Title	A Quality of Service framework for upstream traffic in LTE across an XG-PON backhaul
Authors	Arokkiam, Jerome A.
Publication date	2017
Original Citation	Arokkiam, J. A. 2017. A Quality of Service framework for upstream traffic in LTE across an XG-PON backhaul. PhD Thesis, University College Cork.
Type of publication	Doctoral thesis
Rights	© 2017, Jerome A. Arokkiam. - http://creativecommons.org/licenses/by-nc-nd/3.0/
Download date	2025-08-24 19:21:16
Item downloaded from	https://hdl.handle.net/10468/5055

A Quality of Service framework for upstream traffic in LTE across an XG-PON backhaul

Jerome A. Arokkiam

Contents

List of Figures

List of Tables

I, Jerome A. Arokkiam, certify that this thesis is my own work and I have not obtained a degree in this university or elsewhere on the basis of the work submitted in this thesis.

....

Jerome A. Arokkiam

I dedicate this thesis to my parents and every one of my teachers.

Acknowledgements

The journey of PhD is long and rewarding; I would not have completed mine without the support of many.

I am extremely grateful to my supervisors Prof. Cormac J. Sreenan and Prof. Kenneth N. Brown, for converting my amateur aspirations in to a meaningful research career within such a short timespan. I am thankful to them for the frequent meetings through which they gradually nurtured me to produce strong research contributions, the invaluable yet untiring feedback that I received from them to generate meaningful publications and for their occasional friendly advice that made me miss my father less.

I am also thankful to every single person who has been a part of the Insight (formerly 4C) and MISL research groups during my stay in UCC, including Prof. Barry O'Sullivan, Caitriona Walsh, Eleanor O'Riordan, Peter Machale and Mary Noonan for their constant support and reminder that doing a PhD in Ireland, far away from home, is a joyous journey when it is within a large group of like-minded people. I will always cherish the lasting friendships of Saim Ghafoor and Samreen Umer, who have been my pillars of moral support for research since I started my PhD and I also appreciate the suggestions and mentorship provided by the senior colleagues, Dr. Xiuchao Wu, Dr. Lanny Sitanaya, Dr. David Stynes and Dr. Mustafa Al-Bado.

I would like thank my examiners Dr. Brenaen Jennings and Dr. Dan Grigoras and once again my supervisors for their suggestions that made this thesis a true representation of all my research contributions. I also acknowledge Science Foundation Ireland for supporting my research in UCC through the CTVR grant.

I am always indebted to the unconditional love of my parents, my siblings and my wife, without whom I would not have had yet another rewarding journey in my life.

Abbreviations

10G-EPON ...	10Gigabit-Capable Ethernet PON
3GPP	Third Generation Partnership Project
AB	Allocation Bytes
ACK	Acknowledgement
ATM	Asynchronous Transfer Mode
BDP	Bandwidth Delay Product
BF	Burst Factor
BS	Base Station
CBR	Constant bit rate
CDF	Cumulative Distribution Function
CWND	Congestion Window at the TCP Sender
DAMA	Data Mining Forecasting DBA
DBA	Dynamic Bandwidth Allocation
DSCP	Differentiated Services Code Point
DWRR	Deficit Weighted Round Robin
E2E	end-to-end
EBU	Efficient Bandwidth Utilisation DBA
eNB	Evolved NodeB
EPC	Evolved Packet Core
EPON	Ethernet PON
EqD	Equalization Delay
FEC	Forward Error Correction
FEC	Forward Error Correction
FI	(Jain's) Fairness Index
FIFO	First-In-First-Out
FMC	Fixed Mobile Convergence
FSAN	Full Service Access Network
FTP	File Transfer Protocol
FTTx	Fibre-To-The-x, with x denoting building, home, cell, etc.
GBR	Guaranteed Bit Rate
GDR	Guaranteed Data Rate
GEM	GPON Encapsulation Method
GIANT	GigaPON Access Network DBA
GLCNA	GPON-LTE Converged Network Architecture
GPON	Gigabit PON
H-TCP	Hamilton TCP
HD	High Definition
HTTP	Hyper Text Transfer Protocol
IACG	Immediate Allocation with Colorless Grant DBA
IP	Internet Protocol
ITU-T	Telecommunication Standardization Unit of Intentional Telecommunication Union
LAN	Large Area Network
LENA	LTE-EPC Network Simulator in ns-3
LTE	Long Term Evolution
MAC	Medium Access Control

MDR .... Maximum Data Rate
MME .... Mobility Management Entity
MPCP .... Multi-Point Control Protocol
MSS .... Maximum Service Size in XG-PON DBA
NSC .... Network Simulation Cradle
ODN .... Optical Distribution Network
OFDMA ...... Orthogonal Frequency Division Multiple Access
OLT .... Optical Line Terminal
OMCI .... ONU Management and Control Interface
ONU .... Optical Network Unit
PDB .... Packet Delay Budget
PDN-GW ..... Packet Data Network Gateway
PFS .... Proportional Fair Scheduler
PHY .... Physical
PLOAM ...... Physical Layer Operations, Administration and Maintenance
PMD .... Physical Medium Dependent Layer
PON .... Passive Optical Networks
PPBP .... Poission Pareto Burst Process
PQ .... Priority Queuing
PSB .... Physical Synchronisation Block
QCI .... QoS Class Identifier
QoS .... Quality of Service
R&F .... Radio-and-Fibre
RAN .... Radio Access Network
RoF .... Radio-over-Fibre
RR .... Round Robin
RTO .... Retransmission Time-out
RTT .... Round Trip Time
SDU .... Service Data Units
SFDBA .... Simple and Feasible DBA
SGW .... Serving Gateway
SI .... Service Interval
T-CONT ...... Traffic Containers
TCP .... Transport Control Protocol
TDMA .... Time Division Multiple Access
TTI .... Transmission Time Interval
UDP .... User Datagram Protocol
UE .... User Equipment
VoIP .... Voice-over-Internet Protocol
WRR .... Weighted Round Robin
XG-PON ..... 10Gigabit-Capable Gigabit PON
XGEM .... XG-PON Encapsulation Method
XGIANT-D ... Deficit XGIANT DBA
XGIANT-P ... Proportional XGIANT DBA
XGTC .... XG-PON Transmission Convergence Layer

Abstract

Passive Optical Networks (PON) are promising as a transport network technology due to the high network capacity, long reach and strong QoS support in the latest PON standards. Long Term Evolution (LTE) is a popular wireless technology for its large datarates in the last mile. The natural integration of LTE and XG-PON, which is one of the latest standards of PON, presents several challenges for XG-PON to satisfy the backhaul QoS requirements of aggregated upstream LTE applications. This thesis proves that a dedicated XG-PON-based backhaul is capable of ensuring the QoS treatment required by different upstream application types in LTE, by means of standard-compliant Dynamic Bandwidth Allocation (DBA) mechanisms.

First the design and evaluation of a standard-compliant, robust and fast XG-PON simulation module developed for the state-of-the-art ns-3 network simulator is presented in the thesis. This XG-PON simulation module forms a trustworthy and large-scale simulation platform for the evaluations in the rest of the thesis, and has been released for use by the scientific community.

The design and implementation details of the XGIANT DBA, which provides standard-complaint QoS treatment in an isolated XG-PON network, are then presented in the thesis along with comparative evaluations with the recently-published EBU DBA. The evaluations explored the ability of both XGIANT and EBU in terms of queuing-delay and throughput assurances for different classes of simplified (deterministic) traffic models, for a range of upstream loading in XG-PON.

The evaluation of XGIANT and EBU DBAs are then presented for the context of a dedicated XG-PON backhaul in LTE with regard to the influence of standard-compliant and QoS-aware DBAs on the performance of large-scale, UDP-based applications. These evaluations disqualify both XGIANT and EBU DBAs in providing prioritised queuing-delay performances for three upstream application types (conversational voice, peer-to-peer video and best-effort Internet) in LTE; the evaluations also indicate the need to have more dynamic and efficient QoS policies, along with an improved fairness policy in a DBA used in the dedicated XG-PON backhaul to ensure the QoS requirements of the upstream LTE applications in the backhaul.

Finally, the design and implementation details of two standard-compliant DBAs, namely Deficit XGIANT (XGIANT-D) and Proportional XGIANT (XGIANT-P), which provide the required QoS treatment in the dedicated XG-PON backhaul for all three application types in the LTE upstream are presented in the thesis. Evaluations of the XGIANT-D and XGIANT-P DBAs presented in the thesis prove the ability of the fine-tuned QoS and fairness policies in the DBAs in ensuring prioritised and fair queuing-delay and throughput efficiency for UDP- and TCP-based applications, generated and aggregated based on realistic conditions in the LTE upstream.

Chapter 1

Introduction

> If I have seen further, it is by standing on the shoulders of giants
> —Sir Isac Newton, 5 Feb. 1675

Passive Optical Network (PON) is a well-established transport network technology due to its high network capacity, low per-user cost, flexible bandwidth allocation and strong quality of service (QoS) support. The Gigabit PON (GPON [45]) and Ethernet PON (EPON [40]) standards, along with their upgraded versions 10Gigabit-capable GPON (XG-PON [46]) and 10Gigabit-EPON (10G-EPON [42]) respectively, have thus extended the advantages of PON in the transport network, while promising long-term benefits such as easy capacity upgrades and extended physical reach.

Long Term Evolution (LTE) is a popular cellular technology that offers high datarates and is now deployed widely. However, the cost of providing dedicated backhaul links for new LTE base stations is prohibitive due to the need to cater for bandwidth-hungry mobile applications.

The convergence of PON and LTE networks, as part of the Fixed-Mobile Convergence (FMC) provides significant advantages in terms of network infrastructure and easy bandwidth upgrades in the backhaul and flexible sharing of the limited bandwidth in the backhaul between different base stations in LTE. However, the convergence also presents several challenges in terms of designing a converged network architecture and an efficient QoS framework to provide reasonable QoS treatment for different classes of realistic application traffic in the LTE upstream across the PON-based backhaul [85, 70, 75, 81, 4].

This thesis addresses these concerns in the context of designing a suitable DBA for the integrated network architecture of XG-PON-based LTE backhaul.

1.1   Thesis Statement

XG-PON-standard-compliant Dynamic Bandwidth Allocation mechanisms can ensure prioritised and fair Quality of Service treatment for multiple classes of application traffic in the upstream of LTE networks using a dedicated XG-PON backhaul.

1.2   List of achievements and publications

The following are the personal contributions of the author with regard to the development of QoS-aware and standard-compliant DBA mechanisms for the dedicated XG-PON backhaul in LTE:

1) A standard-compliant XG-PON module compatible with the state-of-the ns-3 network simulator, has been developed in collaboration with two other colleagues (Dr. Xiuchao Wu and Pedro Alvarez) and validated for performance using several evaluations in ns-3. Since this is a collaborative work, author's contributions account for about 40% of the details of the XG-PON simulation module presented in Chapter 4. Specifically, the contributions are in terms of designing and implementing the classes related with the scheduling and DBA mechanisms of XG-PON as well as evaluating the performance of the simulation module for functionality validation. The XG-PON simulation module was published in the SIMULATION journal in January 2017[10]; its code, documentation and examples can be found in the public repository[97].

2) Using the above XG-PON module and other existing modules in ns-3, an analysis has been performed for (UDP- and TCP- based) traffic behaviours in XG-PON, for several complex scenarios, using deterministic application models and different congestion control algorithms in TCP. The findings were published in the Institute of Electrical and Electronics Engineers (IEEE) GLOBECOM 2014 conference[13] and won a best paper award in the Access Networks and Systems track (presented by the Transmission, Access and Optical System Technical Committee of IEEE Communication Society). Chapter 5 is based on the contributions of this paper.

3) GIANT DBA, which was originally designed for the GPON standard, has been modified with improved QoS policies and fine-tuned parameters, to design the XG-PON-standard-compliant XGIANT DBA, which provided differentiated QoS treatments for three different classes of upstream traffic in the context of a stand-alone XG-PON. Both XGIANT and EBU, which is another recent GIANT-improved and XG-PON standard-compliant DBA, have been implemented in the XG-PON simulation module in ns-3. After validating the ns-3-based implementation of EBU against its original implementation, further evaluations of XGIANT and EBU has been performed, under a range of upstream loading in XG-PON, to establish the superiority of XGIANT in terms of mean queuing-delay performance for all three applications types; XGIANT was also evaluated for conformity in terms of throughput performance. The findings, which were published in the IEEE International Conference of Communications (ICC) 2015[11], are presented in Chapter 6.

4) The mean queuing-delay performance of the XGIANT and EBU DBAs has been evaluated to identify the suitability of the DBAs in the XG-PON backhaul in LTE, after implementing a standard-compliant integrated network architecture and a QoS met-

rics conversion scheme in ns-3 for the convergence of XG-PON and LTE in ns-3. Based on the evaluations and after identifying several avenues of improvement with regard to the QoS policies of XGIANT and EBU, two standard-compliant DBAs (XGIANT-D and XGIANT-P) have been designed and evaluated for the XG-PON backhaul in LTE, to ensure improved QoS performance in terms of prioritised and fair mean queuing-delay and efficient throughput for the voice, video and best-effort application traffic in the under-loaded and fully-loaded upstream of the XG-PON backhaul. The findings, which were published in the 10th (2016) IEEE International Symposium on Performance Modelling and Evaluation of Computer and Telecommunication Networks[12], are presented in Chapter 7.

5) Several thousands of TCP-based realistic application traffic (of voice, video or best-effort class) flows have been configured and simulated in ns-3 for the above integrated network architecture of XG-PON and LTE, so that the performance of four DBAs, namely XGIANT, EBU, XGIANT-D and XGIANT-P, can be analysed in the contexts of the traditional (point-to-point) and XG-PON backhauls in LTE. Evaluations, validations and discussions have been provided with regard to the robustness of the DBAs when serving the same three types of realistically-generated TCP-based upstream applications (voice, video and best-effort) across the XG-PON backhaul in LTE. These findings are presented in detail in Chapter 8 and are under review for a journal publication.

1.3   Thesis Structure

The thesis is structured as follows:

Chapter 2 presents the technologies, concepts and terminologies.

Chapter 3 presents the related work in the literature.

Chapter 4 presents the design, implementation and evaluation details of the XG-PON simulation module, for the state-of-the-art ns-3 network simulator.

Chapter 5 presents the performance evaluation of the TCP-based application traffic across a stand-alone XG-PON, in the context of challenges encountered by TCP in a large BDP network regarding throughput efficiency, utilisation fairness and convergence and responsiveness of TCP flows under complex network environments.

Chapter 6 presents the design, implementation and evaluations of the XGIANT DBA, which provides standard-compliant and prioritised QoS treatment for three different classes upstream traffic in the stand-alone XG-PON, when compared against the validated implementation of another XG-PON-standard-compliant DBA, EBU.

Chapter 7 presents the design, implementation and evaluation of two tailor-made DBAs, namely XGIANT-D and XGIANT-P, which are developed, based on and improving

upon the performance evaluation of XGIANT and EBU DBAs, for providing prioritised, efficient and fair QoS treatment across the XG-PON backhaul for three types of UDP-based upstream traffic in LTE.

Chapter 8 presents the performance evaluation of TCP-based upstream application traffic in LTE, across a traditional (point-to-point) and the XG-PON backhauls, so that the robustness of the QoS-aware DBAs designed for the XG-PON can be validated for the QoS assurances towards the different types of realistically-generated application traffic in the LTE upstream.

Chapter 9 concludes the thesis with a summary of the contributions and discusses the research avenues for the future work.

Chapter 2

Background

This chapter introduces the technologies, concepts and terminologies used in this thesis, whose focus is to present the design, implementation and evaluation of fine-tuned DBA mechanisms in the context of a dedicated XG-PON backhaul for LTE. The details of XG-PON and LTE, which are the two primary technologies involved in the contributions of this thesis, are discussed in detail, in sections 2.1 and 2.3 respectively, with particular attention towards the existing standard definitions and QoS frameworks of both technologies. Popular scheduling concepts which are relevant for communication networks are presented in section 2.4 followed by the description and algorithms of Transport Control Protocol (TCP) in section 2.5. Finally, the details of the state-of-the-art ns-3 network simulator is presented in section 2.6.

2.1   Passive Optical Network (PON)

PON is a point-to-multipoint fibre network which has two kinds of equipment: the first kind, the active elements, includes the Optical Line Terminal (OLT) in central office and Optical Network Units (ONU) at/near the customer premises; the second kind, the passive elements, includes the optical splitters/jointers, which connect OLT and ONUs, without the need for any other active electronic equipment in the middle, thereby reducing capital and operational costs significantly.

