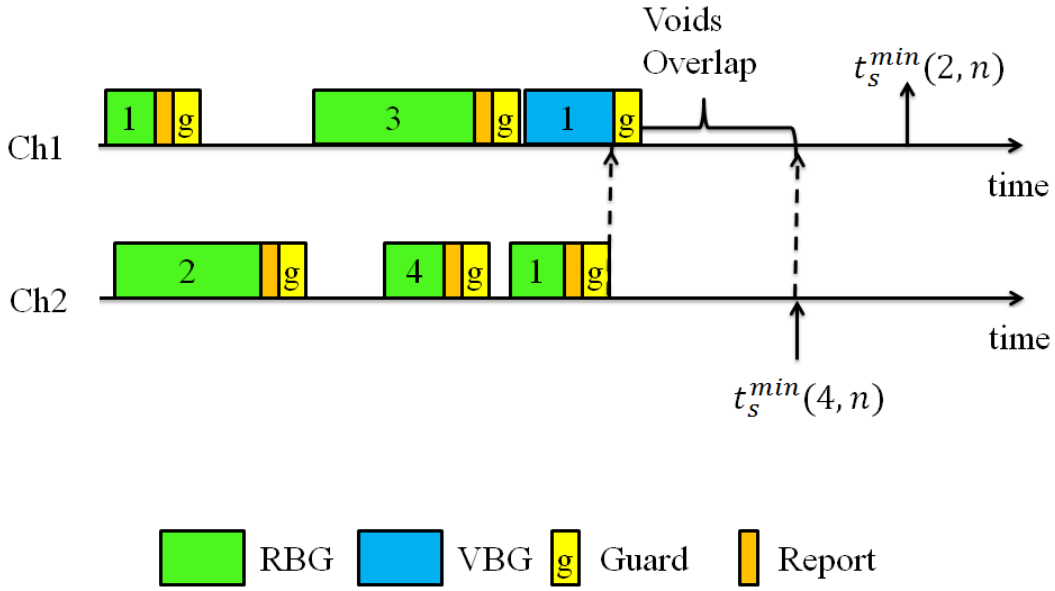


Figure 8.2: VBG-RBG overlap illustration.

availability. This problem shall complicate the VBG allocation process and some solutions might be of non-linear complexity.

The last type of void filling challenges is RBG-VBG overlap. Figure 8.4 shows that this case happens when an RBG allocation overlaps with previously allocated VBG. This

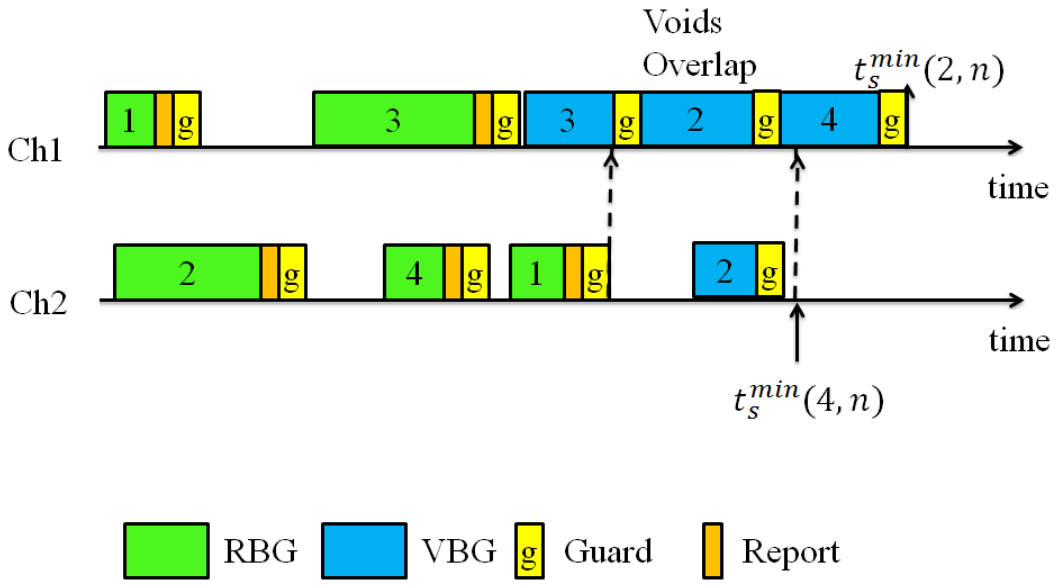


Figure 8.3: VBG-VBG overlap illustration.

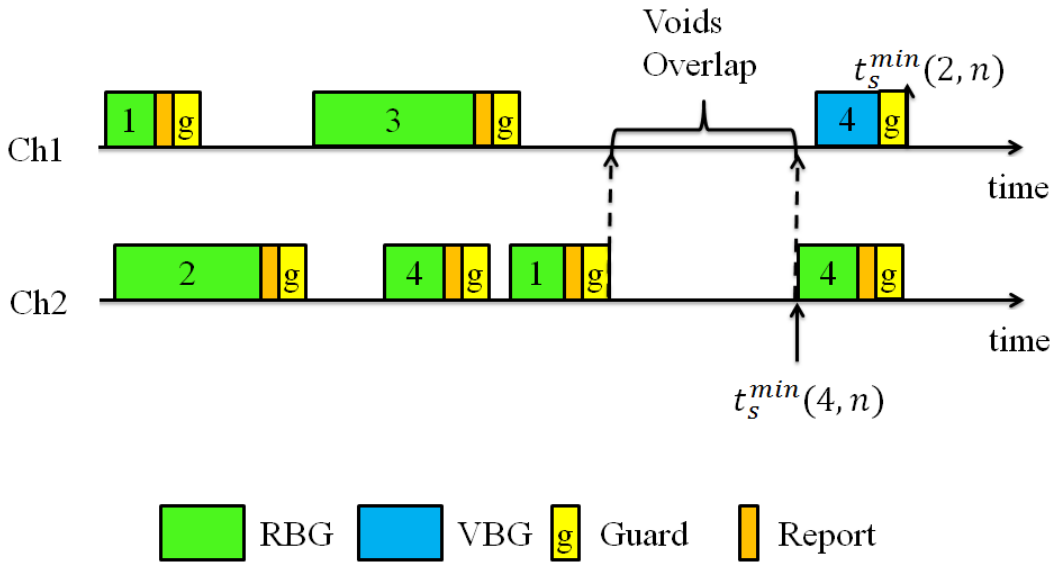


Figure 8.4: RBG-VBG overlap illustration.

RBG slot has to be shifted to start after the end of the VBG slot. This shift has two negative effects. First, the original thread (RBGs) faces additional delay due to VBG allocation. This violates PVT design goals as a plus function to the original thread and should not

negatively effect the average delay. Second, the shifting process leads to idle bandwidth and hence poor bandwidth utilization and increased delay.

8.3 Wavelength Based Parallel Void Thread (WPVT)

In this section, we describe Wavelength based Parallel Void Filling (WPVT) scheme. WPVT aims to detect the voids succeeding RBGs and to fill them with VBGs in order to better utilize bandwidth and early flush buffered packets. Unless mentioned explicitly, we use the same variable notations in Table 7.1.

8.3.1 WPVT Void Detection

As we mentioned earlier, void detection challenges in TWDM-EPONs are next RBG start detection and the associated upstream channel with it.

Regarding the next node to report, the OLT should maintain a list of the time it will receive the next report message, $t_{gate}(i, n)$ from each ONU. The next node will be the one with minimum $t_{gate}(i, n)$. Detection of the next node complexity is $O(N)$, where N is the number of ONUs. This complexity can be reduced to $O(logN)$ if ordered list of $t_{gate}(i, n)$ is used.

The second challenge, the wavelength allocated with the next RBG, cannot be addressed separately from the running wavelength assignment scheme. In [35], the authors presented a detailed study about wavelength assignment schemes. To the best of our knowledge, all proposed wavelength assignment schemes do not depend on the reported buffer size in their decision (considering single traffic class). They depend on the horizon time (see Chapter 7) of each channel. Consequently, upon allocation of an RBG slot, WPVT identifies the next node to report and invokes the wavelength assignment scheme to determine the wavelength channel allocated to the next RBG. After doing so, the OLT compares the minimum start time of the next node with the horizon time of the allocated wavelength

channel. If

$$t_s^{min}(i,n) - t_h(w) > t_g + t_c, \tag{8.1}$$

where i is the next node index and $t_h(w)$ is the horizon time of channel w, then the OLT detects a void, $V(i,n)$, on channel w with start time of

$$V_s(i,n) = t_h(w), \tag{8.2}$$

and end time of

$$V_e(i,n) = t_s^{min}(i,n). \tag{8.3}$$

It is important to note that void detection and void filling should be done on channel by channel and RBG by RBG basis. Otherwise, filling voids at all channel at once leads to delaying the start of next RBG(s).

8.3.2 WPVT Void Filling

WPVT void filling challenges is not easy as TDM case. This is because the overlapping problems mentioned in Section 8.2. The problems we described might have variety of solutions. However, WPVT is designed to maintain two gaols. First, never delay RBG allocation. Second, reduce the complexity of VBG allocation and maximizing void allocation as possible. Both Void Extension (VE) and Count Controlled Batch Void Filling (CCBVF) cannot be used with WPVT due to the nature of the identified overlapping challenges.

We present Wavelength based Size Controlled Batch Void Filling (WSCBVF) as a void filling scheme of WPVT. The relative advantage of WSCBVF (inherited from SCBVF) is it allocates VBGs one by one. Hence, it is much easier to detect and resolve grant overlapping. Upon detecting a void, WSCBVF polls an ONU independent from the basic running thread and identifies its proposed VBG start ($t_s^{VBG}(i)$) and end times ($t_e^{VBG}(i)$).