In a classical Time Division Multiple Access (TDMA)-based PON, as in Figure 2.1, OLT broadcast the downstream traffic to all the ONUs that share the same optical fibre, with encryption used to prevent eavesdropping. The upstream traffic from the ONUs is interleaved by the OLT for using the optical fibre in a time-shared manner. Since PON networks are capable of reaching tens of kilometres in distance, PON has the capability to cover large geographical regions. In the traditional regions of telecommunications, therefore, PON fits in between the metro networks and the end-users, with the OLT
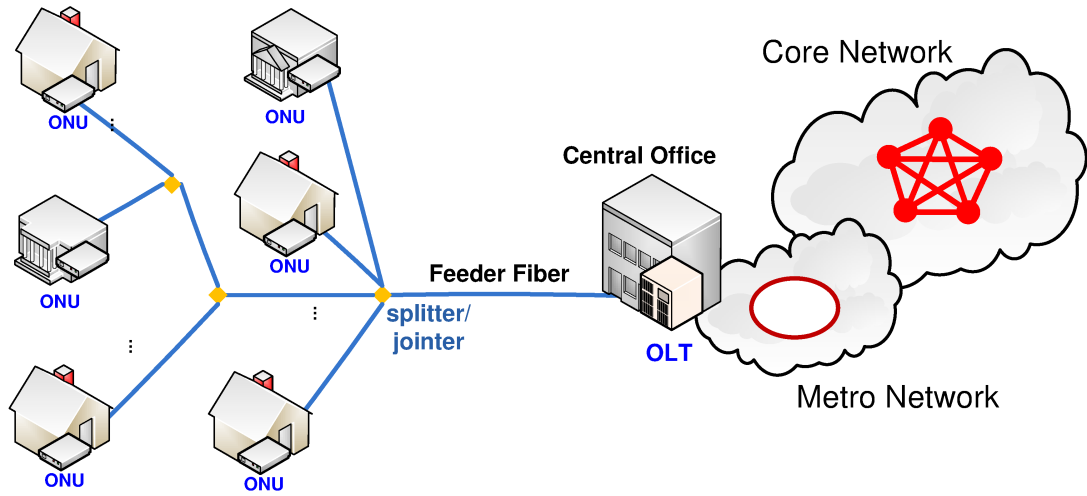
Figure 2.1: An illustration of PON as part of the regional network

placed closer to the Central Office and the ONUs placed near the end-users. Since the downstream (broadcast) traffic is controlled in a TDMA manner by a single entity (OLT), no collisions can be expected to occur in a downstream transmission of traffic in PON. However, since ONUs are allowed to have different distances to the OLT, the data bursts from these ONUs require carefully coordinated scheduling for providing a collision-free and efficient upstream data communication in the upstream. That is, in order to accommodate the dynamics in bandwidth demands from users and exploit the gain of statistical multiplexing, a dynamic bandwidth allocation (DBA) mechanism is required in the PON to share the upstream bandwidth. DBA works on the principle of polling-based scheduling between the ONUs and the OLT: ONUs first report their buffer occupancy to OLT, which then dynamically allocates the upstream bandwidth to the ONUs based on their bandwidth demands and their QoS definitions/requirements.

PON gained its recognition for large-scale deployment when EPON was standardised by the Ethernet in the First Mile task force of IEEE[41]; Full Service Access Network (FSAN) group of the Telecommunication Sector of the International Telecommunications Union (ITU-T) also introduced its version of PON standard with GPON [45] shortly after. Since then, the Fibre-To-The-x (FTTx, where x denotes the aggregation point near the customer, such as home, building, cell, etc.) networks, based on standardised PON technologies, have been widely deployed in many countries around the globe, including in the US, Korea and Japan.

XG-PON[46] is the new standard released by the FSAN. XG-PON improves GPON in numerous aspects, including physical datarates, physical reach, split ratio and number of ONUs supported; notable changes include increasing the default downstream data rate to 10 Gb/s from 2.5Gb/s, while increasing the upstream data rate from 1.25Gb/s
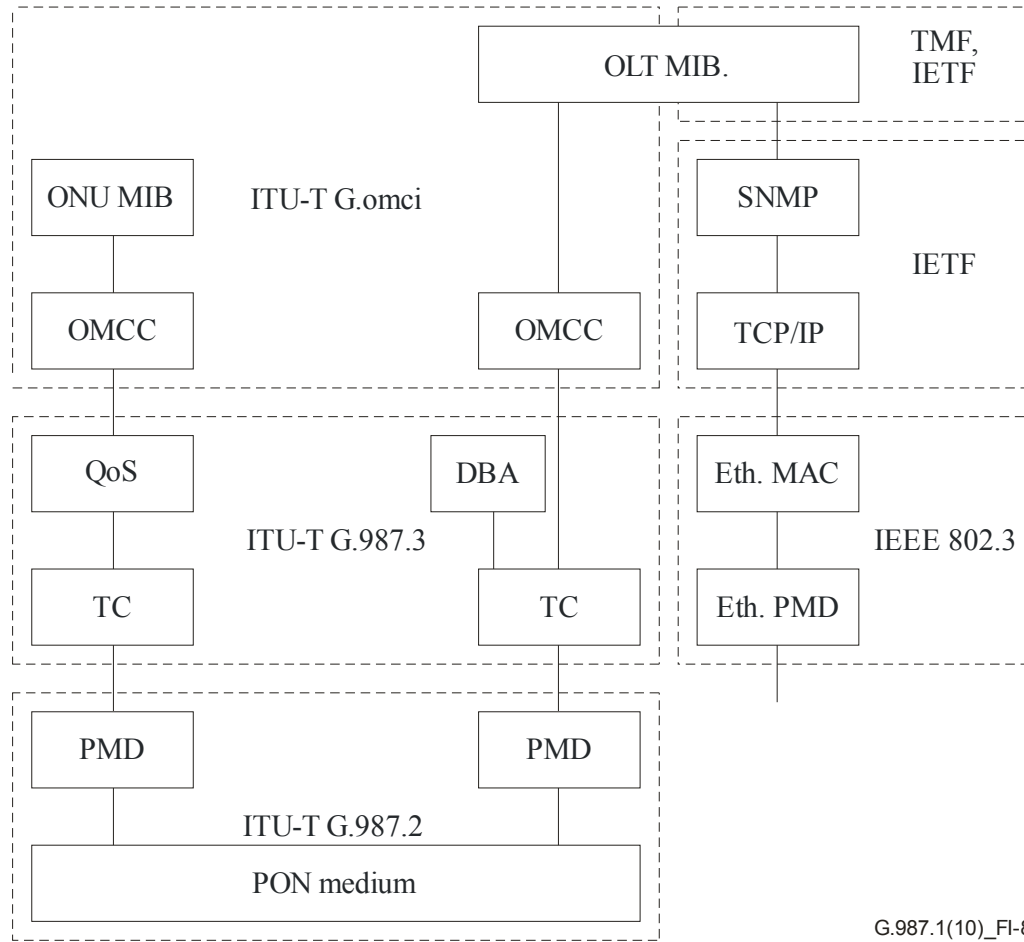
Figure 2.2: XG-PON common functions [43]

to 2.5 or 10 Gb/s; the maximum number of ONUs supported is increased (from 64 in GPON) to 256, and the physical reach is extended (from 20km in GPON) up to 60 Km.

2.2   XG-PON (G.987.x) standard by ITU-T

This section presents the details of the G.987.x standard (XG-PON) by ITU-T with specific attention to the protocol layers, QoS definitions and scheduling mechanisms of the standard.

XG-PON is defined in a series of recommendations in the ITU-T G.987.x specifications: ITU-T G.987 explains several important concepts of XG-PON; ITU-T G.987.1 presents the general requirements, services to be supported, hardware specifications and protocol stack of XG-PON as well as the network migration from and coexistence of XG-PON with GPON; ITU-T G.987.2 focuses on issues of the physical medium dependent (PMD) layer, such as the used wavelength and the supported data rates whereas ITU-T G.987.3 presents the details of transmission convergence layer for XG-PON (XGTC). Besides the

protocols for data communication, G.987.3 also covers the QoS management and the DBA scheme for the sharing of the upstream capacity. Another related recommendation is ITU-T G.988, which specifies an ONU management and control interface (OMCI) for both GPON and XG-PON. Figure 2.2 illustrates the common functions of XG-PON and the recommendations in which they are specified.

XG-PON has been proposed for various deployment scenarios to serve different customers, such as residential, business, and cell site; to serve these customers, XG-PON lists the services to be provided, such as the Telephony, high-speed Internet access, mobile backhaul, etc, while introducing many ONU variants that provide different functions and interfaces. As for the optical distribution network (ODN), XG-PON can be deployed as a classical PON with its reach up to 60km; to support longer physical reach, active reach extenders (RE) can be applied before and/or after the passive splitter to connect multiple passive segments belonging to a single XG-PON network. An XG-PON can be composed of multiple passive segments connected through active REs, which can be optical amplifiers or optical- electrical-optical regenerators that could fulfil the necessary optical link budget.

As seen in Figure 2.2, XG-PON has two primary layers in the PMD layer and TC layer.

2.2.1   Physical Medium Dependent (PMD) Layer

The PMD layer is the physical layer of XG-PON standard. There are two flavours of XG-PON based on the upstream line rate: XG-PON1 features 2.5 Gb/s and XG-PON2 10 Gb/s. The downstream line rate is 10 Gb/s in both XG-PON1 and XG-PON2. ITU-T G.987.2 focuses on the PMD layer for XG-PON1. XG-PON2 has only been standardized recently (June 2016) and is identified as Symmetric XG-PON or XGS-PON (ITU-T G.9807.1).

2.2.2   Transmission Convergence (XGTC) Layer

The Transmission Convergence layer of XG-PON (XGTC) contains the definitions of the MAC protocol of XG-PON. The XGTC layer maintains logical connections between OLT and each ONU, in pairs, in order to carry a downstream and an upstream traffic between the OLT and each ONU. Each connection is identified by a unique XG-PON encapsulation method (XGEM) Port-Id, which, while ensuring that packets are sent to/received from the correct ONU, associates every connection to a certain QoS agreement. One connection can be configured to carry either downstream or upstream traffic only. To reduce the overhead of the DBA scheme, upstream bandwidth is allocated to groups of clients connected to each ONU. These groups are designated as Transmission

Containers (T-CONT) and each group (or T-CONT) is identified by a unique identifier called *AllocId*.

XGTC comprises of three sublayers: service adaptation sublayer is at the top of the protocol stack, followed by the framing sublayer in the middle and the Physical adaptation sublayer near the PMD layer.

2.2.2.1   Service Adaptation Sublayer

The service adaptation sublayer is responsible for adapting the upper layer traffic to the transmission mechanisms of XG-PON. It does this by mapping the upper layer traffic to the corresponding connections, encapsulating/decapsulating data, segmenting/reassembling payloads or service data units (SDU) when necessary and inserting padding when there is insufficient data to fill an XGTC frame. If needed, it is also this sublayer's responsibility to encrypt/decrypt SDUs.

When mapping upper layer data to and from the connections of XGTC layer, while the OLT will maintain information pertaining to all the connections, an ONU will only maintain the connections that it owns. When the upper layer has something to transmit, it is also the service adaptation sublayer's responsibility to select the connections to be served according to their QoS parameters. When a connection is scheduled to be served, the service adaptation sublayer will then get data from its queue and insert an XGEM header to create an XGEM frame. The XGEM header will contain an XGEM Port-Id and some other information related to segmentation, padding, encryption, etc. When receiving an XGEM frame, the service adaptation sublayer will get the XGEM Port-Id from the XGEM header. If the connection corresponding to the XGEM Port-ID is part of the connections maintained by the OLT/ONU, this sublayer will carry out reassembly (if necessary) and pass the data to the upper layer. Otherwise, this XGEM frame will be discarded.

2.2.2.2   Framing Sublayer

The framing sublayer in XG-PON is responsible for generating and parsing the periodic XGTC frames/bursts. In XG-PON, the OLT sends downstream XGTC frames every 125 μs, to broadcast traffic to all ONUs. In the upstream, ONUs send variable-length XGTC bursts to the OLT for their upstream traffic. The length and synchronised start time of these upstream bursts are determined by the OLT through a DBA algorithm. When generating one downstream XGTC frame at the OLT, the framing sublayer gets XGEM frames from service adaptation sublayer and joins them together into an XGTC payload. To create an upstream XGTC burst at the ONU side, the framing sublayer may create multiple XGTC payloads, with each of the payloads carrying XGEM frames from a single
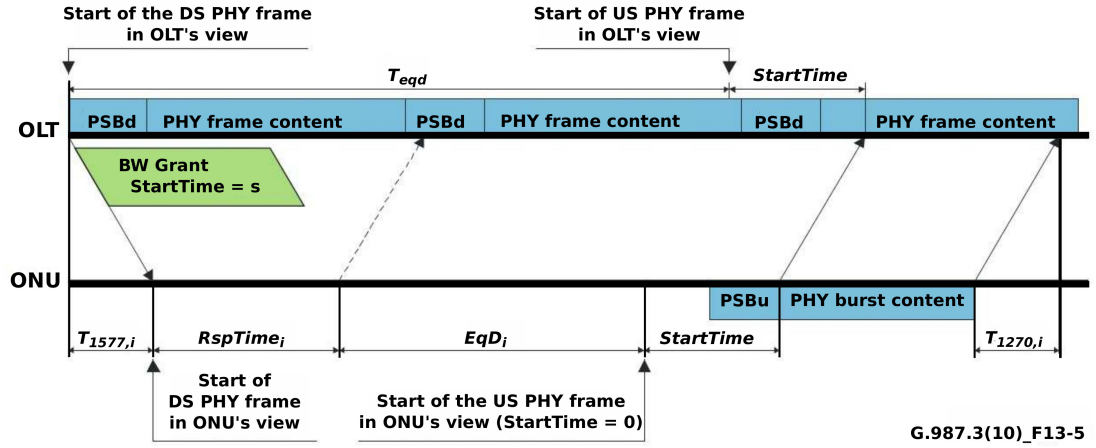
Figure 2.3: Time-line in XG-PON [43]

T-CONT. When parsing an XGTC frame/burst, the framing sublayer sends its payloads to the service adaptation sublayer for further processing.

The header of the downstream XGTC frame contains a field named $BWmap$ which contains instructions for upstream wavelength/bandwidth sharing in TDMA-line manner, between the different T-CONTs and ONUs as decided by the DBA. More specifically, $BWmap$ specifies the size of bandwidth allocations for T-CONTs, the used burst profile (the length of a preamble/delimiter, forward error correction used or not, etc.), and the time to start the data transmission. Since the OLT-ONU physical distance could be quite different for ONUs, each ONU should adjust the start time for avoiding collisions in the upstream direction, before sending its upstream XGTC burst, which may also contain the queue occupancy reports for its T-CONTs in the burst header. A detailed description of the QoS framework in XG-PON can be found in section 2.2.3.

2.2.2.3 PHY Adaptation Sublayer

The PHY adaptation sublayer interacts with the PMD layer directly. Its main functions are forward error correction (FEC), scrambling, and frame delineation through a Physical Synchronization Block (PSB). In the downstream, the PHY adaptation sublayer will get an XGTC frame to create a PHY frame. These PHY frames are sent continuously every 125 μs. In the upstream, the PHY adaptation sublayer will get the XGTC burst and create a PHY burst. These PHY bursts have variable length due to the variable-length XGTC bursts. In the PHY burst, the PSB is determined by a burst profile selected (through the $BWmap$) by the OLT, from the many burst profiles configured through the PLOAM messages.

2.2.3   QoS in (X)GPON

As part of the QoS specification, XG-PON (and GPON) should be capable of providing differentiated QoS for a range of traffic types in both directions. XG-PON defines the following portions of the link capacity for prioritisation among different traffic types:

- **Fixed Bandwidth** : Reserved portion of the link capacity, regardless of the demand and overall traffic load.

- **Assured Bandwidth**: Portion of the link capacity that is allocated to a flow that has unsatisfied traffic demand, regardless of the overall traffic conditions

- **Non-Assured Bandwidth**: Additional bandwidth the OLT dynamically assigns to an Alloc-ID in proportion to the ONU's total guaranteed bandwidth

- **Best Effort Bandwidth**: Additional bandwidth the OLT dynamically assigns to an Alloc-ID in proportion to the ONU's total non-guaranteed bandwidth.

Priority among the four bandwidth types is in descending order, with Fixed Bandwidth having the highest priority and Best Effort the least. XG-PON defines a fifth and optional class of traffic to facilitate decentralised (ONU-based) bandwidth allocation, where the OLT provisions upstream data transmission based on the aggregated queue status for all four classes of traffic above, such that the bandwidth allocation to each traffic class is at the discretion of the ONU.

XG-PON also classifies both the Fixed and Assured Bandwidth types as Guaranteed Bandwidth and the rest as Non-guaranteed Bandwidth. The standard commands that the DBA provide the guaranteed portion of bandwidth to all ONUs before allocating the non-guaranteed portion. That is, when the total upstream traffic load exceeds the capacity of XG-PON in the upstream, XG-PON requires that the non-guaranteed bandwidth types are deprived of upstream transmission opportunities before preventing the guaranteed bandwidth type traffic from the upstream transmission.