In order to resolve the identified overlapping cases, three conditions must be met. To

resolve VBG-RBG overlap,

$$t_s^{VBG}(i) > t_e(i,n) \tag{8.4}$$

or

$$t_e^{VBG}(i) < t_s(i,n) \tag{8.5}$$

These conditions ensures that a VBG never overlaps with previously allocated RBG slot. In order to resolve RBG-VBG overlap, the OLT should maintain the following condition:

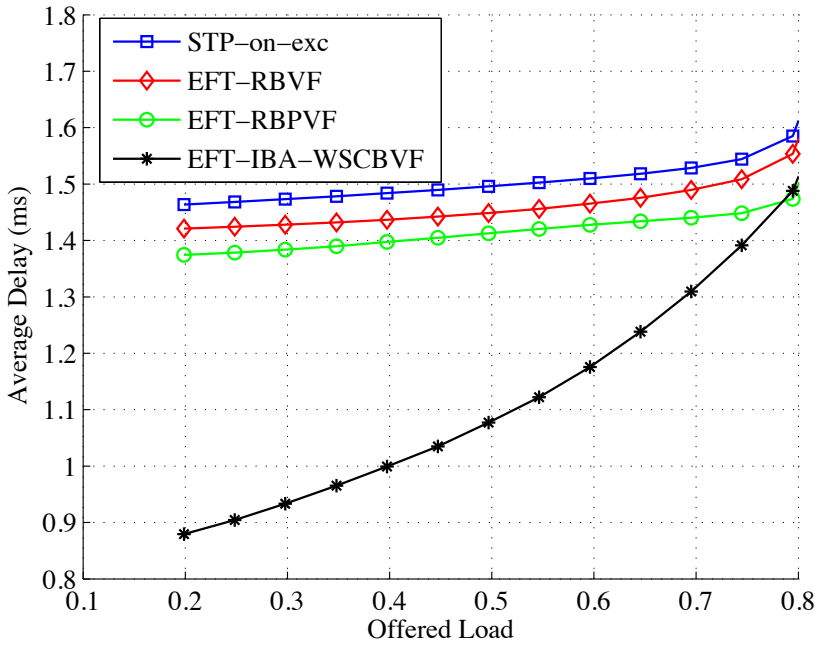
$$t_e^{VBG}(i) < t_s^{min}(i,n). \tag{8.6}$$

This conditions ensures that the intended VBG allocation ends before the potential start of the next RBG slot of that particular ONU. This condition is necessary to keep WPVT as a plus function to the running original DBA thread. In order to resolve VBG-VBG overlap, we used fixed wavelength assignment for VBGs. By using this we canceled this overlapping type completely. The other dynamic wavelength assignment solutions have high complexity ($O(N^2)$). Moreover, these solutions might lead to some unallocated voids as a result of resolving overlapping. From our point of view, this solution is very competitive as long as the basic thread applies wavelength allocation scheme that ensure load balancing on all wavelength channels.

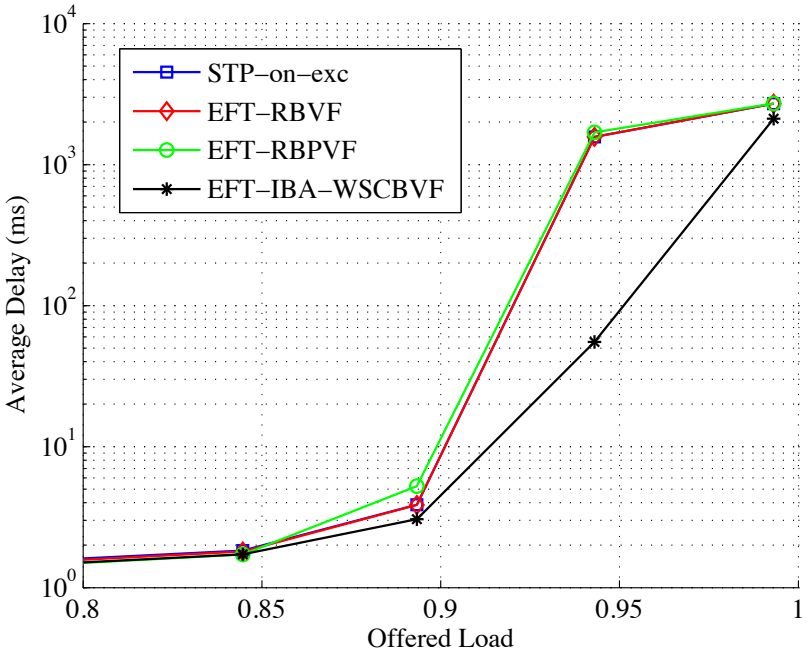
8.4 IBA-WSCBVF Performance Evaluation

We consider an EPON with single OLT and 128 ONUs with 10 MB buffer size with four upstream channels. The upstream and the downstream transmission rates are symmetric with 1 Gbps. Ethernet frames size ranges from 64 to 1518 B, (we used the packet size distribution reported in [11], with minimum inter-frame-gap (IFG) of 12 B and preamble of 8 B). The incoming traffic is self-similar traffic with long range dependence (LRD) and Hurst parameter 0.8 and packets are served in first-come-first-served (FCFS) order. T_{cycle}^{max} is

set to 4 ms, t_g is set to 1 μs and G_i^{max} = 15500 B. The offered load is distributed uniformly over ONUs.



(a)



(b)

Figure 8.5: Average packet delay comparison (a) low load (b) high load.

We combine IBA with WSCBVF (IBA-WSCBVF) and compare it against EFT-VF [35] (referred to as EFT-RBVF) and Earliest Finishing Time with Partial Void Filling (EFT-PVF) [38] (referred to as EFT-RBPVF). We choose STP-on-exc [4] as the underlying DBA scheme for EFT-RBVF and EFT-RBPVF with excess pool bound of NG_i^{max} bytes. IBA-WSCBVF employs Earliest Finishing Time (EFT) [35] due to its reported efficiency. Regarding VBG allocation, we choose to use distance-based-grouping (DBG) [35] as static wavelength allocation.

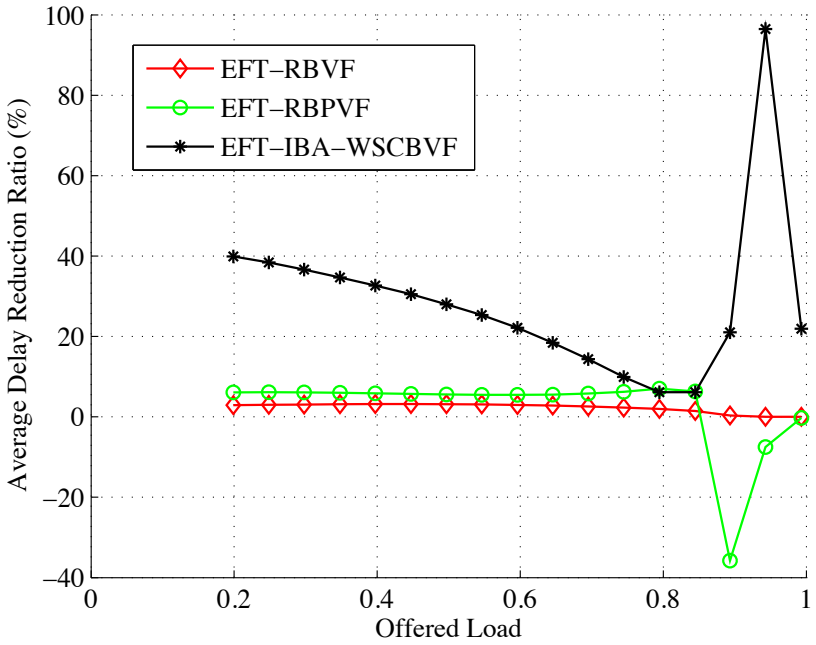
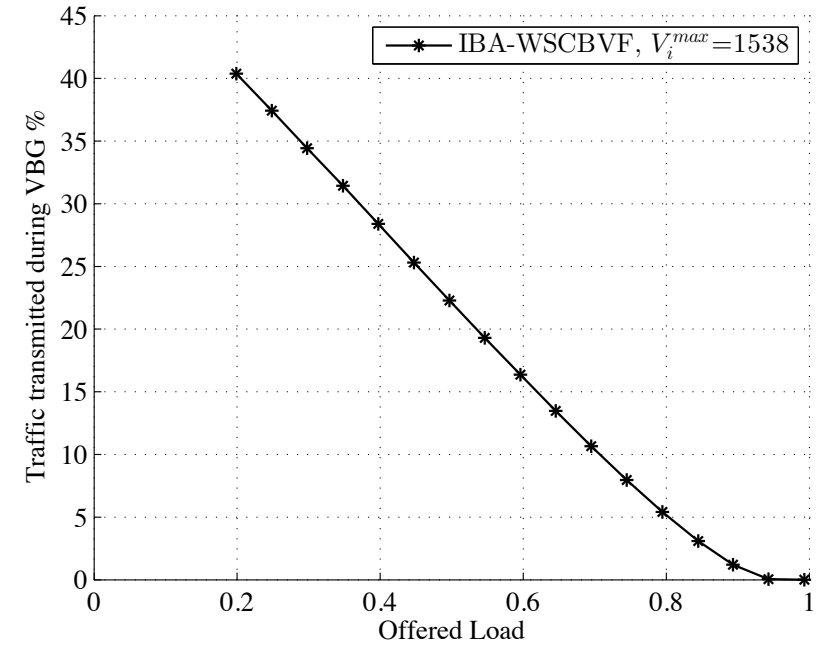
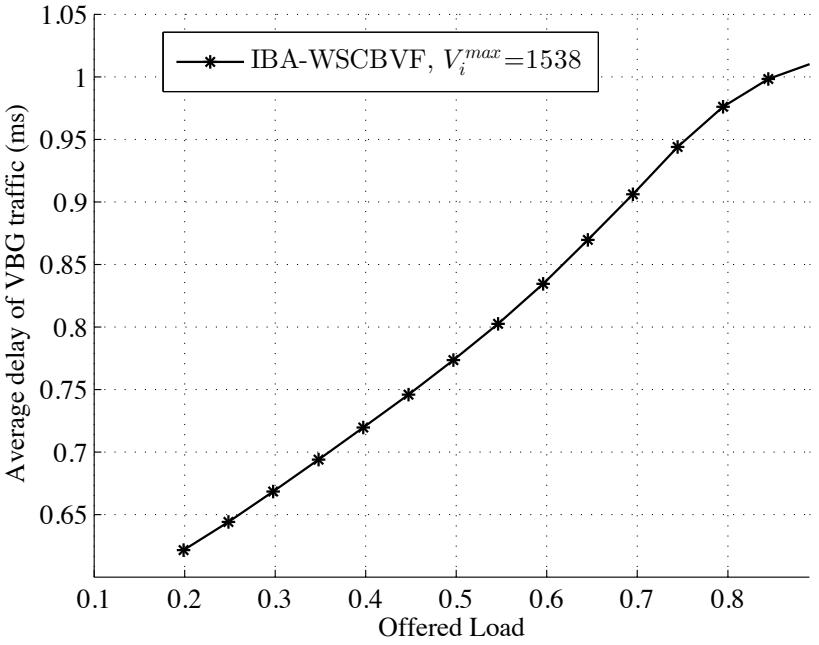


Figure 8.6: Average packet delay reduction.