The XG-PON standard also states that a DBA mechanism can operate on the basis of either status reporting (SR) or traffic monitoring (TM) method, while being able to have the implementation of both methodologies in the XG-PON simultaneously. Since the SR mechanism is simple to implement and avoids over-provisioning of upstream bandwidth, most published DBAs use SR instead of TM.

2.2.4   Scheduling in XG-PON

Since downstream traffic is broadcast by the single OLT, scheduling in the downstream is a simple task handled by the OLT, requiring no specific guidelines from the standards. However, since several ONUs time-share the capacity of XG-PON in the upstream, the standard requires a polling-based scheduler in the upstream, in the form of a DBA
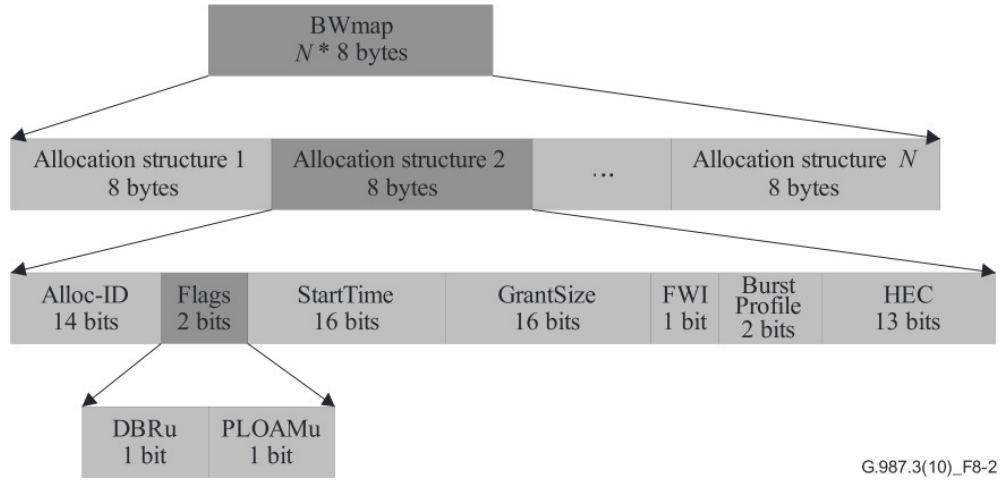
Figure 2.4: XG-PON downstream $BWmap$ for upstream data transfer [47]

mechanism, so that collision between the XGTC bursts from different ONUs can be avoided. That is XG-PON requires that the ONUs first send the queue occupancy reports to the OLT, after which the DBA can provide transmission opportunities to the ONUs. To inform the ONUs of the DBA decisions, with regard to the upstream data transfer from the ONUs and corresponding to a particular upstream frame, the OLT uses a header field called the $BWmap$ in the downstream physical frame. A $BWmap$ can have several *Allocation Structures*, each consisting of the following significant fields, as in Figure 2.4:

- *AllocID* : Identifier used by the OLT for the traffic flow(s) associated with each ONU.

- *Start Time*: The synchronised time[1] given to the *AllocID*, to begin transmission in a corresponding upstream frame.

- *GrantSize*: The amount of data allowed to be transmitted by the *AllocID* in the corresponding upstream frame

- *DBRu*: A flag used by the OLT to give permission to the *AllocID* to request for data for the next upstream frame

Given a scenario that an *AllocID* does not have any $GrantSize$ provision in a particular $BWmap$, the minimum value for the $GrantSize$ is defined to be 1 word (4 bytes) in XG-PON, to allow the *AllocID* to request for upstream transmission opportunity in the subsequent upstream frames.

Based on the queue occupancy reports sent by the ONUs in an earlier upstream frame, and the QoS policies defined, the DBA is required to make the decisions with regard to the time-sharing nature of the XGTC bursts of different ONUs in the upstream. The DBA

---

[1]Due to the differences in the distances of ONUs from OLT, this is a synchronised time, achieved by the ranging procedure of XG-PON
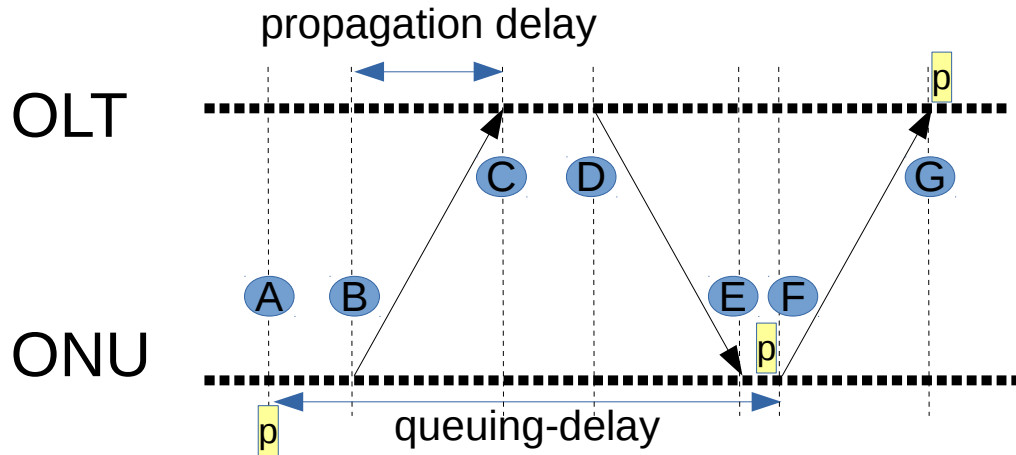
Figure 2.5: Delays experienced by a packet arriving at the ONU for upstream transmission

algorithm, generally implemented in an OLT is therefore assigned with the following tasks before the transmission start time of the corresponding upstream frame: selecting the ONUs to be served in an upstream frame; reserving a short gap time (guard band) between the consecutive XGTC bursts for tolerating clock synchronization errors; determining the size of the XGTC burst per ONU; calculating the transmission start time for each ONU; and sending the downstream frame (with decisions of the DBA in the $BWmap$ field) to the ONUs.

Figure 2.5 shows all the possible delays experienced by a packet (p) from the time it enters an ONU queue (point A) till it reaches the OLT after being transmitted in the upstream (point G). Circles A-G represent the relative (to A) points in time where the following events occur:

- A - p enters an ONU queue

- B - The transmission of the $125\mu s$-periodic upstream, which carries the queue occupancy report from an ONU.

- C - The upstream frame with the queue occupancy report reaches the OLT after a fixed propagation delay ($delay_{propagation,up}$).

- D - The $125\mu s$-periodic downstream (broadcast) frame is sent from the OLT, carrying the decision of transmission opportunity provision for the ONU, after a processing delay of $delay_{processing,olt}$

- E - The downstream frame from OLT reaches the ONU, after a propagation delay of $delay_{propagation,down}$.

- F - ONU transmits the packet, p after a processing delay of $delay_{processing,onu}$.

- G - The transmitted packet, p reaches the OLT after $delay_{propagation,up}$.

With with assumptions that a fixed and equal propagation delay ($delay_{propagation} = delay_{propagation,up} = delay_{propagation,down}$) is experienced in the upstream and downstream directions, the total delay ($delay_{Total}$) experienced by the packet, between arriving at the ONU queue and reaching the OLT can be given by:

$$delay_{Total} = 3 * delay_{propagation} + delay_{processing,onu} + delay_{processing,olt} \qquad (2.1)$$

However, as seen in the figure, between circles A and F, p merely stays in the ONU queue while the scheduler at the OLT is busy with its operation. Hence, $delay_{Total}$ can be represented in terms of the queuing-delay, $delay_{queue}$ as,

$$delay_{Total} = delay_{queue} + delay_{propagation} + delay_{processing,onu} \qquad (2.2)$$

where,

$$delay_{queue} = 2 * delay_{propagation} + delay_{processing,olt} \qquad (2.3)$$

The above scenario is a highly simplified state of scheduling in XG-PON. However, in reality, since the XG-PON is able to support several ONUs, with each ONU having more than one logical upstream connection, an actual scheduling and provision mechanism in the XG-PON can be complex and require efficient scheduling policies in the DBA. Hence, to generate healthy competition among vendors and encourage research, the details regarding the implementation of an efficient DBA algorithm were intentionally omitted from the scope of the standards. When such a vendor-implemented DBA follows the basic QoS frameworks given in the XG-PON standard, the DBA can be referred to as a (XG-PON) standard-compliant DBA.

2.3   Long Term Evolution (LTE)

This section presents the details of LTE, with special attention to the standardisation by Third Generation Partnership Project (3GPP) and the QoS definitions in LTE.

LTE, a cellular technology, is the latest standard of 3GPP. LTE is also known as the fourth generation mobile network (4G) and the standardisation started with 3GPP's Release 8. The goals of 3GPP, which began work on LTE with its Radio Access Network (RAN) Evolution Workshop in Toronto, Canada in November 2004, was to design a system that can surpass the older mobile standards (e.g. GSM in 2nd Generation, UMTS in 3rd Generation and HSPA between the 3rd and 4th Generation, etc.) and that can still stay competitive at least for the next decade[100].

At a basic level, LTE defines the following major components in its architecture:

- Evolved Packet Core (EPC): This comprises of all the significant components of the core network portion of the mobile communication in LTE. Major components of EPC are the Mobility Management Entity (MME), which assists mobility of a user equipment (UE), the Serving Gateway (SGW), which routes user data packets while acting as a mobility anchor for intra-LTE mobility, and the Packet Data Network Gateway (PDN-GW), which provides connectivity for the UE towards an external data networks such as the Internet.

- Evolved Node B (eNB): An eNB is the base station in LTE, which comprises the access network of LTE. It also interfaces the UE and the SGW for data packets (user-plane) and UE and MME for signalling messages (control-plane).

At the PHY layer, LTE introduced the Orthogonal Frequency-Division Multiplexing or OFDM-based multi-carrier access technology such that the Orthogonal Frequency Division Multiple Access (OFDMA) was chosen for the LTE's downlink (or downstream) which is from the eNB to a UE; an OFDM variant, the Single-Carrier FDMA (SC-FDMA) was chosen for the uplink (or upstream) which is from a UE to the eNB. Use of OFDM in LTE provided several benefits to LTE, such as support for dynamic frequency reuse techniques and enhanced system throughput, it also enabled LTE and its evolution (LTE-Advanced) with the feasibility of several attractive PHY layer features such as advanced Multi-Input-Multi-Output techniques, Carrier Aggregation , Coordinated Multi-Point transmission/reception and relaying[100].

In the air interface (between eNB and UE), both LTE and LTE-Advanced support TDD and FDD duplexing modes. In FDD, different frequency bands are utilized for the downlink and uplink transmissions, while in TDD the downstream and upstream share the same frequency bands but are separated in time. All transmissions are organized into radio frames of 10 ms each, with each frame further divided into ten equally sized subframes. In turn, a subframe is divided into two equally sized slots of 0.5 ms. For both downstream and upstream, and independent of the duplexing mode utilized, the base LTE radio resource is defined as a time-frequency resource block that spans 0.5 ms in time and 12 contiguous OFDMA/SC-FDMA sub-carriers[3].

The objectives of LTE MAC layer are mapping between logical and transport channels, multiplexing and scheduling MAC segments, relaying scheduling information, error correction (HARQ), and priority handling. LTE packet scheduling in the eNB executes resource allocation decisions periodically once every 1 ms, which is defined as Transmission Time Interval (TTI)[3]. The challenge for an LTE air-interface scheduler lies in distributing the limited radio resources to maximize the system performance (throughput, fairness, etc.) in the air-interface, for multiple UEs. Due to the maturity of LTE network both in industry and the literature, several schedulers exist today for the uplink scheduling in LTE. This thesis uses the Proportional Fair Scheduler (PFS[57]) for
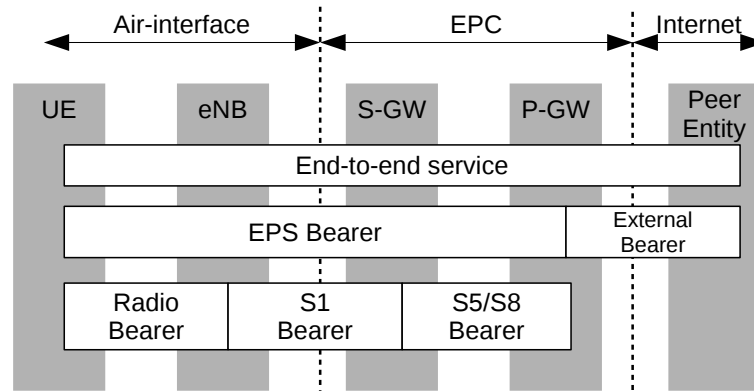
Figure 2.6: Bearers in 3GPP LTE[82]

the scheduling of upstream air interface resources among multiple users since the PFS is capable of providing a good balance between the throughput and fairness between multiple UEs attached to an eNB. That is in PFS, the selection of the UE to schedule in the next time slot or TTI is based on a balance between the current possible rates of and fairness among all the UEs. PFS performs this by comparing the ratio of the feasible rate for each UE with the UE's average throughput, tracked by an exponential moving average, which is defined as the preference metric. The user with the maximum preference metric will be selected for transmission at the next TTI.

2.3.1 QoS in LTE

The QoS framework in LTE is designed to provide a fine-grained QoS classification and treatment for each application flow between a UE and the Core Network of LTE (or EPC). LTE defines several bearers or virtual QoS classifications, based on the scope of the entities involved in establishing and maintaining the bearers (see Figure 2.6). Between a UE and the P-GW in the EPC, EPS bearers control the QoS classification of the traffic flows. This is done hierarchically by defining additional classification of bearers at each hop between the UE and the P-GW: for the LTE air-interface (between a UE and the eNB) *radio bearers* are used; between the eNB and the S-GW, *S1 bearers* are used; and between the S-GW and the P-GW, *S5/S8 bearers* are used. LTE defines a one-to-one relationship between an EPS and Radio bearer.

QoS classification between the application flows is provided by classifying bearers into different types, and then assigning the traffic flows to the required bearer types. At the high-level, LTE defines two major types of bearers, namely Guaranteed Bit Rate (GBR) and non-GBR bearers. LTE also classifies bearers based on their scope, such as default and dedicated bearers at the EPS bearer level. A default bearer, which is initiated and established for a UE at the time of connecting to an eNB (the Attach procedure), is always a non-GBR bearer that does not provide bit rate guarantees; a dedicated bearer,

Table 2.1: QCI Values in 3GPP-TS-23.203 Release 8 [1]

QCI	Type	Priority	PDB	PEL	Example Services
1	GBR	2	100ms	10^{-2}	Conversational Voice
2	GBR	4	150ms	10^{-3}	Conversational Video
3	GBR	3	50ms	10^{-3}	Real Time Gaming
4	GBR	5	300ms	10^{-6}	Non-Conversational Video (Buffered Streaming)
65	GBR	0.7	75ms	10^{-2}	Mission Critical user plane Push To Talk voice (eg: MCPTT)
66	GBR	2	100ms	10^{-2}	Non-Mission-Critical user plane Push To Talk voice
5	non-GBR	1	100ms	10^{-6}	IMS Signalling
6	non-GBR	6	300ms	10^{-6}	Video (Buffered Streaming) TCP-Based (for example, www, email, chat, ftp, p2p and the like)
7	non-GBR	7	100ms	10^{-3}	Voice, Video (Live Streaming), Interactive Gaming
8	non-GBR	8	300ms	10^{-6}	Video (Buffered Streaming) TCP-Based (for example, www, email, chat, ftp, p2p and the like)
9	non-GBR	9	300ms	10^{-6}	Typically used as default bearer
69	non-GBR	0.5	60ms	10^{-6}	Mission Critical delay sensitive signalling (eg: MCPTT signalling)
70	non-GBR	5.5	200ms	10^{-6}	Mission Critical Data (e.g. example services as in QCI 6/8/9)

which is associated with a UE in addition to the default bearer, can either be a GBR or a non-GBR bearer. Bearers configured for a UE are characterized by a Traffic Flow Template with QoS parameters associated to it[3].

LTE groups bearers into several more classes, each of which is identified by a scalar number called the QoS Class Identifier (QCI). QCI also indicates a predefined group of QoS parameters describing the packet forwarding treatment in terms of priority, tolerated delay (or packet delay budget, PDB), and tolerable packet error loss (PEL) for each traffic flow. Table 2.1 shows the QCI classifications in the latest (Release 13) 3GPP specifications (TS-23.203 [1]).

2.4 Packet scheduling methodologies in wired networks

Scheduling in telecommunication networks, both wired and wireless, is a very mature concept in the literature with numerous packet schedulers designed to suit many technologies and many more scenarios.