Figure 8.5 shows the average packet delay of the simulated schemes. EFT-IBA-WSCBVF outperforms STP-on-exc, EFT-RBVF, and EFT-RBPVF at both low and high load. Figure 8.6 shows the average delay reduction with respect to STP-on-exc. EFT-RBPVF outperforms EFT-VF at low and medium load. At high load, EFT-RBPVF becomes unstable and achieve negative delay reduction up to 40%. Performance excel of EFT-IBA-WSCBVF can be explained based on the load range. At low and medium load, WSCBVF works to allocate VBG slots and reduces packet delay according to the mechanism explained in

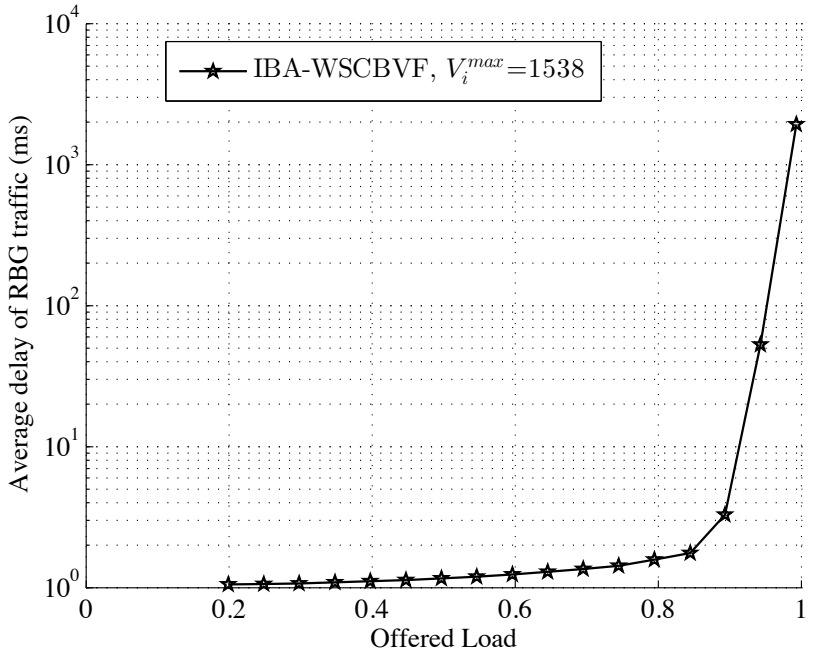


(a)

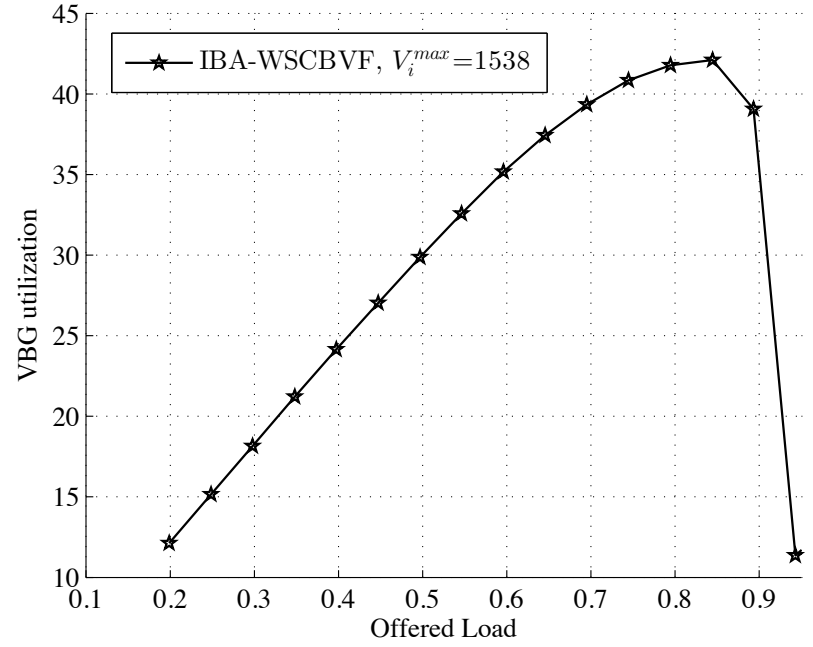


(b)

Figure 8.7: WSCBVF (D_i = 80-100 Km) (a) Traffic % transmitted during VBG (b) Delay of transmitted packets during VBGs.



(a)



(b)

Figure 8.8: WSCBVF (D_i = 80-100 Km) (a) Delay of transmitted packets during RBG (b) VBG utilization.

Chapter 7. It can be seen that the delay reduction decreases with load increase as the available void size reduces. At high load, WSCBVF contribution reduces and IBA takes over to achieve high reduction up to 95% (similar to EGSIP performance).

In depth analysis of WSCBVF performance is explained in Figure 8.7. It shows the traffic percent transmitted in VBGs and the average delay of this traffic. At load below 0.8, WSCBVF transmits 40-5% of the traffic in VBGs with average delay ranges from 0.55-.95 ms. This explains why WSCBVF is capable of achieving significant delay reduction at low and medium load.

The average RBG traffic delay and VBG utilization are shown in Figure 8.8. At load below 0.7, the average RBG traffic is below the 1.5 RTT_{max} bound. This is because the large amount of traffic directed to VBGs left much room in RBGs to accommodate newly arrived packets to be transmitted without being reported first. Figure 8.8 also shows that VBG utilization increases with offered load till high load. At high load, voids becomes smaller and are not be able to accommodate large Ethernet frames.

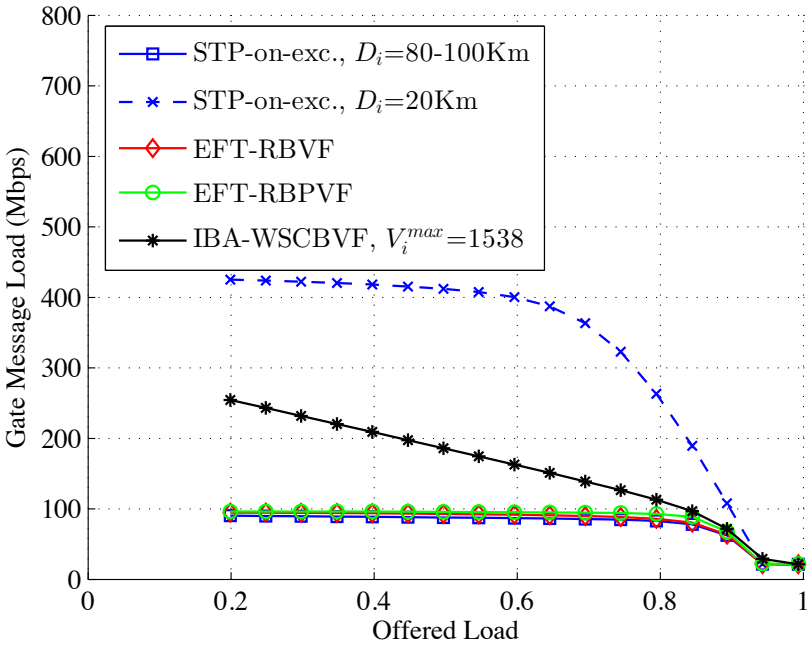


Figure 8.9: Gate message proliferation, D_i = 80-100 Km.

Figure 8.9 shows the gate message overhead comparison. Both EFT-RBPVF and EFT-RBVF slightly increase the control overhead. EFT-IBA-WSCBVF increases the control message overhead by 150% compared to STP-on-exc with D_i = 80-100 Km. However it is less that the control message overhead for short reach case (D_i = 20 Km)

8.5 Summary

This chapter investigated the integrability of some of our solutions presented in this dissertation. We presented a multi-feature DBA scheme, namely IBA-WSCBVF. Our scheme combines void filling, excess bandwidth allocation, and frame delineation. IBA-WSCBVF solves all these challenges within the context of LR-TWDM-EPONs. IBA achieves significant delay reduction ratio against its peers. Unlike MTP behavior in TWDM-EPONs [38], IBA-WSCBVF is robust at medium and high load. The control overhead cost is only in the downstream direction (gate messages). Moreover, it is not fixed as MTP as it decreases with offered load increase. At high load, IBA-WSCBVF almost converges to IBA and achieves notable delay reduction.

Chapter 9

Sleep-time Sizing and Scheduling in Green Passive Optical Networks

9.1 Introduction

Energy consumption is emerging as a major challenge in today's information and communication technologies. The expansion of data communication usage in different applications has resulted in a tremendous increase in energy consumption worldwide. According to recent statistics, the energy cost to operate wireless devices accounts for half of the operating expenses for any wireless deployment. Currently 3% of the world-wide energy is consumed by the information and communication technology infrastructures that cause about 2% of the world-wide CO_2 emissions. This figure is comparable to the worldwide CO_2 emissions by airplanes or 25% of the world-wide CO_2 emissions caused by cars. Therefore, making communication devices energy efficient will not only benefit the environment, but also increase the profitability of business for communication companies.

In order to fulfill the bandwidth-thirsty demands and achieve the best operational efficiency, we are experiencing fast development of fiber-wireless (FiWi) access technologies [71, 72, 73, 74, 75], which are envisioned to serve as the enabling technologies for the next-generation metropolitan-area networks. Such infrastructure aims to take the advantage of an integrated design of the optical and broadband wireless access systems in both

phases of network planning/dimensioning and real-time operations.

Passive optical networks (PONs) are introduced as a promising approach for solving the first mile problem in broadband access networks [15]. The structure of PON consists of an optical line Terminal (OLT), located at a central office (CO) that is connected to a set of optical network units (ONUs) located at the end-user locations [1]. In the past decade, PON has been a subject of extensive study concerning quality of service (QoS), dynamic bandwidth allocation (DBA), and recently energy saving mechanisms.

In our study, we present a simplified, yet effective, novel PON scheme that supports energy saving. Our analytical and simulation results show that our scheme saves energy without introducing significant delay to the network traffic. The main advantage of our scheme is that the OLT allocates the bandwidth for each ONU considering the downstream buffer size. In other words, the ONU remains in the sleep mode until its downstream buffer reaches a certain threshold. The OLT uses a separate control traffic plane to wake-up the ONU. There are several ways to implement the proposed control plane. For example, we may use a dedicated wavelength or use the Internet. The latter design can be easily done by adding a network interface to the public Internet at both the OLT and The ONU. This interface is used by the OLT to awake the ONU. We claim, to the best of our knowledge, that this work is the first to successfully decouple the average traffic delay from the sleeping time portion. In other words the average delay is only dependent on the QoS constraints, while the sleeping time portion only depends on the traffic load.

9.2 Sleep Sizing Framework

In this section, we provide an overview of our proposed sleep sizing scheme [76]. This includes the signaling mechanism between the OLT and ONUs.

9.2.1 Overview

Our scheme uses conventional PON tree topology and assumes that the OLT and ONU signaling messages are exchanged over a separate control plane. Also, we assume that the downstream traffic is larger than the upstream one, hence the ONU wake-up by OLT according to the downstream queue size threshold. Upon reaching certain threshold, OLT uses the control plane to wake-ups the ONU for serving both downstream and upstream traffic. Two classes of packets are received by the ONUs called control and data packets. Control packets are assumed to be delay intolerant, while data packets can tolerate certain average delay constraint $\mathbb{D}_{QoS}$. Thus, control packets are assumed to be of higher priority than data packets. Furthermore, the queuing discipline within each class is assumed to be FIFO, while the service prioritization between classes is based on non-preemptive service. Moreover, the ONU is allowed to sleep for energy saving purposes. Initially, ONU remains in the sleep mode until either a control packet arrives or data queue size reaches certain threshold Q_{th}, which depends on the average delay constraint $\mathbb{D}_{QoS}$ in addition to the arrival and service parameters. This is known to be an N-threshold policy queuing system [77]. Later in this section, we obtain a closed form solution for the average sleeping time duration and the percentage of time ONU spends in the sleeping mode to obtain the energy saving gain. Also, we obtain a numerical solution for the data queue threshold Q_{th} computed in terms of arrival rate, service rate, and average delay constraint. In this study, we limit our investigation to a single ONU due to problem complexity. The actual implementation of this scheme can be carried out by modifying the current design of both OLT and ONU to add the capability of separate control plane.

9.2.2 System Parameters

Arrival and service processes

Control and data arrival processes are assumed to be independent Poisson processes of rates, λ_c and λ_d, respectively. Their service is assumed to be generally distributed. The mean transmission time for control and data packets is given by $\overline{X_c}$ and $\overline{X_d}$ respectively. Their corresponding reciprocals are given respectively by μ_c and μ_d denoting the service rates. The second moments of the service time for control and data packets are given by $\overline{X_c^2}$ and $\overline{X_d^2}$, respectively.

Data packets delay constraint

The framework assumes that data packets can tolerate an average delay constraint $\mathbb{D}_{QoS}$. This allows the ONU to sleep as long as there is no control packets to serve and the data queue size is below the threshold.

9.2.3 Notation

Each ONU sleeping cycle is composed of two successive periods, where it begins by a sleeping period for random duration T_s followed by an active period for random duration T_a. The corresponding averages are denoted by $\overline{T_s}$ and $\overline{T_a}$, respectively. Their probability distributions are denoted by f_s and f_a, while F_s and F_a are the cumulative distributions. When needed, $\overline{F_i}$ would denote the complementary cumulative density function associated with F_i. Table 9.1 summarizes the notation used in our analytical model.

Table 9.1: Notation.

λ_i	Poisson arrival rate of class $i \in \{c, d\}$
λ_{tot}	Sum of control and data arrival rates = $\lambda_c + \lambda_d$
X_i	Random variable for service time of class $i \in \{c, d\}$
$\overline{X_i}$	Mean service time of class $i \in \{c, d\}$
$\overline{X_i^2}$	Second moment of service time of class $i \in \{c, d\}$
$\overline{W_d}$	Average waiting time of data packets in the queue
$\overline{D_d}$	Average arrival-to-departure data packet delay
μ_i	Service rate of class i = $\frac{1}{\overline{X_i}}$
ρ_i	Utilization of class i = $\frac{\lambda_i}{\mu_i}$
$\mathbb{D}_{QoS}$	Average delay constraint for data packets
$Q_i(t)$	Queue length of class i at time t
Q_{th}	Data queue threshold depending on the delay constraint
T_s	ONU sleep period duration random variable
T_a	ONU active period duration random variable
$\overline{T_s}$	Average ONU sleeping time
f_s	Probability distribution of sleeping time
f_a	Probability distribution of active time
F_s	Cumulative distribution of sleeping time
F_a	Cumulative distribution of active time
$\overline{F_a}$	Complementary cumulative distribution of active time
$S(t)$	state of ONU at time t (0: sleep, 1: active)

9.3 Analytical Model

9.3.1 Average sleeping time

In this section, we derive an expression for the probability distribution of the ONU sleeping time. Once the distribution is derived, we accordingly present a closed form expression for the average ONU sleeping time as a function of the data queue threshold and the arrival rates of data and control packets. The probability distribution is derived in three phases based on the independence of the traffic arrival processes of control and data packets. First, we consider that the system is serving data packets only. Second, we consider that the system is serving control packets only. Finally, we combine the two processes to get an expression for the distribution of ONU sleeping time with both arrivals allowed. For each of the above system phases, we present the PDF and CDF of the sleeping time. Accordingly, we average the obtained distribution to get the ONU average sleeping time $\overline{T_s}$.

System with data packet arrivals

In the absence of control packet arrivals, the ONU remains sleeping until the arrival of certain number of data packets equal to Q_{th}. Since the arrival process is assumed to be Poisson distributed with rate λ_d, the inter-arrival time distribution is exponentially distributed with mean $\frac{1}{\lambda_d}$. In this case, the time required to reach Q_{th} packets in the queue for the ONU to wake-up is equal to the sum of Q_{th} i.i.d. exponential random variable with mean $\frac{1}{\lambda_d}$ (*i.e.,* average inter-arrival time). Thus, the sleeping time has an Erlang distribution that is given by,

$$f_{(s,d)}(t; Q_{th}, \lambda_d) = \frac{\lambda_d^{Q_{th}} t^{Q_{th}-1} e^{-\lambda_d t}}{(Q_{th}-1)!} \tag{9.1}$$

while the corresponding cumulative density function is given by,

$$F_{(s,d)}(t; Q_{th}, \lambda_d) = 1 - \sum_{n=0}^{Q_{th}-1} \frac{e^{-\lambda_d t}(\lambda_d t)^n}{n!} \tag{9.2}$$

System with control packet arrivals

In the absence of data packet arrivals, the allowable time for an ONU to sleep is the inter-arrival time of control packets which is exponentially distributed with mean $\frac{1}{\lambda_c}$, assuming the service time is very small compared to the inter-arrival time[1]. Thus, the probability density function is given by

$$f_{(s,c)}(t; \lambda_c) = \lambda_c e^{-\lambda_c t} \tag{9.3}$$

while the corresponding cumulative density function is given by

$$F_{(s,c)}(t; \lambda_c) = 1 - e^{-\lambda_c t} \tag{9.4}$$

[1]The arrival rate of control packets is expected to be small leading to relatively long inter-arrival time.

System with both control and data packet arrivals

Since the ONU does not go to sleep mode except when there is no packets in the queue, we consider the start of the sleeping cycle as a reference point in the following analysis (*i.e.,* $t = 0$ at the beginning of the sleep cycle). Thus, $Q_c(0) = Q_d(0) = 0$.

Sleeping time can be equal to T_s in two cases only. First case occurs when there is no control arrivals $\forall t \in [0, T_s]$ while the Q_{th} data packet arrives at $t = T_s$. The second case occurs when the data queue size is less than Q_{th} $\forall t \in [0, T_s]$ while a control packet arrives at $t = T_s$. This can be expressed using the previously derived PDF and CDF expressions. Using the independence between the two processes in which the joint distributions become the product of the marginal ones is shown below,

$$\begin{aligned} f_s(T_s) &= f_{(s,d)}(T_s)\overline{F_{(s,c)}}(T_s) + f_{(s,c)}(T_s)\overline{F_{(s,d)}}(T_s) \\ &= \frac{\lambda_d^{Q_{th}} T_s^{Q_{th}-1} e^{-\lambda_d T_s}}{(Q_{th}-1)!} e^{-\lambda_c T_s} \\ &\quad + \lambda_c e^{-\lambda_c T_s} \sum_{n=0}^{Q_{th}-1} \frac{e^{-\lambda_d T_s}(\lambda_d T_s)^n}{n!} \\ &= \lambda_d^{Q_{th}} \frac{T_s^{Q_{th}-1} e^{-\lambda_{tot} T_s}}{(Q_{th}-1)!} \\ &\quad + \sum_{n=0}^{Q_{th}-1} \lambda_c \lambda_d^n \frac{T_s^n e^{-\lambda_{tot} T_s}}{n!} \end{aligned} \tag{9.5}$$

$$\begin{aligned} \overline{T_s} &= \int_0^\infty t f_s(t) dt \\ &= \frac{\lambda_d^{Q_{th}}}{(Q_{th}-1)!} \int_0^\infty t^{Q_{th}} e^{-\lambda_{tot} t} dt \\ &\quad + \sum_{n=0}^{Q_{th}-1} \frac{\lambda_c \lambda_d^n}{n!} \int_0^\infty t^{n+1} e^{-\lambda_{tot} t} dt \end{aligned} \tag{9.6}$$

By changing the variables and using Gamma function, we obtain the above integral to be

equal to,

$$\int_0^{\infty} t^{n+1} e^{-\lambda_{tot} t} dt \;\; = \frac{(n+1)!}{\lambda_{tot}^{n+2}} \tag{9.7}$$

Thus, the average sleeping time is given by the closed form expression as follows:

$$\begin{aligned} \overline{T_s} &= Q_{th} \frac{\lambda_d^{Q_{th}}}{\lambda_{tot}^{Q_{th}+1}} + \sum_{n=0}^{Q_{th}-1} \frac{(n+1)\lambda_c \lambda_d^n}{\lambda_{tot}^{n+2}} \\ &= \frac{1 - (\frac{\lambda_d}{\lambda_c+\lambda_d})^{Q_{th}}}{\lambda_c} \end{aligned} \tag{9.8}$$

From the above, we notice that the average sleeping time is a function of the control and data packet arrival rates and the allowable data queue threshold. However, we still need to find the average active time in order to know the percentage of time the ONU is allowed to sleep without degrading QoS.