This section outlines some of the popular scheduling methodologies in the wired networks as they will play a role in designing and tuning the (upstream) DBA mechanisms for XG-PON in the context of being a dedicated backhaul for LTE. In XG-PON, OLT is the upstream scheduling controller which receives queue occupancy requests from many ONUs, making the upstream nature of XG-PON a packet switched single-server model with error-free channels[95] (assuming an error free link between the OLT and the ONUs); each ONU may have more than one T-CONT to serve different classes of aggregated upstream traffic from LTE, while adhering to the QoS policies of the XG-PON standard. Ideally, XG-PON requires a QoS-aware scheduler which is capable of guaranteeing delay and throughput metrics for the aggregated LTE traffic arriving in each ONU's T-CONT queues, while ensuring strict priority and fairness. However, the com-
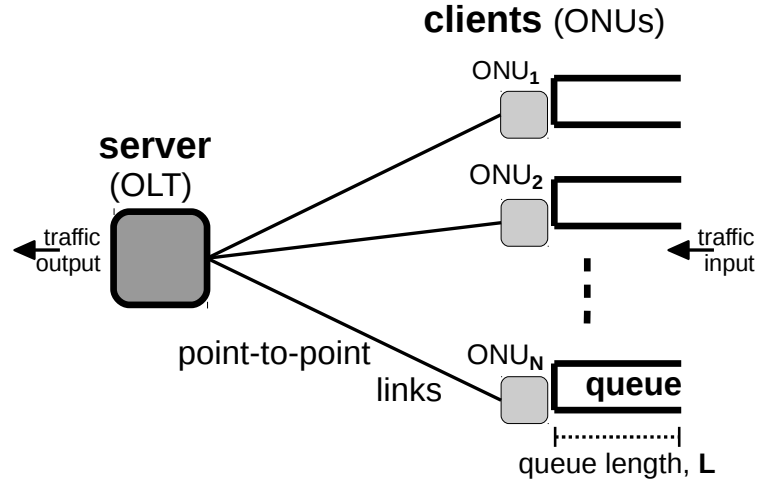
Figure 2.7: Server-client set-up equivalent to XG-PON upstream

plex nature of integration between XG-PON and LTE can be spared for additional complexity only when using a simple-to-implement dynamic scheduling algorithm which can be executed at the OLT within the small time-frame (125μs) of each downstream frame, which carry the result of DBA calculation to the ONUs in the form of $BWmap$.

Although the scope of this section is not to design an ideal (optimal and simple) scheduler for the XG-PON to serve different classes of LTE traffic in the backhaul, the thesis expects a reader to be aware of popular scheduling mechanisms to understand the principles used later for the DBA design. A few popular scheduling algorithms are explained in the remaining section to provide an insight into the relevance of schedulers in designing a fine-tuned DBA for the dedicated XG-PON backhaul in LTE.

2.4.1 Round Robin Scheduler

The Round Robin (RR) scheduler is one of the oldest, simplest, fairest and most widely used scheduling algorithms, designed especially for TDMA-based systems.

Illustrating the logical view for scheduling in XG-PON, Figure 2.7 gives the scenario of a single server (OLT) serving several clients (ONUs), with each client having a single queue, of length L to store packets arriving from a traffic source of unknown packet distribution. A RR scheduler at the server would visit the clients in a given order (eg: ONU_1, ONU_2, ..., ONU_N) to provide a transmission opportunity to each client across the point-to-point links. The transmission opportunity given by RR is limited, in general, only by the queue length (L) of the client, and in XG-PON, also by the upstream frame capacity which is of 125μs or 38880 Bytes long for a total line rate or bandwidth of 2.48832 Gb/s.

When system bandwidth (R_{total}, in bps) is the only limiting factor with a relatively unlimited queue length in the clients, the transmission rate given to a client, i (where

$1 \leq i \leq N$) by the RR scheduler can be given by,

$$R_{i,RR} = R_{total}/N \tag{2.4}$$

Because RR visits every client sequentially and is limited by parameters not related to the traffic arrival nature in the queues, the scheduler utilises bandwidth (capacity) efficiently while being fair to all the clients. An implementation of RR scheduler in XG-PON can be seen in Chapter 4 of this thesis.

2.4.2 Priority Queuing

The Priority Queuing (PQ) scheduling algorithm provides a simple method of supporting multiple (differentiated) service classes in a system. After a classification scheme has classified the packets and placed them into different queues with different priorities the PQ algorithm handles the scheduling of the packets following a priority-based model. Packets are scheduled to be transmitted from the head of a given queue only if all queues of higher priority are empty. So the different levels of priority assigned to queues introduce unfairness based on how queues are serviced. However, inside a single priority queue, packets are still scheduled in a First-In-First-Out (FIFO) order, as with any other queuing and scheduling method.

However, PQ also has its own set of limitations. If the volume of high- priority traffic becomes excessive, lower-priority queues may face complete starvation. When the queuing rate remains constant but the rate of removal from the queue decreases, packet drops would be inevitable in the queues. The drop rate for traffic placed in low-priority queues increases as the queue space allocated to the low-priority queues starts to overflow. The result is not only packet drops but also increased latency. In addition, misbehaving high-priority flows can also add delay and jitter to other high-priority flows that share the same queue. As the number of classes increase, a simple PQ scheduler may not be sufficient to guarantee the QoS requirements of all the classes while ensuring fairness and avoiding lower-priority starvation in the system.

2.4.3 Weighted Round Robin

RR, despite all its advantages, is not capable of providing different QoS-based treatment to different clients by having class-based differentiation in the system, a feature also known as *flow isolation* in many scheduling contexts. *Flow isolation* is compulsory in the scheduler to provide differentiated QoS treatment in XG-PON upstream. Though PQ provides flow isolations, it introduces limitations such as low-priority starvation.

Weighted Round Robin (WRR) scheduler, which is an approximation of the ideal Generalised Processor Sharing[77] scheduler, is an example of a scheduler which is capable of ensuring flow isolation and trouble-free priority in packet-based systems. That is, for each client in the same system in Figure 2.7, WRR would first assign a weight, w_i (for ONU_i), to enable different clients receiving a weighted share ($R_{i,WFQ}$) of the system bandwidth (R_{total}).

$$R_{i,WFQ} = \frac{w_i}{w_1 + w_2 + ... + w_N} R_{total} \qquad (2.5)$$

The weight in WRR is usually a percentage of the system bandwidth, thereby reflecting the service differences between the queues and the traffic classes assigned to those queues. When the weights of all the clients are equal ($w_i = w_j \; for \; i \neq j; 1 \leq i, j \leq N$), the WFQ scheduler shrinks into a simple RR scheduler. WRR can also be scaled to serve multiple queues per client with different queues of each client having different weights to reflect the priority of the queues within each client and eventually within the system itself.

Due to its simplicity and capability, WRR is well-suited for handling a large number of flows and sessions, thereby enabling the scheduler to be something of a core QOS solution that can deal with large volumes of traffic and numerous congestion states. However, since WRR is unaware of how different packet lengths are scheduled, scheduling can be unfair when queues have different sized packets. In WRR, services that have a very strict demand on delay and jitter can also be affected by the scheduling order of other queues as WRR offers no priority levels in its scheduling.

2.4.4 Deficit (Weighted) Round Robin Scheduler

Shreedhar and Varghese[87] proposed the Deficit Round Robin scheduler (also known as the Deficit Weighted Round Robin or DWRR scheduler) to mitigate the limitations found in WRR, especially regarding the variation in packet lengths.

DWRR introduces a deficit counter in conjunction with a proportionate weight (or *quantum*) to eliminate the unfairness caused by packet size variation. The difference between the *quantum* and the queue occupancy forms the basis for the deficit counter. While *quantum* is the throttle mechanism that allows the scheduling to be aware of the bandwidth, the deficit can be positive or negative; positive deficit occurs when, at the end of a scheduler turn, there are leftover bytes that were not used in the queue's scheduling turn. This value is deferred to the queue's next scheduling turn; negative deficit accrue when the queue has transmitted more than its bandwidth value in a scheduling turn, and thus the queue is in debt when the next scheduling turn comes around. The deficit counter, which provides bandwidth fairness, is the sum of the

quantum and the deficit. The scheduler removes packets from a client's queue until the queue's deficit counter reaches a value of zero or until the size of the next packet is larger than the remaining deficits. But if the queue does not have enough deficits to schedule a packet, the value of the deficit is retained until the next scheduling round, and the scheduler moves to the next queue in the system.

However, DWRR still remains vulnerable to other limitations of WRR: services that have very strict demand on delay and jitter can be affected by other queues by the scheduling order in DWRR, which is also incapable of explicitly prioritising its scheduling.

2.5   Transport Layer Protocols: UDP and TCP

For IP-based traffic in today's Internet, User Datagram Protocol (UDP) and Transport Control Protocol (TCP) are two significant and popular protocols at the Transport Layer of the TCP/IP protocol stack. UDP is generally used in real-time applications such as Voice-over-IP (VoIP) and video conferencing due to its simplistic nature in data transmission procedures while incurring only a small percentage of overhead for data transmission. However, TCP is used in applications requiring guaranteed delivery of packets (eg: File Transfer Protocol or FTP, and Hyper Text Transfer Protocol or HTTP) due to the connection-oriented nature of TCP which also uses flow control. This section explains the nature of TCP and the congestion control mechanisms which are later used extensively in the evaluations presented in this thesis.

2.5.1   Transport Control Protocol (TCP)

TCP is best known for its 3-way handshake and congestion control mechanisms, which are used to guarantee the end-to-end (E2E) delivery of an application data across challenging networks. When a sender wishes to send application data to a receiver, TCP first uses the 3-way handshake mechanism between the Transport Layers of the sender and the receiver, to establish an E2E TCP flow/connection, before the data is transmitted across the established TCP flow. Then application data is relayed over the TCP connection using the congestion control mechanism based on the Additive Increase Multiplicative Decrease (AIMD) principle. AIMD is a feedback control mechanism which adheres to two different flavours of the sender rate to maintain control and fairness in network capacity utilisation: linear growth is used when the in-between network has sufficient capacity and exponential reduction is used when network capacity is less than the aggregated sending rate of the sender(s). The challenge in TCP is to manage a reasonable sending rate, using a congestion control mechanism, so the throughput performance between the sender and receiver can be justified at any time during the data transmission.
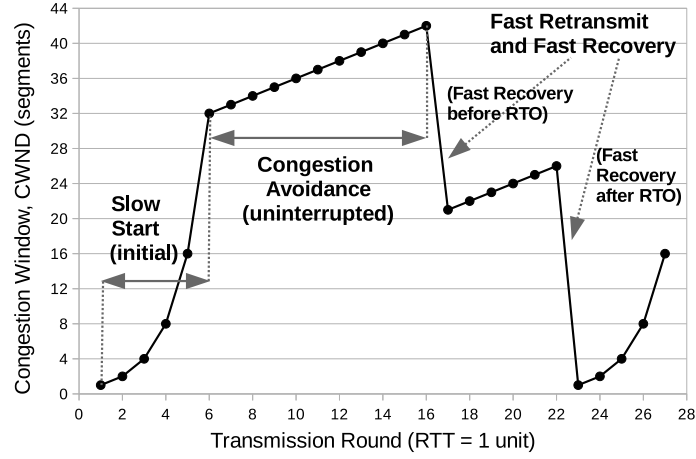
Figure 2.8: Phases of TCP NewReno

2.5.2   Congestion Control Algorithms in TCP

Several congestion control algorithms, starting from its earliest TCP Tahoe[52] to the recent high-speed variations such as the Hamilton TCP (H-TCP), have been proposed for TCP to address different scenarios of congestions occurring in different types of networks. Each algorithm differs in how the congestion window ($CWND$) at the Transport layer of the sender is varied when transmitting TCP segments depending on network conditions at the receiver and the in-between network. TCP NewReno, CUBIC TCP and H-TCP algorithms are briefly introduced here due to their relevance in chapters 5 and 8. An interested reader is referred to the publications of Marrone et al. in [66] and Leith et al. in [26] for a detailed description of these congestion control algorithms.

2.5.2.1   TCP NewReno

TCP NewReno[30] is the improved version of the TCP Tahoe and TCP Reno[92] congestion control algorithms. Today NewReno is a highly stable and popular congestion control algorithm used in real-word TCP stacks. NewReno has four distinct phases. as indicated in Figure 2.8:

- Slow Start: TCP begins data transmission starting with a $CWND$ value of 1 and increasing as below, for every ACK(nowledgement) received from the receiver.

$$CWND \ = \ CWND \ + \ 1$$

  Since $CWND$ is increased every time an ACK is received, the actual $CWND$ doubles for every round-trip-time (RTT) -long time interval, resulting in an exponential increment of $CWND$, until a threshold ($ssthresh$) is reached in the slow start phase.
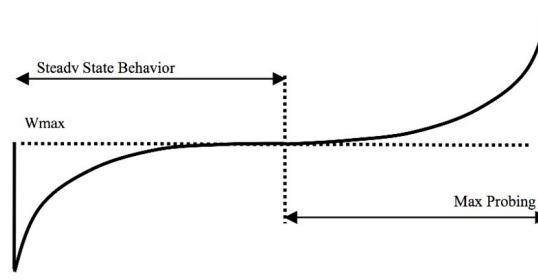
Figure 2.9: $CWND$ growth of a CUBIC TCP flow as per equation 2.6[9]

- Congestion Avoidance: Beyond $ssthresh$, $CWND$ is incremented as follows, to show a linear and comparatively slower increment of $CWND$ rather than an exponential increment in the slow start phase.

$$CWND = CWND + 1/CWND$$

- Fast Retransmit and Fast Recovery: Fast Retransmit phase is initiated by the sender when a packet loss is detected by the reception of three duplicate ACKs (in addition to the original ACK) at the sender. In this case, only the lost packet, and not the rest of the packets, is retransmitted before the retransmission time-out (RTO) occurs at the sender, who then continues with the congestion avoidance phase. In fact, this phase was part of the TCP Tahoe as well.

  Fast Recovery defines the procedure followed by the TCP sender immediately after the fast retransmit procedure. The following series of actions are performed in fast recovery.

  1. First, the $ssthresh$ is reduced to the value of $CWND/2$ followed by the new $CWND$ value being set to $ssthresh + 3$

  2. Thereafter $CWND$ is increased by 1 for every new duplicate ACK received.

  3. When the first non-duplicate ACK arrives, $CWND$ is reset to $ssthresh$ and the congestion avoidance phase is initiated.

  4. However, if RTO occurs while waiting for the first non-duplicate ACK, then the slow start phase is initiated.

2.5.2.2 CUBIC TCP

CUBIC TCP[33], a common congestion algorithm used in the UNIX-based clients and servers, was designed to take advantage of networks with large bandwidth-delay product (BDP). It differs from TCP NewReno mainly in the congestion avoidance phase and after the fast recovery phase. Once TCP segment loss is detected in the congestion

avoidance phase in CUBIC, Fast Recovery phase is initiated with the $CWND$ value first recorded as W_{max} and then decreased by a factor β. From that time a new $CWND$ value, W_{CUBIC} is calculated as[90]:

$$W_{CUBIC} = C(t - K)^3 + W_{max} \qquad (2.6)$$

where C is a scaling factor, W_{max} is the window size just before the last window reduction, t is the time since the last window reduction and $K = \sqrt[3]{W_{min} \star (\beta/C)}$, with W_{min} being the reduced window size just after the last congestion event. After a congestion occurs, CUBIC is first in the $steady - state$ (Steady state behaviour in Figure 2.9) where W_{CUBIC} grows concavely up to W_{max}, after which it enters $exploratory - state$ (Max Probing in Figure 2.9) and grows convexly until the next congestion event, in order to determine currently available network bandwidth.

2.5.2.3   H-TCP

H-TCP [26], another high-speed variant of TCP, uses a $\triangle$ (the elapsed time since the last congestion event) to adjust the speed of increasing $CWND$. Incrementation steps of $CWND$ are varied as a function of $\triangle$, which is adjusted to improve link utilization based on an estimate of the queue provisioning, while being scaled with RTT to mitigate unfairness between competing flows with different RTTs. In more detail[60],

$$CWND \leftarrow CWND \ + \ \frac{2 \star (1 - \beta) \star f_\alpha(\triangle)}{CWND}$$
$$CWND \leftarrow g_\beta(B) \star CWND$$

with,

$$f_\alpha(\triangle) = \begin{cases} 1, & \triangle \leq \triangle_L \\ max(f_{\alpha(\triangle)T_{min}}, 1), & \triangle > \triangle_L \end{cases}$$

$$g_\beta(B) = \begin{cases} 0.5, & |\frac{B(k+1)-B(k)}{B(k)}| > \triangle_B \\ min(\frac{T_{min}}{T_{max}}, 0.8), & otherwise \end{cases}$$

where $\triangle_L$ is a specified threshold such that the standard TCP update algorithm is used during a short period immediately after a congestion event. A quadratic increase function f_α is suggested in [26], namely $f_\alpha(\triangle) = 1 + 10(\triangle - \triangle_L) + 0.25(\triangle - \triangle_L)^2$. T_{min} and T_{max} are measurements of the minimum and maximum RTT experienced by a flow. $B(k + 1)$ is a measurement of the maximum achieved throughput during the last congestion epoch.

Being a latest high-speed variant of TCP, H-TCP is capable for achieving very large values of sending rates, even when using a single TCP flow in the downstream and upstream directions of XG-PON. Hence, H-TCP is used, along with NewReno and CUBIC, in Chapter 5 to evaluate the behaviour of application flows across XG-PON with regard to the influence of large-BDP behaviour in the different congestion control algorithms of TCP.

2.6   ns-3 network simulator

This section presents the ns-3 [2] network simulator and its existing modules which form the basis for the simulation platform in Chapters 4- 8.

ns-3 is a state of the art open-source network simulator. Based on many lessons from the well-known ns-2 simulator, ns-3 is written from the scratch, with no backward-compatibility with ns-2. The simulation core and modules of ns-3 are implemented in C++ as a library which may be statically or dynamically linked to a C++ main program. ns-3 also exports nearly all of its API to Python, allowing Python programs to import an "ns3" module in much the same way as the ns-3 library is linked by executables in C++. ns-3 is targeted primarily for research and educational use, being a free software, licensed under the GNU GPLv2 license, and is publicly available for research, development, and use.

ns-3 has many attractive features, such as a real-time scheduler that facilitates a number of "simulation-in-the-loop" use cases for interacting with real systems, emphasis on the reuse of real application and kernel code, good support for virtualisation and testbeds, a novel attribute system for configuring simulation parameters, automatic memory management, a configurable tracing system and the ability to incorporate existing open-source module through simple APIs[37]. It has also been reported that ns-3 performs much better than other simulators in terms of simulation speed and memory overhead [96]. The first release of ns-3 was made in June 2008 with support for a number of modules including CSMA, Point-to-Point, WiFi (IEEE 802.11), TCP, UDP and IP version 4 (IPv4). In the last few years, many new modules have been developed and added into ns-3, such as WiMAX module from Inria [28] and LTE module from CTTC [79]. ns-3 also consists of several application modules that can be used to and build customer applications which can either use TCP or UDP in their Transport layer and IP version 4 at the Network layer.

Chapters 4- 8 of this thesis, however, uses only the simulation and not the 'real-time' features of ns-3 to set up the experiments. In addition to the core of the ns-3 simulator itself, this thesis uses following modules of ns-3.
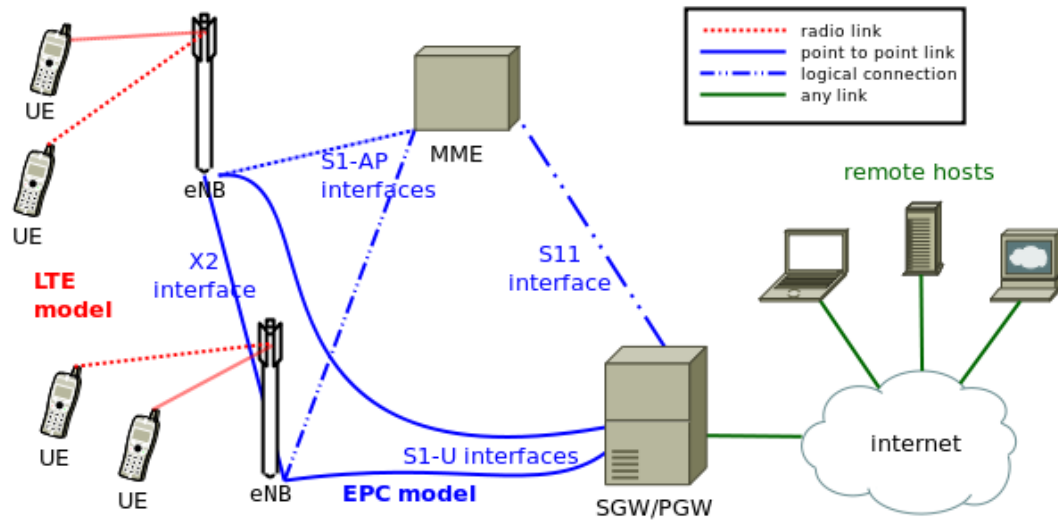
Figure 2.10: Overview of the LTE architecture in the LENA module in ns-3[25]

2.6.1 LTE Simulation Module

The LTE module [16] for ns-3, also known as LTE-EPC Network Simulator (LENA), is an open-source product-oriented LTE-EPC network simulator that allows the small/macro cell vendors of LTE to design and test algorithms and solutions for Self Organized Networks.

Figure 2.10 shows a simplified architecture of LENA in ns-3. LENA provides a very detailed Evolved Universal Terrestrial Radio Access Network which includes models of both user plane (i.e., Packet Data Convergence Protocol, Radio Link Control, MAC and PHY) and control plane protocols (i.e., Non-access stratum and Radio Resource Control); PHY layer enables fading and propagation models[71]; features such as Radio Resource Management, QoS-aware packet scheduling, inter-cell interference coordination and dynamic spectrum access are designed in detail in LENA, which also consists of several MAC schedulers (RR, PFS, etc.).

A simple EPC model is provided in LENA to facilitate E2E IP (version 4) connectivity over the LTE model. EPC supports the interconnection of multiple UEs to the Internet, via the many eNBs connected to a single and combined SGW/PGW node. Though EPC models data (S1-U) and control planes (S1-AP, X2-AP and S11) separately, the combined SGW/PGW is a simplified model without the need for S5 or S8 interfaces. Figure 2.11 shows the protocols involved in between the UE and a remote host in the Internet, connected via the eNB and SGW/PGW. As in the figure, S1-U interface is a simplified model supported by the GPRS Tunnelling Protocol, UDP and IP protocols in the data plane.
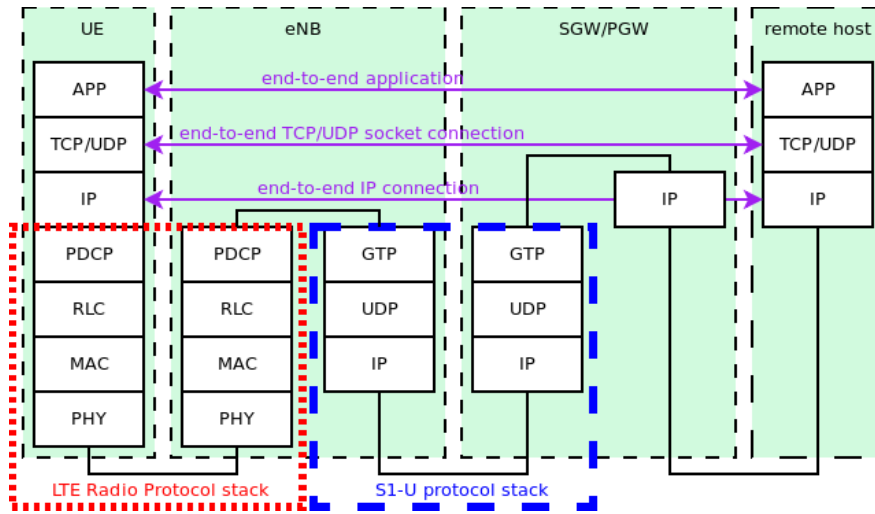
Figure 2.11: LTE-EPC data plane protocol stack[2]

2.6.2 Application modules in ns-3

In order to simulate the behaviour of realistic applications, ns-3 provides several modules at the Application Layer, to be used with common (UDP and TCP) Transport Layer protocols. This thesis uses *OnOffApplication*, *BulkSendApplication* and *PPBP*[7] application modules of ns-3 extensively to generate traffic for the evaluations, depicting realistic applications.

2.6.2.1 OnOffApplication module

The *OnOffApplication* [2] module generate traffic using an ON-OFF traffic model, with the application generating traffic during the ON period and remaining idle (no packet generation and transmission) during the OFF period. The application module also supports several parameters such as packet size and datarate to be used during the ON period, while supporting several distribution models (Uniform, Log Normal, etc.) for ON and OFF period durations *OnOffApplication* be used to generate application traffic representing VoIP, and with minor changes, the constant/variable bit rate Video applications.

2.6.2.2 BulkSendApplication module

The *BulkSendApplication* [2] module in ns-3 is designed to generate a predefined amount of data at the application layer and send the data as soon as possible to the lower layer until the application is stopped for data transfer. The ability of the *Bulk-SendApplication* application module to pause, restart and stop packet generation at the application layer based on the feedback from lower layers (eg: buffer limitations from

TCP), enables this module to be a good application traffic generator for realistic scenarios of TCP-based FTP file transfer in the communication networks.

2.6.2.3   PPBP application module

Ammar et al. [7] recently presented the long-range dependent Poisson Pareto Burst Process (PPBP) -based module, which can generate realistic Internet traffic on the basis of aggregating several individual bursts, each of whom is a Poisson process with a constant bit-rate and a Pareto-distributed burst lengths characterised by the Hurst parameter. The Hurst parameter, H, is commonly used in the literature to tune the burstiness of an application traffic model to represent realistic Internet traffic models (eg: mobile Internet). Since the PPBP application module models long-range dependency accurately, has easily-configurable interface for ns-3 and uses low computational and memory resources in the simulation platform, this thesis uses PPBP module in many of its applications such as peer-to-peer video and mobile Internet traffic in large-scale and complex network conditions.

2.7   Chapter Summary

This chapter presented the technologies and terminologies which form the basis of the subsequent chapters of this thesis. Major technologies such as the XG-PON and LTE were discussed in detail, with special attention towards their standards and QoS framework at the MAC layer. The basic scheduling methodologies, common TCP algorithms and the simulation modules used in the sphere of communication networks were also presented in detail, as the contributions of the main focus of the thesis is to present the design, implementation and evaluation (simulation-based) of the scheduling-based DBA mechanisms in the context of XG-PON backhaul in LTE, with support for both UDP-based and TCP-based application traffic in realistic network scenarios.

Chapter 3

Literature Review

This chapter reviews the published literature which relates to the contributions presented in this thesis. Section 3.1 presents an overview regarding the convergence of PON and LTE. Section 3.2 provides the challenges from the literature regarding the FMC of XG-PON and LTE, especially with regard to the converged network architecture and common QoS framework. Section 3.3 presents the work and challenges in the literature in terms of designing a standard-compliant DBA for the stand-alone XG-PON. Section 3.4 critically analyses the literature for avenues of improvement in terms of a QoS metric conversion scheme, DBA design for the converged network architecture and the suitability of DBAs when serving TCP-based application traffic in the LTE upstream. Section 3.5 concludes the chapter.

3.1 Overview

Standardisation of Gbps-range PON more than a decade ago, in the form of EPON and GPON, provided a strong justification for using PON as a promising transport medium in the access region of the service provider infrastructure. Recently, in 10G-EPON and XG-PON, both EPON and GPON were respectively upgraded for longer passive reaches and increased capacities, among many other advantages, thereby further promoting the concept of FTTx.

On the other hand, the demand for mobile application bandwidth, both in the downstream and upstream, has been increasing at a very high rate over the past decade. Recent forecast by CISCO[24] indicates that the global mobile data traffic will grow at a compound annual growth rate of 53% from 2015 to 2020. LTE, with its current features of Carrier Aggregation, Multi-Input-Multi-Output and 20MHz bandwidth and futuristic Cloud Radio Access Network option, promises to be a popular cellular and wireless technology over the next decade, to support the last-mile wireless users meeting their increased bandwidth demand.

Growing demand for the capacity in the last-mile also means frequent upgrades in the cellular (mobile) backhaul to meet the very high datarates and support for strong QoS framework. With long-term promises of easy capacity upgrades, longer passive reach, lower expenditures and the ability to generate market opportunities on the scale of billions of dollars, the concept of Fixed Mobile Convergence (FMC) has gained a significant recognition in the literature[14, 20].

Due to the many similarities outlined in section 2.1 of Chapter 2, both 10G-EPON and XG-PON are two strong candidates for the fixed part of the FMC. However, because of 1) supporting easy accommodation of several transport technologies such as ATM, IP and Ethernet under the same standard, 2) large-scale deployment and use of GPON and not EPON in the European region and 3) the advantage of having a detailed QoS framework, XG-PON becomes a more suitable candidate that 10G-EPON for the FMC of PON and LTE.

This chapter, in particular, presents the discussions regarding the FMC of XG-PON and LTE in the literature. Where applicable, explicit references are presented for the applicability of the discussions for the convergence of optical and wireless technologies, which is a wider scope with regard to the FMC of XG-PON and LTE.

3.2 Challenges for the FMC of XG-PON and LTE

Chapter 2 established that the FMC of (XG)PON and LTE is a highly feasible solution due to the ability of (XG)PON to support the growing datarate demand of LTE by having defined (in the standards) high physical datarates and long (up to 60km) passive reach in the access region. However, several authors[85, 70, 75, 81, 4] have identified the following technical challenges in terms of the practicality of implementing and validating the FMC of XG-PON and LTE.

1. (X)GPON defines only the Layer 1 and Layer 2 functions in the TCP/IP protocol stack, as PON is merely a transport network. Hence, any traffic arriving at the edge of (X)GPON, from the IP layer of the LTE (or last-mile) network, is added as a payload to the unique Layer 2 (X)GTC frame in (X)GPON, causing additional segmentations and concatenations of the frames at the interface of XG-PON and LTE. Hence, the effective bandwidth achieved by the clients of LTE (or any last-mile technology) is much less than the physical datarates defined by the (X)GPON standards. Additionally, the (XG)PON standard(s) require that a polling-based DBA mechanism is used to provide non-colliding bandwidth allocation in the upstream for ONUs with simultaneous requests for data transmission, which is inevitable when serving an LTE (or last-mile) network. As a result, the effective datarates in the upstream of (XG)PON is further reduced by the overhead related to the operation of the DBA mechanism in (XG)PON [31]. Hence,

FMC of XG-PON and LTE is presented with the challenge of designing a (simple and cost-effective) converged architecture, that can efficiently channel the effective datarate of (XG)PON towards the clients of LTE, while preserving the QoS differentiations of the applications in LTE and ensuring QoS guarantees across the XG-PON if possible.

2. Assuring bandwidth (datarate) to each eNB in LTE can be a challenging task. A DBA mechanism in PON has the capability to exploit the bandwidth sharing (or TDMA) nature of PON by provisioning different instantaneous bandwidth for different ONUs in the PON, at the small frequency of $125\mu s$. Since the upstream traffic in each eNB is bursty, an ONU that is directly connected to an eNB may experience a highly varying instantaneous (every 125microseconds) upstream load at each ONU, while remaining completely uncorrelated in terms of traffic loading between different ONU-eNB pairs. Hence, it will be a challenging task to design an effective DBA mechanism in (X)GPON and LTE, that can both facilitate significant network sharing options between the two technologies and maintain a high traffic utilisation for all the eNBs in LTE.

3. The issue of added burstiness in traditional mobile backhauls caused by the over-provisioning of bandwidth in the backhaul, is also applicable in the context of XG-PON-based LTE backhauls. The solution for this issue relies in designing an effective DBA mechanism, which can employ traffic monitoring capabilities and fine-grained optimisations to counter the burstiness. This is a crucial aspect of designing a suitable DBA for the FMC of (XG)PON and LTE since 1) a DBA in (XG)PON always causes the upstream traffic in PON to be bursty[68] and 2) burstiness in mobile (including LTE) traffic requires added over-provisioning at the backhaul[99]. Hence, the DBA in (X)GPON is challenged for managing the over-provisioning of the upstream capacity in backhaul for the bursty LTE applications, while ensuring differentiated QoS in the upstream.

These challenges prompt for 1) a simple and realistic integrated network architecture and 2) a suitable DBA framework for the FMC of (X)GPON and LTE. The following two subsections present the related work in the literature in detail, regarding the integrated network architectures and the DBA framework for the FMC of (X)GPON and LTE.

3.2.1   Types of Network architectures for the FMC of (XG)PON and LTE

The following two general categories of network architectures are presented in the literature for the convergence of optical and wireless technologies[85, 14, 31, 98]:

- **Radio-over-Fibre (RoF) architecture**, which refers to the Physical Layer integration of PON and Wireless technologies. In RoF, the Radio Frequencies are

carried from a wireless client all the way up to the core network of the wireless technology via the optical network

- **Radio-and-Fibre(R&F) architecture**, which refers to the convergence of the wireless and optical technologies specifically at the MAC layers, while preserving the unique features of both technologies (eg: scheduling).