We denote the average sleeping time by $\overline{T_s}$ and the average number of data and control arrivals within this period is given by $\lambda_d \overline{T_s}$ and $\lambda_c \overline{T_s}$. Within the active period $\overline{T_a}$, the expected number of data and control arrivals is equal to $\lambda_c \overline{T_a}$ and $\lambda_c \overline{T_a}$, respectively. Each data packet is served with an average service time $\overline{X_d}$, while the control packets have an average service time of $\overline{X_c}$. Hence, the ONU should remain active until the queue is empty. This can be expressed as follows:

$$\begin{aligned} \overline{T_a} &= \lambda_c \overline{T_s X_c} + \lambda_d \overline{T_s X_d} + \lambda_c \overline{T_a X_c} + \lambda_d \overline{T_a X_d} \\ &= (\rho_c + \rho_d)\overline{T_s} + (\rho_c + \rho_d)\overline{T_a} \end{aligned}$$

Thus,

$$\overline{T_a} \;\; = \frac{\rho_c + \rho_d}{1 - \rho_c - \rho_d} \overline{T_s} \tag{9.9}$$

We write the equation (9.9) in the following alternative form,

$$\frac{\overline{T_a}}{\overline{T_a} + \overline{T_s}} = \rho_c + \rho_d \tag{9.10}$$

This form states that the ONU is only active for a fraction of time, which is equal to the introduced aggregate load. This fact is one of the main advantages of our framework since it shows that we successfully decouple the sleeping time from the QoS constraints. In other words, the average sleeping period depends only on the traffic load. Using (9.5), we can evaluate the second moment of sleeping time as follows:

$$\overline{T_s^2} = \frac{2\big[1 - (Q_{th}+1)(\frac{\lambda_d}{\lambda_{tot}})^{Q_{th}} + Q_{th}(\frac{\lambda_d}{\lambda_{tot}})^{Q_{th}+1}\big]}{\lambda_c^2} \tag{9.11}$$

9.3.2 Average delay of data packets

In order to calculate the data packets end-to-end delay, ONUs are modeled using N-threshold policy M/G/1 queue [77] with multi-class traffic. For simplicity, we assume that the sleeping time is independent of the arrival process and follows the distribution in (9.5). This approximation converts our model to M/G/1 non-preemptive priority queue with server vacations [78].

The average remaining vacation (sleeping) time can be evaluated as,

$$\overline{R_v} = \frac{\overline{T_s^2}}{2\overline{T_s}}, \tag{9.12}$$

while the average remaining service time is,

$$\overline{R} = \rho_c \overline{r_c} + \rho_d \overline{r_d}, \tag{9.13}$$

where $\overline{r_c} = \frac{\overline{X_c^2}}{2\overline{X_c}}$ and $\overline{r_d} = \frac{\overline{X_d^2}}{2\overline{X_d}}$. We proceed by computing the average waiting time spent

in buffer for data packets as follows:

$$\overline{W_d} = \frac{\overline{R} + (1 - \rho_c - \rho_d)\overline{R_v}}{(1 - \rho_c)(1 - \rho_c - \rho_d)} \tag{9.14}$$

Finally, the average delay of data packets is,

$$\overline{D_d} = \overline{X_d} + \overline{W_d} \tag{9.15}$$

With the average QoS delay in hand, we can compute Q_{th} from (9.8), (9.11), and (9.12) numerically.

9.4 Simulation Results

In this section, we show the results obtained from both the analytical model and the simulation. Several parameters are fixed for the simulation part, including the service rate and the simulation time. The data packet is assumed to be exponentially distributed with average size of 1000 B. The control packet size is assumed to be exponentially distributed with average size of 100 B. The link rate is set to 1 Gbps. Accordingly, the average service time for the data and control packets is $\overline{X_d}$ = 1μs and $\overline{X_c}$=0.1 μs, respectively.

Figure 9.1 shows the data packet delay vs. ρ_d with different values of λ_c. The threshold for switching from sleeping to active mode (*i.e.,* Q_{th}) is set to 20. For lightly loaded ONU, the delay is more dependent on control packets load rather data packets load. The greater control traffic results in lower data packets delay. This is because the sleeping time depends mostly on control traffic rather than data traffic. As the data load increases, higher control traffic results into higher data packets delay since data packets has lower priority than control packets. The impact of ρ_d and ρ_c on the data packets delay is shown in Figs. 9.2 and 9.3. Figure 9.2 shows that data packets delay linearly increase with Q_{th} at small control traffic. On the other hand, in Figure 9.3 the data packets delay tends

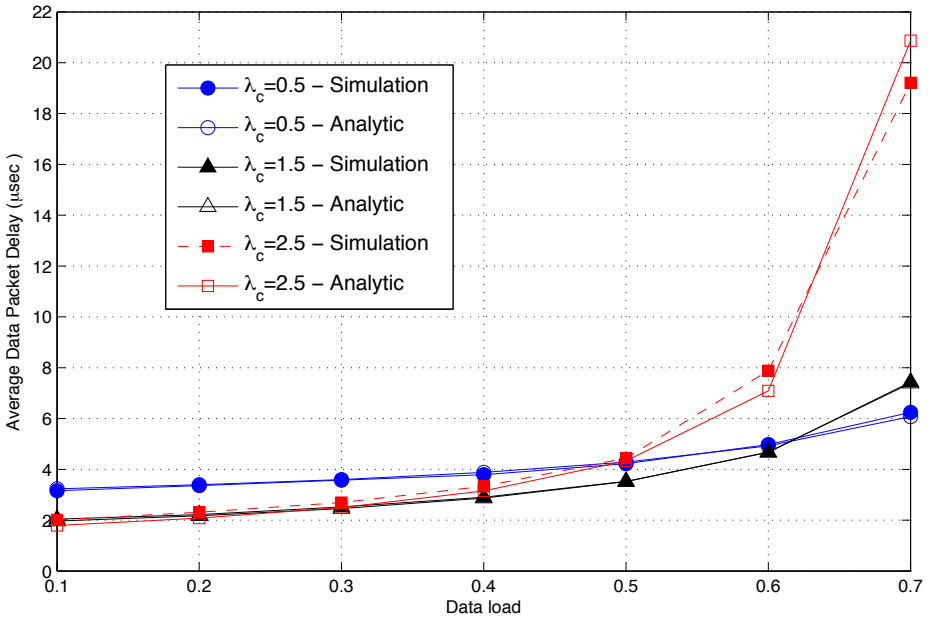


Figure 9.1: Average data packet delay vs. data load for $Q_{th} = 20$.

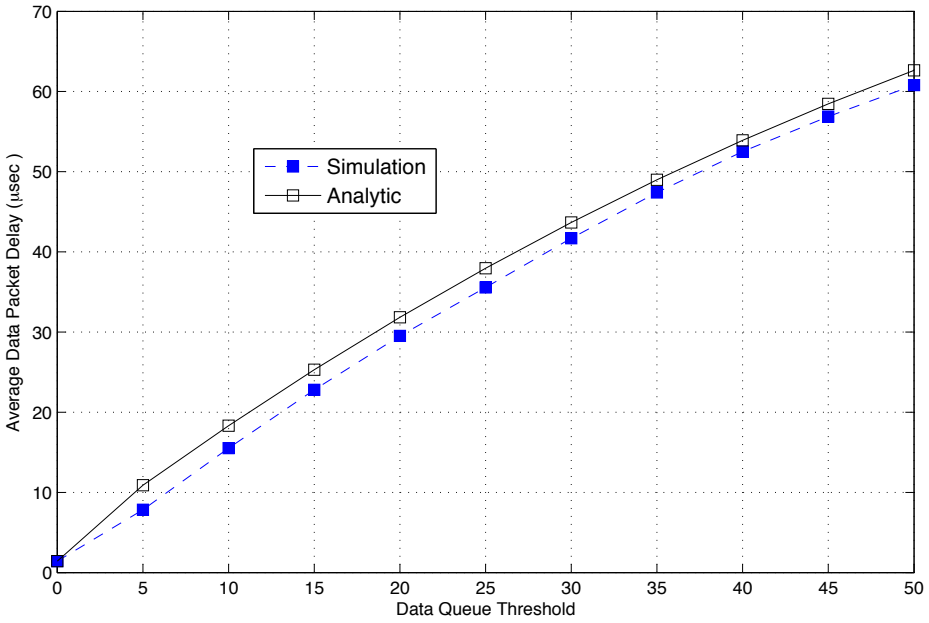


Figure 9.2: Average data packet delay vs. Q_{th} for $\lambda_c = 0.01 \times 10^6$.

to saturate with the increase of Q_{th}. The reason for such behavior is that the arrival of control packets triggers the ONU to active mode before reaching Q_{th}. These figures show how important it is to estimate the data queue threshold, as underestimation will result in energy waste, while overestimation might violate the QoS constraints. In order to examine

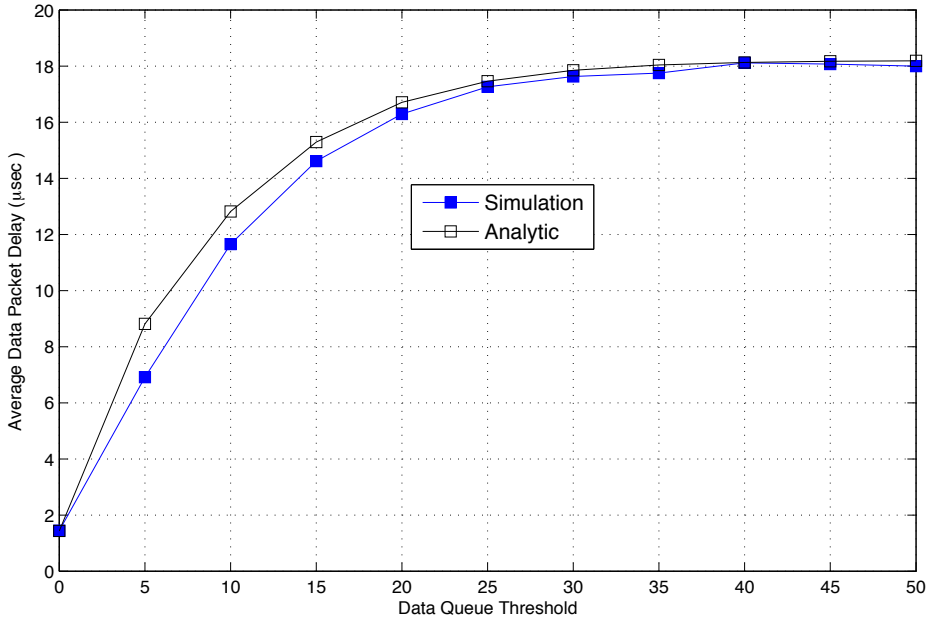


Figure 9.3: Average data packet delay vs. Q_{th} for $\lambda_c = 0.06 \times 10^6$.