For the FMC of XG-PON and LTE, however, the R&F architecture is a more suitable candidate due to the many advantages it offers over RoF, such as preserving decentralised MAC layer functionalities of both technologies, the relatively low RTT between the LTE users and the LTE core networks when XG-PON is placed at the backhaul of LTE and the failure-redundancy R&F offers against a single and crucial point of failure at the central scheduler in RoF [85]. R&F architectures are further classified into *independent* and *hybrid* architectures [85]. *Independent* architecture allows for the development, implementation and operations of the MAC layers of optical and wireless technologies to be independent of each other; common integration protocols can be used to connect the optical and wireless technologies. This architecture also requires minimal changes for the functionalities (eg: QoS mapping) related to the integration. In the *hybrid* architecture, both the wireless and optical domains are connected via a single MAC controller placed between the ONU from the PON and Base Station (BS) from the wireless domain. Generally, the ONU and the BS are bundled into a single device along with the MAC controller to enable this architecture. Hybrid architecture may take several flavours depending on the physical placement of the MAC controller, such as at the same box as the ONU and the BS (generic hybrid architecture), at a new node between the ONU and the BS (combined architecture) or at the OLT, away from the integrated box of ONU-BS (unified architecture) [85].

There are numerous examples of network architectures suggested for the convergence of (E)PON and WiMAX/LTE due to the simplicity that these standards offer. However, for the (X)GPON and LTE integration, only a limited number of R&F network architectures are proposed in the literature with even fewer of them being implemented for validation and evaluations of DBAs. The most notable ones are the unified architecture proposed by Hwang et al.[39] for the FMC of GPON and LTE and the independent architecture by Alvarez et al. [6] and Stynes et al.[93] for the FMC of XG-PON and LTE.

Discussion: As this thesis respects the standard-compliant nature of several components (simulation module, TCP analysis, DBA design, QoS metrics, etc.) in each of the (XG-PON and LTE) technologies in FMC, the *independent* network architecture used by Alvarez et al. and Stynes et al. is implemented in this chapter in the ns-3 simulation platform. Figure 3.1 shows an example *independent* network architecture for the FMC of XG-PON and LTE, as per the proposal in the literature.
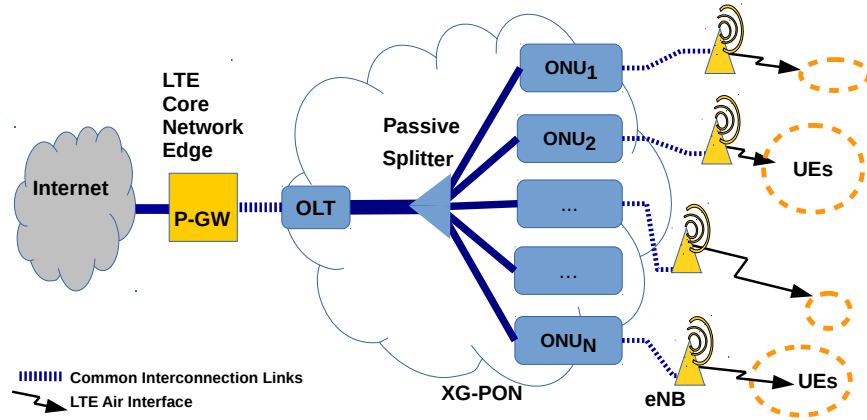
Figure 3.1: Example network architecture of an XG-PON backhaul in LTE
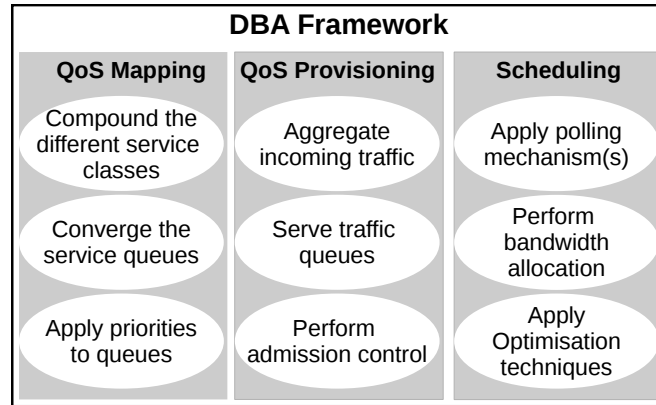
Figure 3.2: DBA Framework [85]

3.2.2 A DBA Framework for the FMC of (X)GPON and LTE

In their recent survey[85], Sarigiannidis et al. present a detailed study of the components involved in establishing a DBA framework for the integration of optical and wireless technologies. These components, grouped into 3 blocks, namely (a) the QoS Mapping block, (b) the QoS provisioning block and (c) the Scheduling block, as in Figure 3.2, are also directly applicable to the FMC of XG-PON and LTE.

While all three blocks are applicable in the context of XG-PON backhaul in LTE, both the *QoS Provisioning* and *Scheduling* blocks are also applicable in the context of the stand-alone XG-PON network, where the XG-PON serves fixed-broadband or Large Area Network (LAN) users directly connected to the ONU. When moving from the context of stand-alone XG-PON to the FMC of XG-PON and LTE, the *QoS Provisioning* and *Scheduling* blocks will only require minor changes to their sub-components (given within the circles in the figure) with the *QoS Mapping* block added fresh for the context of FMC.

Therefore, this chapter first presents the related work in the literature (in section 3.3) leading to the DBA design for the stand-alone XG-PON, before outlining the related work for the FMC of XG-PON and LTE, in section 3.4

3.3   DBA design for the stand-alone XG-PON

This section presents the related work in the literature, leading to: 1) the development of a standard-compliant XG-PON module, 2) the TCP traffic analysis in the XG-PON network and 3) the design, implementation and evaluation of a standard-compliant DBA mechanism for the XG-PON, as presented later in the Chapters 4, 5, and 6 of this thesis respectively.

3.3.1   Standard-Compliant simulation model for XG-PON

In order to expose challenges in, and provide solutions for the real-world deployments of the FMC of XG-PON and LTE, by means of performance evaluation in a large-scale converged network, it is necessary to have detailed models for the XG-PON and LTE standards. However, it is too expensive and time-consuming to set-up a large-scale (in terms of the number of ONUs or eNBs respectively) and fine-grained (in implementing all the details of the standards) XG-PON or LTE test-bed due to the numerous physical and software components needed to build the network environment. However, it is possible to have the above realistic challenges exposed and addressed in a simulation environment if such fine-grained (or standard-compliant) and highly-scalable simulation models of XG-PON and LTE are available freely or at a reasonable cost for the research community.

Since LTE was standardised almost a decade ago, there are several standard-compliant simulation models available for LTE. Due to the recent standardisation and the limited industrial adaptation of XG-PON at the time of writing this thesis, however, there is a scarcity of standard-compliant and highly-scalable simulation modules for XG-PON.

The literature has several simulation models of PON that have the potential to be the base model for developing a standard-compliant XG-PON model from scratch. Song et al. developed their own PON simulator to study the impact on the DBA, when having longer physical reaches than the EPON and GPON networks[89] in their model. However, this simulator lacks the IP stack, which is needed to study research topics related to the convergence of optical and wireless technologies. EPON, GPON and XG-PON [22, 78, 29, 8, 76] have all been implemented in OPNET [94]. However, since these models have highly simplified MAC layer for each of the PON standards, the challenges identified using these modules may not hold true in realistic deployments of the XG-PON network; as these OPNET models of PON were implemented at datarates lower than 1 Gb/s, they will also restrict the scalability of the FMC networks, which, in a real-world scenario, is capable of supporting several hundreds of eNBs, due to the high datarates of XG-PON; finally, as OPNET is not an open-source simulator, there is limited public access to the models, and the core of OPNET cannot be easily accessed and

modified to simulate a 10Gb/s XG-PON network with a reasonable simulation speed. A simple EPON module has been developed for OMNeT++ [17] with the code available in the public domain for further extensions. However, since there are several differences between EPON and (X)G-PON, this EPON module may either require extensive work for redesign or may not be helpful at all for implementing a fine-grained and highly-scalable XG-PON module with reasonable simulation speed in OMNeT++.

Discussion: The above reasons indicate that it is a difficult task to use the existing PON modules to build a standard-compliant simulation module of XG-PON, with specific attention to the MAC layer specifications, scalability, simulation speed and support (eg: Ethernet, IP, etc.) for integration with other wireless technologies. Hence, this thesis, in Chapter 4 designs, implements and validates a standard-compliant XG-PON simulation module for the state-of-the-art and open-source ns-3 simulator, so that the module can be used for the performance evaluations required to design a suitable DBA for the real-world deployments of the converged XG-PON and LTE networks, at a large-scale.

3.3.2 Traffic Analysis in XG-PON

TCP[80] is a popular Transport Layer protocol that is used in several applications, such as FTP, HTTP and SSH in the Internet [91]. TCP, unlike UDP, is also a reliable but complex Transport Layer protocol which first establishes an E2E connection before initiating application data transfer and then employs a complex congestion control methodology to leverage its sending rate across the underlying network architecture. Due to the complex nature of TCP, and its reliance on the latency and datarate of the underlying network architecture for performance, a network architecture with large bandwidth-delay product (or BDP[1]) may cause several challenges for the TCP flows with regard to the efficient utilisation of the network capacity.

Due to its high downstream and upstream capacities and the added upstream delays (on top of the propagation delays) caused by the polling-based DBA mechanism in the upstream, all the standards of PON qualify as a large BDP network, both in the downstream and upstream directions. The literature presents ample examples for the interactions between the large-BDP nature of EPON and GPON and the efficient utilisation of TCP across such networks. Chang et al.[23], investigated the performance of TCP traffic over EPON with focus on the interactions between the MAC layer of EPON and TCP; hence very little attention was given to the impact of different congestion control algorithms in TCP and realistic latency values in the network architectures, thereby presenting insufficient analysis into the utilisation nature of TCP flows across the large-BDP EPON network. The interaction between TCP traffic and the DBA employed in MAC layer of 10GE-PON was studied by Nishiyama et al.[72]; yet the scope

---

[1]BDP between two nodes in a network is calculated by the product of the network capacity between the nodes and the delay experienced by a packet in being transmitted from one node to the other

of the analysis was restricted to single congestion control based tests, with more focus on using multiple ONUs and smaller-than-BDP buffer values used for the ONUs. Orozco et al. analysed the effect of ONU buffers and TDMA Frame duration, for GPON, using single ONU and generic TCP [74]; however, based on their preliminary studies, the authors admitted that further analysis, which uses multiple flows of high-speed TCP congestion control algorithms across the GPON MAC layer, is required to accurately understand the possible interactions between TCP flows and GPON network.

Discussion: The above discussion indicate that so far, there has been no detailed study in the literature with regard to the performance of TCP flows across the XG-PON network, let alone the interactions between different congestion control algorithms of TCP and the large-BDP nature of XG-PON under complex scenarios. Therefore, in Chapter 5, this thesis presents the performance evaluation of generic TCP-based applications in a stand-alone XG-PON network, in order to identify the potential challenges faced by three TCP congestion control algorithms (NewReno, CUBIC and H-TCP), both in the downstream and upstream directions of the large-BDP XG-PON network.

3.3.3   Designing a DBA for the context of a stand-alone XG-PON

DBA design for the stand-alone (X)GPON network has been a popular field of research since the introduction of the GPON standard. Yet, only a few standard-compliant DBAs have been proposed with sufficient evaluations for the stand-alone (X)GPON, due to the challenges in adhering to the detailed QoS definitions presented in the (X)GPON standards and the complexity associated with the implementation of the DBA.

Simple[21] and two-state DMB[5], GIANT[59] and the DBA by Kanonakis and Tomkos based on the Offset-Based Scheduling with Flexible Intervals concept [56] are a few notable standard-compliant DBAs designed for GPON. GIANT, which was originally implemented in a GPON test-bed, was later optimised further to result in IA(CG) DBA[36] for GPON and finally resulted in EBU[35] and SFDBA[34] for XG-PON. DAMA[86] is an optimisation-based DBA, recently proposed for XG-PON and evaluated in a simplified (non-standard-compliant) model of XG-PON.

Though the DBAs designed for GPON and XG-PON can be algorithmically and conceptually similar, the XG-PON may require a DBA designed for the GPON to be redesigned completely and implemented differently due to the differences in the influential components in the DBA such as the instantaneous datarate achievable per-ONU, lengths and number of ONU queues required, frequency of polling and bandwidth allocation for the ONUs and the difference in the guard bands and the overheads.

There are numerous examples of DBAs designed for (10G)EPON, starting from the popular IPACT[58]. However, due to the differences in the MAC protocols and DBA

framework in EPON and GPON, re-implementing an EPON-compatible DBA in GPON, with all its properties and performance thresholds, is also a very complex task.

Hence, only the DBAs designed for the XG-PON are presented in detail in this section. DBAs for GPON are referenced only for the case of GIANT and IACG as they form the basis for Chapter 6 of this thesis.

3.3.3.1 GIANT and IACG

The GigaPON Access Network (GIANT[59]) DBA algorithm was both the first standard-compliant GPON mechanism and the first physically-implemented DBA algorithm for (X)GPON. Being simple in its design and implementation, GIANT proposed QoS differentiation between the four T-CONT types (T1 - T4) of GPON to provide prioritised bandwidth allocation for fixed, assured, non-assured and best-effort bandwidth types. GIANT used two main parameters, namely the Allocation Bytes (AB) and the Service Interval (SI), which defined the maximum amount and frequency of $GrantSize$ allocation, respectively, for $AllocIDs$ of different T-CONT types, thereby ensuring prioritisation between the T-CONT types.

Since the primary focus of GIANT was to have a physical implementation of the DBA, the authors focused more on managing the hardware resources than to the actual design of the DBA, which as acknowledged by the authors in its hardware and software evaluations, was heavily dependent on its statically-configured parameters. Specifically the SI parameter in GIANT presented restrictions for the bandwidth provision by allocating $GrantSize$ to an $AllocID$ at a minimum interval equivalent to *SI* multiplied by *the duration of an upstream frame*, regardless of the queue occupancy status of the $AllocIDs$ in the ONUs, thereby causing inefficient utilisation of upstream capacity in GPON and causing large queuing-delay values for all the $AllocIDs$. The combined choice of AB and SI also resulted in ambiguous effective priorities among the T-CONTs (T2, T3 and T4) when the upstream load was below the capacity of GPON. Since GIANT was evaluated using an OPNET model implemented by the authors for its performance against IP-based data traffic from fixed-broadband users, GIANT, to date, also remains invalidated for suitability in the context of FMC of GPON and LTE.

Improving the inefficient bandwidth utilisation in GIANT, Immediate Allocation with Colorless Grant (IACG)[36] DBA was proposed by Han et al. for GPON. IACG ensured the inefficient utilisation of the upstream capacity in GPON created by the use of SI in GIANT was mitigated to a certain degree by introducing additional parameters which assigned bandwidth to an $AllocID$, even when the down counter of the $AllocID$ has not expired. The colorless grant concept introduced in IACG also provided prediction-based bandwidth allocation to packets arriving at a T-CONT queue after the request was sent by the ONU request bandwidth allocation in a certain upstream frame.

Since IACG also followed the same design methodologies of GIANT and focused on real hardware implementation of the DBA as for GIANT, the DBA suffered the same drawbacks as in GIANT, in terms of inefficient utilisation of upstream capacity, prioritisation ambiguity and compromising performance over hardware limitations.

3.3.3.2   EBU and SFDBA

Efficient Bandwidth Utilisation (EBU [35]) and Simple and Feasible DBA (SFDBA [34]) are two improved flavours of IACG, proposed for and evaluated in XG-PON, by the same authors of IACG.

EBU significantly improved IACG by introducing further parameters to efficiently utilise the upstream capacity at the cost of high design complexity in terms of additional parameter calculations. SFDBA simplified parameters in IACG to avoid the inefficiency in upstream bandwidth utilisation, however, at the cost of assuming the existence of traffic policing (eg: leaky bucket) rules in GPON. EBU and SFDBA also introduced the inter-ONU-fairness policy to provide fairness among T-CONTs of each T-CONT type in terms of upstream bandwidth utilisation.

Since both EBU and SFDBA improved GIANT by keeping the fundamental QoS policies of GIANT intact, both these DBAs were also restricted heavily by the static nature of configuration parameters (AB and SI) and bandwidth allocation methodologies, though showing significant improvement over IACG in terms of bandwidth utilisation for different T-CONT types. Between the two DBAs, SFDBA provided marginal improvements over EBU with the advantage of simplicity in implementation and the cost of lacking traffic policing capabilities which were implicitly available in IACG and EBU.

3.3.3.3   DAMA

The Data Mining Forecasting (DAMA) DBA is proposed by Sarigiannidis et al.[86], for the purpose of assuring latency and jitter performances for voice and video applications, by means of optimised prediction methodologies. The prediction is geared towards providing the additional bandwidth required for the packets arriving at an ONU queue after a queue occupancy report has been sent by the ONU and before the next upstream frame is transmitted by the same ONU.

Evaluations were performed using a MATLAB implementation of abstracted XG-PON for a range of ONUs situated at a uniformly distributed distance of [20,60] km from the ONU. The results demonstrate the improvements in DAMA, for latency and jitter in VoIP and video traces, when compared with an equivalent non-prediction-based DBA methodology.

However, the purpose and the design of DAMA is an isolated scenario of addressing the impact of differential propagation delays between different ONUs when serving multimedia traffic. Hence, DAMA assumes several aspects of DBA provisioning in XG-PON, in order to focus on the impact of different propagation delays on the application performance. As a result, DAMA only be a good addition to a general-purpose DBA for providing QoS in a stand-alone XG-PON; that is, once the general purpose DBA has assured prioritised and differentiated QoS for and efficient bandwidth utilisation of different classes of traffic in the stand-alone XG-PON for general deployment conditions, DAMA can then be used to assure the prediction-based latency guarantees for the multimedia traffic under isolated conditions.

Discussion: DBAs designed to provide differentiated and prioritised QoS assurances among several traffic types for the stand-alone XG-PON are both limited in numbers and are far from maturity, in terms of validation and extensive evaluation capability. Both EBU and SFDBA are very similar DBAs in terms of algorithmic and QoS principles, with the difference merely in the implementation complexity: EBU is a highly complex DBA and SFDBA is a simple DBA. The resulting performances are therefore only marginally better for SFDBA, though SFDBA succeeds EBU in terms of timeline of introduction in the literature, leaving EBU as the only contender at the time a QoS-aware and standard-compliant DBA is developed by the author for the standard-alone XG-PON. Since EBU (and SFDBA) is based on the algorithmic and QoS policies of GIANT DBA, in Chapter 6, this thesis first designs and implements (in ns-3) another GIANT-improved DBA, namely the XGIANT, before presenting comparative performance analysis between XGIANT and EBU under similar traffic loading conditions in the XG-PON upstream.

3.4   DBA design for the FMC of XG-PON and LTE

This section presents the related work with regard to DBA design for the FMC of XG-PON and LTE. As explained in section 3.2.2, once a DBA is designed for the stand-alone XG-PON featuring the sub-components of *QoS Provisioning* and *Scheduling* blocks (see Figure 3.2), the same sub-components can easily be extended to design a DBA for the FMC of XG-PON and LTE; the extension should however understand the strong differences in traffic patterns between LTE UEs and fixed-broadband users. The only fresh component required for the FMC, compared to the stand-alone XG-PON context, is the *QoS Mapping* block which focuses on statically/dynamically mapping the different classes of upstream traffic in LTE to that in XG-PON when XG-PON is the backhaul for LTE as in the *independent* R&F architecture. With all three blocks implemented in the *independent* network architecture, it will be up to the combined effect of the extended sub-components of the *QoS Provision* and *Scheduling* blocks to ensure, if possible, the QoS requirements of the upstream LTE traffic in the XG-PON backhaul.

This section first presents the QoS mapping policies proposed in the literature for the context of XG-PON-based LTE backhaul, before presenting the related work regarding the DBA design in the same context.

3.4.1   QoS mapping policies suitable for the FMC of XG-PON and LTE

Both (XG)PON and LTE were designed to service clients at different regions (transport and wireless last-mile regions respectively) of the service provider's network infrastructure. Hence, the FMC of XG-PON and LTE, with each of the technologies having a different QoS framework as explained in sections 2.2.3 and 2.3.1 of Chapter 2 respectively, requires a suitable and effective QoS metric conversion scheme[61], in addition to a common scheduling mechanism[81] to serve the user traffic across the integrated network architecture.

The literature presents very few proposals for mapping schemes between the QoS metrics of (X)GPON and LTE (T-CONTs and QCI respectively), among which only a handful are implemented for the performance analysis of multiple classes of traffic in LTE, across the XG-PON backhaul. In [61], Lim et al. presented two types of conversion schemes, namely the simple *1:1 mapping scheme* and the more sophisticated *group mapping scheme*, while presenting a comparative performance (packet delay) evaluation of the two schemes with regard to the non-mapping scheme (all LTE classes assigned to single T-CONT type) to validate the superiority of the mapping schemes. Based on Lim et al.'s proposal, Hwang et al. [39] implemented a group mapping scheme between 9 QCI values in LTE and 5 T-CONT types in GPON for the *unified* network architecture of GPON and LTE, with evaluations presented for T2, T3 and T4 T-CONT types with regard to packet delay and throughput.

Though Astudillo & Da Fonseca [14] proposed a standard-complying scheme between EPON and LTE, by first converting the QCI values to the Differentiated Services Code Point (DSCP) values and then mapping all the DSCP categories to a single ONU queue, such a scheme: 1) neglects the preservation of LTE classes in XG-PON, 2) provides no insight into the relative performance in the integrated network architecture if multiple classes are maintained in ONU and 3) leaves a question as to the manner the DSCP values are mapped onto the T-CONT types.

Hence, a standard-compliant QoS mapping scheme, for the *independent* architecture-based integration of XG-PON and LTE, remains only the in the scope of proposals, let alone the implementation (after configuring multiple queues in each technology and classifying traffic in the respective queues) of and performance evaluations based on such schemes.

3.4.2   DBAs for the XG-PON backhaul in LTE

There are several proposals in the literature for the integration of EPON and LTE technologies, in terms of both network architecture and the DBA. Astudillo et al.[14] discuss a standard-compliant QoS provisioning scheme for the integration of LTE and EPON networks. They use an architecture where ONU from EPON and eNodeB from LTE are combined at the hardware and software level, inside a single box. As a result, eNodeB becomes a client of the EPON for scheduling and resource allocation. They also propose a QoS mapping scheme, for integrated EPON and LTE. A single FIFO queue ONU accepts packets of all QCI values from eNB; however, eNodeB makes sure - based on the ratio of Grant and Report message, C(k) - the bearers with highest ratio of the head-of-line delay[2] of packets to that of the PDB of the packets' bearer are served first, using a congestion feedback received from EPON. Astudillo et al. use two schedulers, Z-Based QoS Scheduler (ZBQoS) and Hybrid ZBQoS (HZBQoS) for bandwidth allocation in the converged network architecture. For a slightly different EPON and LTE integration, which combines the residential and mobile traffic of different classes at a single ONU, Lim, et al.[61] proposed a multi-queue based QoS mapping scheme between backhaul and mobile traffic. They laid out 1) one-to-one and 2) group mapping between the traffic priorities in LTE and OFDMA-PON, in order to evaluate the packet delay of 3 (single/grouped) bearers in each ONU-eNodeB.

Due to the complexity of both the QoS framework and the standard specifications of (X)GPON as well as the integration of (X)GPON and LTE network, the literature has only two proposals[39, 6] for the design and evaluation of DBAs for the FMC of (X)GPON and LTE network.

In [39], Hwang et al. presented the implementation of GPON-LTE converged network architecture (GLCNA) which is used to evaluate the proposed Synchronous Interleaved DBA (SIDBA) in GLCNA. GLCNA assumes a dependent integration of GPON and LTE, in that the traffic transmission requests from the UE in LTE is carried all the way to the OLT, via the intermediate ONU-eNB node, which aggregates the requests before sending the aggregated request to the OLT. The aggregated grant given by the OLT, first reaches the ONU-eNBs, where the individual grants for the UEs are further managed by the LTE air interface scheduler. At the intermediate ONU-eNB node, GLCNA features two Grant Aggregators (one each for GPON and LTE), a common Grant Generator which coordinates the upstream schedulers in GPON and LTE and a QoS mapping mechanism which classifies both the GPON traffic and LTE traffic into four ONU queues to provide differentiated QoS treatment across GPON. SIDBA provides grant to the ONUs at different frequencies (1ms and 2ms), assuming that the grant from an ONU-eNB is a combined grant of the ONU and the UEs associated with the eNBs. That

---

[2]head-of-line delay for a series of packets in a queue refers to the (queuing) delay of the first packet waiting to be transmitted from the queue

is the requests of the UEs arriving at the ONU-eNB queue every 5ms or 10ms, waits in the ONU queues, along with the ONU traffic before requesting grant from the OLT every 1ms or 2ms, thereby causing asynchronous coordination between the LTE framing frequency and the OLT polling frequency. Statistically modelled per-ONU-aggregated and per-eNB aggregated traffic patterns are used for GPON and LTE respectively, to evaluate the throughput and queueing-delays for three types of traffic in the GPON upstream. The authors also provide evaluations of SIDBA against IPACT DBA (designed for the EPON originally) using an OPNET implementation of GPON.

Overall, the combined proposal of GLCNA and SIDBA is an attractive solution to the integrated GPON, LTE network, where the abilities of SIDBA is evaluated for providing throughput and queuing-delay assurances to the combined GPON and LTE traffic across the GPON backhaul. However, SIDBA is evaluated under highly abstracted upstream scenario in the GLCNA: the OPNET model is a simple network model which merely simulates the propagation delays and guard bands in the Physical Layer of PON with the DBA implemented as a function rather than part of the GPON operation; traffic in LTE is generated using statistical values from CISCO representing Long-Range Dependent self-similar traffic, which is generally used to represent LAN traffic. SIDBA also assumes fixed values for the polling frequency of GPON (1ms, 2ms) and frame lengths of LTE (5ms and 10ms), both of which are unrealistic values due to the scheduling frequency of $125\mu s$ and 1ms in GPOn and LTE respectively; hence it may be harder to attribute the final results, both in terms of behavioural pattern and the absolute values of the evaluation metrics, to realistic scenarios of upstream load in GPON as well as in LTE.

In [6], Alvarez et al. presents the latest proposal found in the literature with regard to the performance evaluations of the DBAs designed for the converge network of (X)G-PON and LTE. The authors use the same standard-compliant XG-PON module presented later in this thesis (Chapter 6) and a modified GIANT, in the name of Group GIANT (g-GIANT) to evaluate the concept of group assurances in the XG-PON backhaul to dynamically support the co-existence of ONUs serving LAN users and eNBs (ONU-eNBs) in the same XG-PON network. A major contrast between the earlier proposal and this is that: Hwang et al. in [39] assume a dependent (*uniform*) network architecture in GLCNA where an ONU always serves both the LAN and eNB traffic; Alvarez et al., however, propose an independent network architecture for the convergence where the XG-PON and LTE retain their independence in scheduling while the ONUs serve either LAN traffic or eNB traffic and not both. The evaluations present the performance of g-GIANT when serving only the assured bandwidth type traffic generated by a Poisson Distributed traffic model, to prove the success of g-GIANT in utilising the unutilised assured bandwidth, to grouped ONU-eNBs. However Alvarez et al. assumes statistically generated and LAN-based traffic distribution to represent both the LAN and LTE traffic, while providing evaluations for only 16 ONUs which, on average divides the per-ONU datarate to 140Mbps, which is a very high value for the capacity of an eNB. Though

conceptually explained, the authors avoided evaluation-based results with regard to other traffic types (fixed and best-effort) in XG-PON, thereby failing to justify the relative performance of g-GIANT during the simultaneous presence of more than one traffic type in LTE and different upstream traffic loading across the XG-PON backhaul.

Discussion: Overall, though the proposals by Hwang et al., and Alvarez et al. have provided good motivation for the DBA design suitable for the FMC of (X)GPON and LTE, neither papers address the problem of designing suitable DBAs for realistic application models in the XG-PON backhaul in LTE. While Hwang et al. used fixed and per-ONU/eNB-aggregated statistical models to evaluate the DBAs in a *unified* network architecture, Alvarez et al. proposed policies only for the isolated scenario of a single traffic type being shared among a group of eNBs in providing datarate assurances. The evaluation methodologies used by both proposals also provided little knowledge regarding the performance of a DBA in the (X)GPON backhaul when serving different classes of realistically-generated upstream application traffic in LTE, while using the *independently* network architecture between XG-PON and LTE.

Addressing these challenges, this thesis, in Chapter 7, first evaluates the existing DBAs (XGIANT and EBU) of stand-alone XG-PON, in the converged and *independent* network architecture after implementing a static one-to-one QoS metric conversion scheme between XG-PON and LTE. Hence, Chapter 7 identifies several avenues of improvement with regard to the mean queuing-delay behaviour for three different classes of realistic traffic models in LTE upstream before presenting the design, implementation and evaluation details of two fine-tuned DBAs (XGIANT-D and XGIANT-P) for ensuring prioritised and fair QoS treatment (for throughput and queueing-delay) across the XG-PON backhaul for the different classes of upstream traffic in LTE.

3.4.3   Performance of TCP in the FMC of XG-PON and LTE

All the past work on the DBA framework for the FMC, including the design of the DBAs in Chapter 7 of this thesis are evaluated for performance using only UDP-based application traffic. Since UDP is a non-responsive and simpler (than TCP) protocol , it is a reasonable choice for the Transport layer protocol, when validating the performances of the DBAs against the un-altered traffic patterns of the realistic applications.

However, since TCP is the most popular Transport Layer protocol for IP-based traffic in the Internet, it is useful to evaluate the performance of the same DBAs when serving TCP-based application traffic, to identify and address the challenges that may occur in more realistic deployments (when compared to the simulation environment of UDP-based traffic) of LTE network with XG-PON backhaul.

Potential challenges with regard to the performance of TCP-based applications in the integrated network of PON and LTE have so far been only hinted by studies in the

context of isolated LTE or PON. In LTE, Huang et al. [38] analysed the impact of queuing-delay on the RTT measurements, which eventually cause different scales of throughput degradation in LTE air interface for short- and long- lived TCP flows. Liu et al. [62] and Jian et al. Jiang et al. [55] studied the impact of using large buffers in LTE base-stations/core networks thereby providing insight into the resulting RTT values at the Transport Layer; the impact of E2E BDP, which is a product of the theoretical datarate and E2E one-way delay is studied by authors in [18] with regard to the effective throughput in LTE. On the other hand, we presented a detailed analysis of several challenges posed by TCP-based applications in a stand-alone XG-PON network using a generic Round Robin DBA [13].

However, the integration of PON and LTE presents several unexplored challenges on its own. Two notable contributions for the combination of TCP, PON and LTE are 1) by Stynes et.al [93] in using opportunistic caching at the eNB for downstream broadcast traffic and 2) by Aurzada et al. [15] in identifying routing issues in the integrated network for TCP-based upstream application traffic. However, so far, no evaluation study is presented for the co-existence of TCP-based applications across the integrated networks of PON and LTE, especially with regard to the challenges faced due to the decoupled, non-synchronised and parallel scheduling mechanisms in PON and LTE, the added queuing-delay by the ONU queues which accumulate the bursty LTE traffic with unpredictable arrival patterns (when aggregating numerous amounts of realistically-generated traffic flows) in the backhaul and the policies used in a standard-compliant QoS-aware DBA mechanism operating at a frequency of $125\mu s$ in the PON.

Discussion: This thesis addresses the above concerns, in Chapter 8, by providing a comprehensive analysis on the performance of realistically-generated TCP-based upstream LTE applications across the XG-PON backhaul, which is individually served by each of the four DBAs, namely XGIANT, EBU, XGIANT-D and XGIANT-P.

3.5   Summary of the Discussions

This chapter reviewed the published literature to identify the open research avenues regarding the FMC of XG-PON and LTE.

First, this chapter presented a brief overview regarding the FMC of PON and LTE, and outlined the challenges from the literature, specifically regarding the FMC of XG-PON and LTE. The *independent* architecture (used by Alvarez et al. and Stynes et al.) of the R&F network architectures in the literature is chosen for implementation in this thesis, so that a common QoS framework for the FMC of XG-PON and LTE can be developed.

The need to have a standard-compliant simulation module for XG-PON is then identified by analysing the existing EPON and (X)GPON simulation modules used by several

authors' published work, so the contributions in Chapter 4 are justified. This chapter also identified two timely research avenues for the stand-alone XG-PON network: a detailed analysis of TCP-based application traffic across the large-BDP network, XG-PON, and developing simple and efficient DBA mechanisms for a stand-alone XG-PON network, which are addressed by the contributions in Chapters 5 and 6 respectively.

Finally, a comprehensive analysis of the DBA mechanisms developed for the FMC of XG-PON and LTE is presented in this chapter, indicating open research avenues for a suitable QoS framework development using standard-compliant methodologies for the converged network architecture of XG-PON and LTE. The chapter stressed, by critically analysing the literature, that developing suitable DBA mechanisms in the XG-PON backhaul for LTE is a significant gap in the literature; the contributions in Chapter 7 addresses this gap by developing two standard-compliant DBA mechanisms for the dedicated XG-PON backhaul in LTE. An extended review of the literature also indicated that a detailed performance analysis of the TCP-based applications is required in the converged network architecture of XG-PON and LTE so that the developed DBAs for the dedicated XG-PON LTE backhaul in LTE is validated, as in Chapter 8, for suitability in realistic network deployment scenarios.