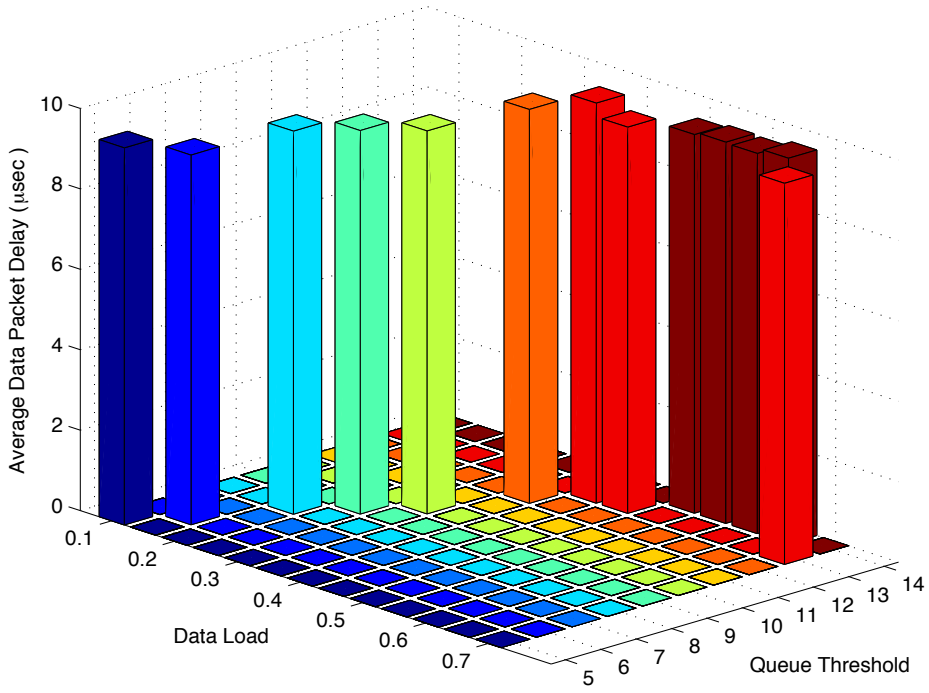


Figure 9.4: Average data packet delay under $\mathbb{D}_{QoS}$ = 10 μs.

the quality of our scheme, we set $\mathbb{D}_{QoS}$ =10 μs, loop over different values of data load, and numerically calculate the corresponding Q_{th} required to maintain the QoS requirement using (9.8), (9.11), and (9.12). Then, we run the simulation with the calculated Q_{th} to

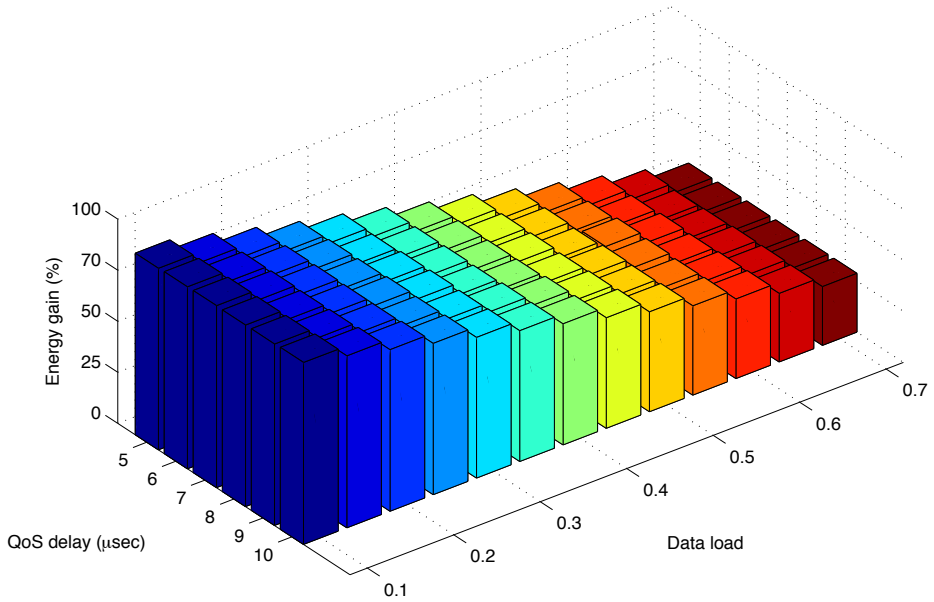


Figure 9.5: Energy gain vs. data load for different $\mathbb{D}_{QoS}$ values.

calculate the average data packets delay. Figure 9.4 shows that for all values of data load, the average data packets delay never violates the $\mathbb{D}_{QoS}$ constraint. Instead, our scheme succeeds to dynamically adjust Q_{th} according to the data load in order to maintain our ultimate objective of maximizing the saved power through the introduced ONU sleeping functionality, while satisfying the imposed average delay constraint.

In Figure 9.5 we show the energy saving gain vs. different values of $\mathbb{D}_{QoS}$ under different load. We prove that regardless of the QoS delay constraint, our framework achieves equal energy saving gain for some data traffic load. This is because that our framework can successfully decoupled sleeping time from QoS delay when fixing the load.

9.5 Summary

In this chapter, we presented a new sleep sizing and scheduling framework for energy saving in PON. We showed through both analytical modeling and extensive simulations that our framework can save energy without violating the QoS constraints. To the best of our knowledge, this was the first study that was able to decouple the sleeping time from

QoS delay constraints. This implies that sleeping time portion and consequently the energy saving ratio will only depend on the aggregated traffic load. Our framework can adaptively change the data Q_{th} to maintain the required $\mathbb{D}_{QoS}$ without affecting the energy saving gain.

Chapter 10

Concluding Remarks

10.1 Contribution in Brief

This dissertation focuses on bandwidth management in EPONs. Our goal is to propose robust DBA schemes that better utilize upstream bandwidth and achieve lower average delay. We investigated five different problems related to dynamic bandwidth allocation in EPONs. The investigated problems are excess bandwidth allocation, frame delineation, congestion, large round trip time delay in LR-EPONs, and downstream based DBA in Green EPONs.

Regarding excess bandwidth allocation, we presented a novel scheme, DES, to effectively distribute upstream excess bandwidth with overloaded ONUs using controlled allocation and without incorporating idle time. DES uses delayed allocation rather than early allocation. DES preserves ONUs polling order and hence reduces delay Jitter. DES shows delay reduction for EF, AF, and BE traffic.

For frame delineation, we presented a novel concept in DBA named bandwidth compensation. We also presented a new DBA scheme called EGSIP which improves TWDM-EOPN performance. EGSIP resolves the frame delineation problem and guarantees that each overloaded ONUs can obtain their maximum grant size. We proved via mathematical analysis that EGSIP is more robust than IPACT-st. Robustness has high impact on delay and buffer size reduction. The main reason behind EGSIP superior performance is its capa-

bility of utilizing the available bandwidth without loss as in IPACT-st and without reduction as in threshold reporting.

For congestion in EPONs, we proposed CALT. CALT incorporates interleaved polling along with offered load controlled maximum cycle time. CALT uses the number of over-loaded ONUs, N_O, as a heuristic to increase/decrease the maximum cycle time value. CALT incorporates a set of levels (values) of ONU maximum transmission window and both upper and lower threshold for N_O associated with each level. Numerical results show significant improvement in packet delay, dropping probability, bandwidth utilization and buffer size. CALT also reduces the delay jitter which is one of the important metrics in Diffserv EPONs.

For LR-EPONS, we addressed two challenges, namely bandwidth over-granting associated with MTP and on-the-fly void filling. For bandwidth over-granting, we proposed an interactive approach called Interactive Dynamic Bandwidth Allocation (IBA). IBA is able to achieve the minimum delay bounds for MTP. We also developed an approach to tune active running threads with the offered load. Our approach is able to choose the most well performing thread count at each individual load value. For void Filling in LR-EPONs, we presented a novel void filling approach, called Parallel Void Thread (PVT). PVT is a polling thread independent from the running DBA thread(s). When the OLT detects a void succeeding a bandwidth grant, PVT is invoked to allocate void based grant(s) (VBG(s)) to fill that void. PVT delay reduction mechanism is to enable ONUs to early transmit part of their upstream traffic during VBGs. This technique reduces the grant delays and enables bandwidth grants more frequently than both RBVF and RBPVF. There are three proposed VBG sizing schemes for PVT, namely VE, CCBVF, and SCBVF.

Within the context of LR-TWDM-EPONs, we presented a multi-feature solution that combines most of the addressed challenges. Our combined approach, IBA-WSCBVF, is capable of achieving significant delay reduction along the whole offered load range.

For bandwidth allocation in Green EPONs, we presented a novel sleep sizing and

scheduling framework for energy saving in PON. We showed through both analytical modeling and extensive simulations that our framework can save energy without violating the QoS constraints. To the best of our knowledge, this is the first study that decouples the sleeping time from QoS delay constraints. This implies that sleeping time portion and consequently the energy saving ratio will only depend on the aggregated traffic load. Our framework can adaptively change the data Q_{th} to maintain the required $\mathbb{D}_{QoS}$ without affecting the energy saving gain.

10.2 Future Research Direction

For future research direction, we propose to investigate Quality-of-Service based Wavelength Assignment (QoS-WA). In general, access networks are designed to carry different types of traffic classes (services). Each service has its own delay constraints. Since LR-TWDM-EPON is potential candidate for next generation EPONs, more investigation should be done to ensure that each service satisfies its delay constraints at all load range.

An easy solution could be to reduce the maximum grant delay and consequently reduces the maximum transmission window. Only higher priority services get benefit of this solution. Lower priority services will suffer from bandwidth starvation. Moreover, reducing grant delays is hard due to large RTT. Part of the solution can be PVT that can help with low load part. At high load, a more sophisticated wavelength assignment scheme is needed to ensure that each service gets its proper bandwidth and satisfies its delay constraints.

REFERENCES

[1] A. Dhaini, "Design and analysis of green mission-critical fiber-wireless broadband access networks," Ph.D. dissertation, University of Waterloo, 2011.

[2] G. Kramer, *Ethernet passive optical networks*. McGraw-Hill, 2005.

[3] A. Helmy, H. Fathallah, and H. Mouftah, "Interleaved polling versus multi-thread polling for bandwidth allocation in Long-Reach PONs," *IEEE/OSA J. Optical Commun. Netw*, vol. 4, no. 3, pp. 210–218, March 2012.

[4] A. Mercian, M. McGarry, and M. Reisslein, "Offline and online multi-thread polling in long-reach PONs: A critical evaluation," *J. Lightw. Technol.*, vol. 31, no. 12, pp. 2018–2028, June 2013.

[5] "IEEE Standard for Information Technology- Telecommunications and Information Exchange Between Systems- Local and Metropolitan Area Networks- Specific Requirements Part 3: Carrier Sense Multiple Access With Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications Amendment: Media Access Control Parameters, Physical Layers, and Management Parameters for Subscriber Access Networks," *IEEE Std 802.3ah-2004*, pp. 1–623, 2004.

[6] "IEEE Standard for Information technology– Local and metropolitan area networks– Specific requirements– Part 3: CSMA/CD Access Method and Physical Layer Specifications Amendment 1: Physical Layer Specifications and Management Parameters for 10 Gb/s Passive Optical Networks," *IEEE Std 802.3av-2009 (Amendment to IEEE Std 802.3-2008)*, pp. 1–227, 2009.

[7] A. Elrasad and B. Shihada, "A practical approach for excess bandwidth distribution for EPONs," in *Optical Fiber Communication Conference (OFC)*, 2014.

[8] H. Song, B.-W. Kim, and B. Mukherjee, "Multi-thread polling: A dynamic bandwidth distribution scheme in Long-Reach PON," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 2, pp. 134–142, February 2009.

[9] A. Elrasad and B. Shihada, "Parallel void thread in long-reach ethernet passive optical networks," *IEEE/OSA J. Optical Commun. Netw*, vol. 7, no. 7, pp. 656–668, 2015.

[10] G. Kramer, B. Mukherjee, and G. Pesavento, "IPACT: A dynamic protocol for an Ethernet PON (EPON)," *IEEE Commun. Mag.*, vol. 40, no. 2, pp. 74–80, 2002.

[11] D. Sala and A. Gummalla, PON functional requirements: services and performance, in IEEE 802.3ah Meeting in Portland OR, July 2001. [Online]. Available: http://www.ieee802.org/3/efm/public/jul01/presentations/sala_1_0701.pdf.

[12] M. McGarry, M. Reisslein, and M. Maier, "Ethernet passive optical network architectures and dynamic bandwidth allocation algorithms," *IEEE Commun. Surveys Tuts.*, vol. 10, no. 3, pp. 46–60, Third 2008.

[13] B. Kantarci and H. Mouftah, "Bandwidth distribution solutions for performance enhancement in long-reach passive optical networks," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 3, pp. 714–733, Third 2012.

[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] "Ethernet PON (EPON): Design and analysis of an optical access network," *Photonic Network Communications*, vol. 3, no. 3, 2001.

[16] M. A. A. Boon, "Polling models: from theory to traffic intersections," Ph.D. dissertation, Technische Universiteit Eindhoven, 2011.

[17] D. Bertsekas and R. Gallager, *Data Networks (2nd Ed.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1992.

[18] C. Assi, Y. Ye, S. Dixit, and M. Ali, "Dynamic bandwidth allocation for quality-of-service over Ethernet PONs," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 9, pp. 1467–1477, 2003.

[19] X. Bai, A. Shami, and C. Assi, "On the fairness of dynamic bandwidth allocation schemes in Ethernet passive optical networks," *Comput. Commun.*, vol. 29, no. 11, pp. 2123–2135, 2006.

[20] A. Shami, X. Bai, N. Ghani, C. Assi, and H. Mouftah, "QoS control schemes for two-stage Ethernet passive optical access networks," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 8, pp. 1467–1478, 2005.

[21] A. Dhaini, C. Assi, M. Maier, and A. Shami, "Dynamic wavelength and bandwidth allocation in hybrid TDM/WDM EPON networks," *J. Lightw. Technol.*, vol. 25, no. 1, pp. 277–286, January 2007.

[22] J. R. Ferguson, M. Reisslein, and M. P. McGarry, "Online excess bandwidth distribution for Ethernet passive optical networks," *J. Opt. Netw.*, vol. 8, no. 4, pp. 358–369, April 2009.

[23] S. Y. Choi, S. Lee, T.-J. Lee, M. Y. Chung, and H. Choo, "Double-phase polling algorithm based on partitioned ONU subgroups for high utilization in EPONs," *J. Opt. Commun. Netw.*, vol. 1, no. 5, pp. 484–497, Oct 2009.

[24] J. Zheng, "Efficient bandwidth allocation algorithm for Ethernet passive optical networks," *Communications, IEE Proceedings*, vol. 153, no. 3, pp. 464–468, 2006.

[25] K. H. Kwong, D. Harle, and I. Andonovic, "Dynamic bandwidth allocation algorithm for differentiated sservices over WDM EPONs," in *Communications Systems, 2004. ICCS 2004. The Ninth International Conference on*, September 2004, pp. 116–120.

[26] H.-Y. Wu, W.-P. Chen, K.-L. Huang, and T.-F. Lee, "A novel offline-based DWBA algorithm with sorting report for WDM-EPON system," in *Communications and Mobile Computing (CMC), 2010 International Conference on*, vol. 1, April 2010, pp. 567–571.

[27] Y.-M. Yang, J.-M. Nho, H. Perros, N. P. Mahalik, K. Kim, and B.-H. Ahn, "Dynamic bandwidth allocation by using efficient threshold reporting for quality of service in Ethernet passive optical networks," *Optical Engineering*, vol. 45, no. 3, pp. 035 001–035 001–6, 2006.

[28] D. Nikolova, B. Van Houdt, and C. Blondia, "Dynamic bandwidth allocation algorithms for Ethernet passive optical networks with threshold reporting," *Telecommunication Systems*, vol. 28, no. 1, pp. 31–52, 2005.

[29] G. Kramer, B. Mukherjee, S. Dixit, Y. Ye, and R. Hirth, "Supporting differentiated classes of service in Ethernet passive optical networks," *OSA J. Opt. Netw.*, vol. 1, no. 8, pp. 280–298, August 2002.

[30] H.-W. J. K. K. Yeon-Mo Yang, Sang-Wook Lee and B.-H. Ahn, "Inter-ONU bandwidth scheduling by using threshold reporting and adaptive polling for QoS in EPONs," *ETRI Journal*, vol. 27, no. 6, pp. 802–805, 2005.

[31] A. Helmy and H. Fathallah, "Taking turns with adaptive cycle time a decentralized media access scheme for LR-PON," *J. Lightw. Technol.*, vol. 29, no. 21, pp. 3340–3349, Nov 2011.

[32] ——, "Taking turns with adaptive cycle time and immediate tagging: A decentralized upstream media access scheme for long-reach PON," in *Global Telecommunications Conference (GLOBECOM 2011), 2011 IEEE*, Dec 2011, pp. 1–5.

[33] M. Kiaei, K. Fouli, M. Scheutzow, M. Maier, M. Reisslein, and C. Assi, "Low-latency polling schemes for long-reach passive optical networks," *IEEE Trans. Commun.*, vol. 61, no. 7, pp. 2936–2945, July 2013.

[34] M. McGarry, M. Reisslein, C. Colbourn, M. Maier, F. Aurzada, and M. Scheutzow, "Just-in-time scheduling for multichannel EPONs," *J. Lightw. Technol.*, vol. 26, no. 10, pp. 1204–1216, May 2008.

[35] K. Kanonakis and I. Tomkos, "Improving the efficiency of online upstream scheduling and wavelength assignment in hybrid WDM/TDMA EPON networks," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 6, pp. 838–848, August 2010.

[36] ——, "Scheduling and wavelength assignment issues in metro-scale hybrid WDM/TDMA EPONs," in *Future Network and Mobile Summit, 2010*, June 2010, pp. 1–7.

[37] ——, "Online upstream scheduling and wavelength assignment algorithms for WDM-EPON networks," in *35th European Conference on Optical Communication (ECOC), 2009*. IEEE, 2009, pp. 1–2.

[38] A. Buttaboni, M. De Andrade, and M. Tornatore, "A multi-threaded dynamic bandwidth and wavelength allocation scheme with void filling for Long-Reach WDM/TDM PONs," *J. Lightw. Technol.*, vol. 31, no. 8, pp. 1149–1157, April 2013.

[39] ——, "New and improved approaches for dynamic bandwidth and wavelength allocation in LR WDM/TDM PON," in *Italian Networking Workshop*, 2013.

[40] Y. Yan, S.-W. Wong, L. Valcarenghi, S.-H. Yen, D. Campelo, S. Yamashita, L. Kazovsky, and L. Dittmann, "Energy management mechanism for ethernet passive optical networks (EPONs)," in *Communications (ICC), 2010 IEEE International Conference on*, May 2010, pp. 1–5.

[41] A. R. Dhaini, P.-H. Ho, and G. Shen, "Toward green next-generation passive optical networks," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 94–101, 2011.

[42] ——, "Energy efficiency in ethernet passive optical networks: for how long can ONU sleep?" *University of Waterloo, Tech. Rep., Mar*, 2011.

[43] J. Zhang and N. Ansari, "Toward energy-efficient 1G-EPON and 10G-EPON with sleep-aware MAC control and scheduling," *IEEE Commun. Mag.*, vol. 49, no. 2, pp. s33–s38, 2011.

[44] L. Shi, S. S. Lee, H. Song, and B. Mukherjee, "Energy-efficient long-reach passive optical network: A network planning approach based on user behaviors," *IEEE Systems Journal*, vol. 4, no. 4, pp. 449–457, Dec 2010.

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

[46] L. Shi, P. Chowdhury, and B. Mukherjee, "Saving energy in long-reach broadband access networks: architectural approaches," *IEEE Communications Magazine*, vol. 51, no. 2, pp. S16–S21, February 2013.

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

[48] S. S. Lee and A. Chen, "Design and analysis of a novel energy efficient ethernet passive optical network," in *Networks (ICN), 2010 Ninth International Conference on*. IEEE, 2010, pp. 6–9.

[49] 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," *IEEE/OSA J. Optical Commun. Netw*, vol. 2, no. 9, pp. 716–729, 2010.

[50] Y. Yan, S.-W. Wong, L. Valcarenghi, S.-H. Yen, D. R. Campelo, S. Yamashita, L. Kazovsky, and L. Dittmann, "Energy management mechanism for ethernet passive optical networks (EPONs)," in *Communications (ICC), 2010 IEEE International Conference on*. IEEE, 2010, pp. 1–5.

[51] S. Chen, A. R. Dhaini, P.-H. Ho, B. Shihada, G. Shen, and C.-H. Lin, "Downstream-based scheduling for energy conservation in green EPONs," *Journal of Communications*, vol. 7, no. 5, pp. 400–408, 2012.

[52] M. P. McGarry, M. Reisslein, and M. Maier, "WDM ethernet passive optical networks," *IEEE Commun. Mag.*, vol. 44, no. 2, pp. 15–22, 2006.

[53] S. Bharati and P. Saengudomlert, "Analysis of mean packet delay for dynamic bandwidth allocation algorithms in EPONs," *J. Lightw. Technol.*, vol. 28, no. 23, pp. 3454–3462, 2010.

[54] F. Aurzada, M. Scheutzow, M. Herzog, M. Maier, and M. Reisslein, "Delay analysis of ethernet passive optical networks with gated service," *J. Opt. Netw.*, vol. 7, no. 1, pp. 25–41, Jan 2008.

[55] F. Aurzada, M. Scheutzow, M. Reisslein, N. Ghazisaidi, and M. Maier, "Capacity and delay analysis of next-generation passive optical networks (NG-PONs)," *IEEE Trans. Commun.*, vol. 59, no. 5, pp. 1378–1388, May 2011.

[56] F. Aurzada, M. Scheutzow, M. Reisslein, and M. Maier, "Towards a fundamental understanding of the stability and delay of offline WDM EPONs," *J. Opt. Commun. Netw.*, vol. 2, no. 1, pp. 51–66, Jan 2010.

[57] A. Dixit, B. Lannoo, D. Colle, M. Pickavet, and P. Demeester, "Delay models in ethernet long-reach passive optical networks," in *Computer Communications (INFOCOM), 2015 IEEE Conference on*, April 2015, pp. 1239–1247.

[58] H. Takagi, "Queuing analysis of polling models," *ACM Comput. Surv.*, vol. 20, no. 1, pp. 5–28, 1988.

[59] M. E. Crovella and A. Bestavros, "Self-similarity in world wide web traffic: evidence and possible causes," *IEEE/ACM Trans. Netw.*, vol. 5, no. 6, pp. 835–846, 1997.

[60] 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," *IEEE Commun. Mag.*, vol. 47, no. 3, pp. S40–S48, 2009.

[61] R. Jain, A. Durresi, G. Babic, Throughput Fairness Index: An Explanation, in: ATM_Forum. [Online]. Available: http://www.cse.ohio-state.edu/~jain/atmf/a99-0045.htm/.

[62] D. Payne, R. Davey, D. Faulkner, and S. Hornung, "Optical networks for the broadband future," *Broadband Optical Access Networks and Fiber-to-the-Home: Systems Technologies and Deployment Strategies*, pp. 189–214, 2006.

[63] B. Skubic, J. Chen, J. Ahmed, B. Chen, L. Wosinska, and B. Mukherjee, "Dynamic bandwidth allocation for long-reach PON: overcoming performance degradation," *IEEE Commun. Mag.*, vol. 48, no. 11, pp. 100–108, 2010.

[64] J. Ahmed, J. Chen, L. Wosinska, B. Chen, and B. Mukherjee, "Efficient inter-thread scheduling scheme for long-reach passive optical networks," *Communications Magazine, IEEE*, vol. 51, no. 2, pp. S35–S43, February 2013.

[65] A. Dixit, B. Lannoo, D. Colle, M. Pickavet, and P. Demeester, "Synergized-adaptive multi-gate polling with void filling: Overcoming performance degradation in lr-pons," *IEEE/OSA J. Optical Commun. Netw*, vol. 7, no. 9, pp. 837–850, 2015.

[66] A. Elrasad and B. Shihada, "Reducing attendance time in LR-EPONs with differentiated services," in *INFOCOM 2015 Student Workshop (INFOCOM'15 Student Workshop)*, Hong Kong, April 2015, pp. 70–71.

[67] N. Merayo, T. Jimnez, P. Fernndez, R. Durn, R. Lorenzo, I. de Miguel, and E. Abril, "A bandwidth assignment polling algorithm to enhance the efficiency in QoS Long Reach EPONs," *Eur. Trans. Telecommun.*, vol. 22, no. 1, pp. 35–44, January 2011.

[68] A. Elrasad and B. Shihada, "Blind void filling in LR-EPONs: How efficient it can be?" *2015 IEEE 16th International Conference on High Performance Switching and Routing (IEEE HPSR15), Budapest, Hungary*, 2015.

[69] H. Takagi, "Frontiers in queueing," J. H. Dshalalow, Ed. Boca Raton, FL, USA: CRC Press, Inc., 1997, ch. Queueing Analysis of Polling Models: Progress in 1990-1994, pp. 119–146.

[70] B. Kantarci and H. Mouftah, "Bandwidth distribution solutions for performance enhancement in long-reach passive optical networks," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 3, pp. 714–733, Third Quarter 2012.

[71] G. Shen and R. Tucker, "Fixed mobile convergence (FMC) architectures for broadband access: integration of EPON and WiMax," pp. 678 403–678 403–13, 2007.

[72] Y. Luo, S. Yin, T. Wang, Y. Suemura, S. Nakamura, N. Ansari, and M. Cvijetic, "QoS-aware scheduling over hybrid optical wireless networks," in *Optical Fiber Communication and the National Fiber Optic Engineers Conference, 2007. OFC/NFOEC 2007. Conference on*, March 2007, pp. 1–7.

[73] Y. Luo, T. Wang, S. Weinstein, and M. Cvijetic, "Integrating optical and wireless services in the access network," in *Optical Fiber Communication Conference, 2006 and the 2006 National Fiber Optic Engineers Conference. OFC 2006*, 2006, pp. 10 pp.–.

[74] S. Sarkar, S. Dixit, and B. Mukherjee, "Hybrid wireless-optical broadband-access network (WOBAN): A review of relevant challenges," *J. Lightw. Technol.*, vol. 25, no. 11, pp. 3329–3340, Nov 2007.

[75] W.-T. Shaw, S.-W. Wong, N. Cheng, K. Balasubramanian, X. Zhu, M. Maier, and L. G. Kazovsky, "Hybrid architecture and integrated routing in a scalable optical–wireless access network," *J. Lightw. Technol.*, vol. 25, no. 11, pp. 3443–3451, 2007.

[76] A. Elrasad, M. Khafagy, and B. Shihada, "Sleep-time sizing and scheduling in green passive optical networks," in *Communications in China (ICCC), 2012 1st IEEE International Conference on*, Aug 2012, pp. 260–265.

[77] N. Tian and Z. G. Zhang, *Vacation queueing models: theory and applications*. Springer Science & Business Media, 2006, vol. 93.

[78] C.-H. Ng and S. Boon-Hee, *Queueing modelling fundamentals: With applications in communication networks*. John Wiley & Sons, 2008.

APPENDICES

A List of Publications

Published

• A. Elrasad and B. Shihada, "Parallel Void Thread in Long-Reach Ethernet Passive Optical Networks," IEEE/OSA Journal of Optical Communications and Networking 7, 656-668 (2015).

• A. Elrasad and B. Shihada, "Reducing attendance time in LR-EPONs with differentiated services," in Computer Communications Workshops (INFOCOM WORKSHPS), 2015 IEEE Conference on, pp.65-66, April 26 2015-May 1 2015.

• A. Elrasad and B. Shihada, "Blind void filling in LR-EPONs: How efficient it can be?," 2015 IEEE 16th International Conference on High Performance Switching and Routing (IEEE HPSR15), Budapest, Hungary, 2015.

• A. Elrasad and B. Shihada, "A Practical Approach for Excess Bandwidth Distribution for EPONs," in Proc. Optical Fiber Communication Conference and Exposition (OFC/NFOEC), ISBN: 978-1-55752- 993-0, 2014.

• A. Elrasad, M. Khafagy, and B. Shihada, "Sleep-time Sizing and Scheduling in Green Passive Optical Networks," in Proc. IEEE International Conference on Communications in China (ICCC), pp. 308-313, 2012.

In Process

• A. Elrasad and B. Shihada, “Delay reduction in LR-TWDM-EPONs: what are the options?,” IEE/OSA Journal of Lightwave Technology.

• A. Elrasad and B. Shihada, “Delayed Excess Bandwidth Allocation in TWDM EPONs,” OSA Optics Express.