Chapter 4

A standard-compliant simulation module for XG-PON in ns-3

This chapter presents an XG-PON simulation module designed for and implemented in the state-of-the-art ns-3 [2] network simulator. The simulation module, being the first XG-PON-standard-compliant module designed for an open-source simulation platform, is based on a series of G.987 recommendations from the FSAN group of ITU-T. The resulting simulation module of XG-PON can therefore be used by the research community in simulating a realistic and standard-compliant XG-PON network with sufficient provision for identifying issues in realistic XG-PON deployment scenarios. The G.987 recommendations by ITU-T mainly define the specifications of PMD and XGTC layers of XG-PON. To study the research topics related with the XGTC layer with reasonable simulation speed, the XG-PON simulation module presented in this chapter, significantly simplify the ODN and the operations of the Physical and PMD layers. XGTC layer is modelled in detail so that its aspects such as frame structure, resource allocation, QoS management, and DBA algorithms for the upstream traffic can be studied extensively.

The design and implementation of the XG-PON simulation module focus primarily on the extensibility and reconfigurability of the simulation module, to support the integration of the XG-PON network with several IP-based technologies. Since an XG-PON should be able to simulate a 10 Gbps (downstream in XG-PON1 standard) network with hundreds of ONUs, significant attention is also given to the overall performance (simulation speed and memory consumption) of the simulation module when designing and implementing the individual components. Specifically, the resulting XG-PON module demonstrates its ability to run at reasonable simulation speed, even when the XG-PON supports more than 1000 ONUs and is simulated at 9.6 Gbps in the downstream (validated by the experiments in section 4.4).

In this thesis, the XG-PON simulation module is used as a major component in the underlying simulation module in Chapters 5 to 8 to develop realistic, XG-PON-standard-

compliant and QoS-aware DBA mechanisms; these DBA mechanism are designed, by performing extensive evaluations using the XG-PON simulation module presented in this chapter, for the architectures of a stand-alone XG-PON and an XG-PON-based LTE backhaul to facilitate prioritised scheduling of multiple classes of upstream traffic in both architectures.

The XG-PON simulation module is developed under the ns-3 namespace, using the C++ programming language with 72 classes and approximately 22,000 lines of code. This code (under the GNU General Public License) and the detailed documentation can be downloaded through sourceforge [97]. The XG-PON simulation module is also a collaborative effort among 3 researchers, with the author's contributions accounting for about 40% of the details of the XG-PON simulation module presented in this chapter. Specifically, the author's contributions are in terms of designing and implementing the classes related with the scheduling and DBA mechanisms of XG-PON as well as evaluating the performance of the simulation module for functionality validation.

The chapter first introduces the principles and the key decisions involved in designing the XG-PON simulation module for ns-3 in sections 4.1 and 4.2 respectively; more definitive details of the design and implementation of the XG-PON module are furnished in section 4.3. The evaluations of the XG-PON simulation module with regard to the functionality, performance and robustness is presented in section 4.4, before concluding the chapter in section 4.5.

4.1 Design principles

This section presents the principles that influenced the design of the XG-PON-standard, G.987-compliant simulation module for the ns-3 network simulator.

4.1.1 Standard-compliance

The ultimate goal of this chapter, in designing an XG-PON simulation module, is to improve the performance bottlenecks surrounding the XGTC layer in XG-PON and the Network and Transport Layers of the technologies in the Internet. That is when a simulation model has a close resemblance to the definitions of the XG-PON standard (G.987x) by ITU-T and the operators adopt the recommendations of G.987 in their real-world deployments of an XG-PON network, not only will a user (of the XG-PON simulation module) be able to identify realistic problems easily but also provide solutions that can directly be applied in realistic deployments of XG-PON in the midst of many other technologies (eg. Fixed Broadband, WiFi, LTE, etc). Hence the ITU-T's recommendations in G.987 standard, especially those related to the XGTC and upper

layers of XG-PON are adopted in a highly-detailed fashion in the design of the XG-PON simulation module presented in this chapter.

4.1.2   Simplification of XG-PON: from standard to a simulation module

While standard-compliance is the primary objective of the XG-PON simulation module, it is also within the interest of this thesis that the simulation module developed for XG-PON standard can be evaluated for realistic scenarios in reasonable time frames. Since XG-PON standard has highly detailed specifications and requirements, it may take a very long time to simulate an XG-PON network, with all its layers, from the Physical to the network management. Hence, only the XGTC and upper layers are implemented in detail with the less significant features omitted or designed as abstract classes for future extensions: the physical layer can be simulated in a very simple way, assuming the power budget for the ODN to be satisfied through various techniques; implementation of the reach extenders and passive optical splitters/jointers can be skipped; the channel, that simulates the ODN, can be simplified in handling the downstream frames and upstream bursts; complex algorithms such as the FEC, can be simulated only for its effect (eg: overhead and packet corruption rate) instead of the entire algorithm being implemented. Since DBA in the upstream is the fundamental feature behind several chapters in this thesis, for developing QoS policies for the XG-PON standard, the classes for the DBA should be designed in detail to allow for easy implementation of various DBA algorithms.

4.1.3   Extensibility of the simulation module

When designing the XG-PON module for ns-3, the extensibility of the simulation module is given due consideration since many other research topics can also be studied using this module. When designing the class architecture of the XG-PON module, abstract classes are used appropriately and the interfaces, if simplified, are well-designed for extensive future implementations. That is, in this phase, only a very simple implementation is provided for the components that are less significant for us. For instance, when designing the class interface for the channel that simulates the ODN, a user of the simulation module may specify the tree structure of the fibre, reach extenders, and the splitters; provisions for specifying the type of the splitter and its physical distance to the ONU are also available when adding an ONU; the interface can also be extended to include the optical signal propagation behaviour and the possible packet corruption in ODN. However, in the current phase, the simple channel model only stores a list of ONUs and merely passes the downstream frame or upstream burst to all the ONUs or the OLT respectively, without any error[1].

---

[1]If needed, a likely low packet corruption rate, with the effects of FEC, may be simulated

4.1.4   Configurability

When an XG-PON network is simulated, the network set-up could comprise of thousands of nodes, including the OLT, few hundreds of ONUs, and several hundreds of data traffic generators/sinks. Thus, while exporting as many configurable parameters as possible, default parameters should also be provided for most of them. Other methods, such as the existing use of helpers in ns-3, should also be considered to ease the researcher's task of configuring the XG-PON network for simulation.

4.1.5   Reasonable simulation speed

Simulation speed is a crucial aspect of the simulation of an XG-PON network due to the very high datarates of 10Gbps in the downstream and 10Gbps/2.5Gbps in the upstream of the XG-PON. An XG-PON module that can simulate XG-PON with extreme accuracy by heavily compromising the simulation speed will be of no use to the research community. So the data structures and algorithms should be selected carefully for enhancing the simulation speed and saving the system memory. For instance, when XG-PON is fully loaded and the size of each packet is 1 KBytes, the simulator needs to process around one million packets per second. Since XG-PON could have hundreds of ONUs (1023 at most), the simulator must run the procedure one billion (1 million * 1023 ONUs) times per second to assert that an ONU is the destination of an XGEM frame, hence requiring very high efficiency at the implementation. If simulation speed is not a concern, a vector can be added at each ONU whose index is the XGEM Port-Id; when configuring the XGEM Ports for this ONU, this vector can be marked accordingly; later, this vector can be used to filter out the traffic for this ONU quickly. However, XGEM Port-Id is a 16-bit number and this vector can consume a lot of memory for a large number of ONUs. Due to the same reason, hash map in which XGEM Port-Id acts as the key, is not adopted in the implementation of this simulation module, which only has a simple relationship among the XGEM Port-Id, $AllocID$, ONU ID, and IP address of the computer the XGEM Port belongs to. This way, only a small amount of memory is consumed overall and $O(1)$ time complexity is achieved by mapping the IP address/XGEM Port-Id to the relevant data structure.

During the implementation, many useful features of the C++ language should also be exploited and some black-holes of CPU cycles be avoided:

- Parameters should be passed by reference whenever it is possible with *const reference* preferred in general; since the smart pointer provided by ns-3 is fundamentally a small object and function-calls in ns-3 use smart pointers lavishly, reference of the smart pointer is used instead of the smart pointer itself.

- Since C++ allows one class to override its *new* and *delete* operators, this feature should be exploited for data structures that are created and destroyed dynami-

4. A standard-compliant simulation module
   for XG-PON in ns-3
4.2 *Key design decisions*

cally and frequently. By overriding these two operators, a pool of pre-allocated memory is used for small and dynamic memory requirements, thereby avoiding frequent calls to *malloc/free* and saving CPU cycles as a result.

- When selecting the data structure for a sequence of objects, *vector* object in C++ should be considered due to its efficiency. However, when too many objects are added into one vector, reallocation may occur and the simulation can be slowed down significantly. Thus, enough memory should be reserved upfront if the largest vector size can be pre-determined. Otherwise, *dequeue* should be considered as the container.

- Although virtual function and inheritance are very attractive, they should only be used when absolutely necessary since a virtual function is much slower than a common function. Class downcast should also be avoided in the implementation since it is unsafe and consumes a lot of CPU cycles. For instance, for each function of XGPON (DBA, etc.), there should be two classes designed for OLT and ONU, respectively, and it is attractive to let them inherit from the same parent. However, the logic at OLT is totally different than that at ONU, the amount of reused code is limited, the interface of the parent becomes more complex, and simulation speed is slowed down. Thus, these classes are designed independently and the inheritance is not used.

4.2 Key design decisions

Based on the high-level design principles presented above, this section presents the key design designs which govern the design and implementation of the XG-PON simulation module in ns-3. Corresponding implementation details will be presented later in section 4.3.

4.2.1 Stand-alone simulation

Since XG-PON is a network with hundreds of ONUs, it is very attractive to use distributed simulation to expedite the overall simulation speed. However, although ns-3 supports distributed simulation through standard Message Passing Interface, this feature only works for point-to-point links and XG-PON is fundamentally a point-to-multipoint network. A large amount of work and time is therefore required, first to enable distributed simulation for XG-PON and then to study how to allocate ONUs to different computers. Thus, in the current phase, this XG-PON module works as a stand-alone simulator. It uses only one core even when one computer has multiple processors or cores. In the future, distributed simulation may be considered for implementation.

A Quality of Service framework for upstream traffic 50
in LTE across an XG-PON backhaul
Jerome A. Arokkiam

4.2.2   Packet-level simulation

Due to the high bandwidth of XG-PON (10Gbps) and the moderate clock-speeds of
the state-of-the-art processors (several GHz), it is not feasible to simulate the details
of data transfer in byte or bit level. But it's too complex to model both XG-PON and
TCP/IP protocol stack in a flow-level simulation with the added issue of not being able
to study the potential subtle interactions between TCP/IP and XG-PON. Since ns-3 is
fundamentally a packet-level simulator, this XG-PON module is simulated at packet-
level. Furthermore, when passing traffic between OLT and ONU, all XGEM frames in
the downstream frame or upstream burst should be handled together; as a result, the
number of simulation events can be reduced significantly. Due to the short XG-PON
frame size ($125\mu s$), the upper layer protocols and the presumed latencies at the XGTC
layer won't be affected if the order of XGEM frames is maintained in the downstream
frame or upstream burst. Based on this decision, many physical layer operations, such
as line coding and FEC, will not be implemented in this module. However, the band-
width overhead of FEC will be considered. Payload encryption/decryption will not be
implemented as well, though the logic used for key management is implemented. It
is also assumed that all the sub-modules in the XG-PON simulation module complete
their execution at the required point in time within every downstream frame or up-
stream burst, regardless of the complexity of the design and implementation of the
sub-modules. The total run time of a simulation using this XG-PON module will, how-
ever, depend on the scale and complexity of the configured XG-PON network as well as
the complexity of other required simulation modules from ns-3.

4.2.3   Focus on the operation mode of XG-PON

Since this chapter focuses on the performance issues of only one XG-PON network
(single OLT and multiple ONUs) in operation, many aspects of XG-PON can be sim-
plified. For instance, the activation procedure that uses PLOAM messages to add an
ONU to the operation XG-PON network need not be implemented. Instead, using a
helper class, all ONUs are statically placed in the XG-PON network before starting the
simulation. The ranging procedure that uses PLOAM messages to measure the one-way
propagation delay of each ONU, is simplified by setting the same delay values to both
downstream and upstream at the time of configuring (before running the simulation of)
the XG-PON network. In XG-PON, XGEM Port and T-CONT configuration is carried out
through OMCI (G.988). For simplifying the dynamic configuration of XGEM Port and
T-CONT by OMCI, all XGEM Ports and T-CONTs are also statically configured through
the helper class, before starting the simulation. Relevant stub classes will be designed
in future extensions, for a detailed implementation of PLOAM and OMCI channels.

4.2.4 Simple ODN and reliable data transfer

In XG-PON standard, the ODN is a highly complex structure comprising of numerous optical fibres, splitters/jointers, and reach extenders. Although these are important to network architecture and optical device research, they are irrelevant to the research topics addressed in this thesis. Thus, the ODN will be modelled as a simple channel with only the propagation delay and line rates simulated. The link power budget is assumed to have been ensured through various techniques (reach extenders, etc.) and the laser receiver is expected to work well. Thus, the optical signal propagation (wavelength-dependent) is not simulated; all downstream frames and upstream bursts are also assumed to be arriving at their recipients correctly. That is, transmission errors are not simulated in the simulation module; this is reasonable since FEC is normally applied to rectify transmission errors; as a result, Cyclic Redundancy Check and Header Error Correction are not executed in the simulation.

In the future, transmission errors may be simulated at the recipient, by dropping an entire downstream frame or upstream burst with a distance-dependent probability. That is, an occurrence of a transmission error, with no frame delineation, should be able to prevent a recipient from decoding an entire frame/burst when FEC fails to identify the frame/burst with sufficient accuracy.

4.2.5 Serialization avoidance and use of meta-data in data structures

Since this XG-PON module is designed for stand-alone simulation, (de)serialization is unnecessary and should be avoided[2] in handling the XGEM frames. At first glance, the $Packet$ class provided by ns-3 looks like an ideal candidate; XGEM frame header can easily be added into and extracted from the $Packet$, which also supports fragmentation and reassembly needed by the XGEM encapsulation. However, if the XGEM frame header is added into the $Packet$, the header is serialized and added into a byte array. When an XGEM frame is received, the recipient has to extract the XGEM frame header from the byte array using means such as creating a new data structure for the XGEM header, and carry out de-serialization. Considering that one XGEM frame in the downstream direction will be processed by hundreds of ONUs, the above operations may consume too many CPU cycles. To solve this issue, the $XgponXgemFrame$ data structure is added to this XG-PON module to represent the XGEM frame. The $XgponXgemFrame$ is designed to have a smart pointer of the $XgponXgemHeader$ for the XGEM header and another smart pointer of the $Packet$ data structure of ns-3 for the payload. This enables the recipient to extract the $XgponXgemHeader$ directly from the $XgponXgemFrame$.

---

[2]All data structures must provide one function to return its serialized size to accurately compose downstream frame and upstream burst.

Another observation is that some meta-data can be added into data structures for various purposes since they are exchanged between OLT and ONU as objects (instead of a byte array). For instance, all the broadcast XGEM frames in the downstream should be checked by all the ONUs, to decide whether to accept the relevant frames or to drop them. However, preliminary evaluations indicated that the traffic in one downstream belongs only to a few ONUs, due to the small size of the downstream frames and the bursty nature of bandwidth allocation. Thus, to speed up the simulation significantly, a bitmap is added to each downstream XGEM frame merely to indicate whether one ONU needs to check the particular frame, so that not every XGEM is checked by each ONU in detail for frame acceptance or drop.

4.2.6 Extensible DBA, scheduling, and queuing schemes

DBA engines, scheduling algorithms and the queue used by each XGEM Port at the sender side are very important for the performance of the whole network and the QoS experienced by user traffic. Thus corresponding classes were designed carefully to support future extensions. Creating abstract classes for these schemes would allow new algorithms to be implemented easily, by redefining only a few key functions. Further details of the DBA classes implemented in the XG-PON simulation module will be discussed in section 4.3.3.

The above key decisions enable the XG-PON simulation module to carry out stand-alone packet-level simulations in ns-3 to study the XG-PON network in operation, especially the behaviours related to the XGTC layer, in detail.

4.3   Implementation Details of the XG-PON module for ns-3

This section presents the implementation details of the XG-PON simulation module for ns-3.

4.3.1   Overview

Figure 4.1 illustrates a common simulation set-up that uses the simulated XG-PON module and other ns-3 components to study the performance issues that may occur in XG-PON. The OLT is simulated as a node that has an XG-PON related network device (*XgponOltNetDevice*) and another common network device in ns-3, such as a *PointTo-PointNetDevice*, to connect to an external network. Similarly, the ONU is simulated as a node with an *XgponOnuNetDevice* and another network device (Ethernet, WiFi, WiMAX, LTE, etc.) for connecting user equipment to the ONU. In ns-3, network devices
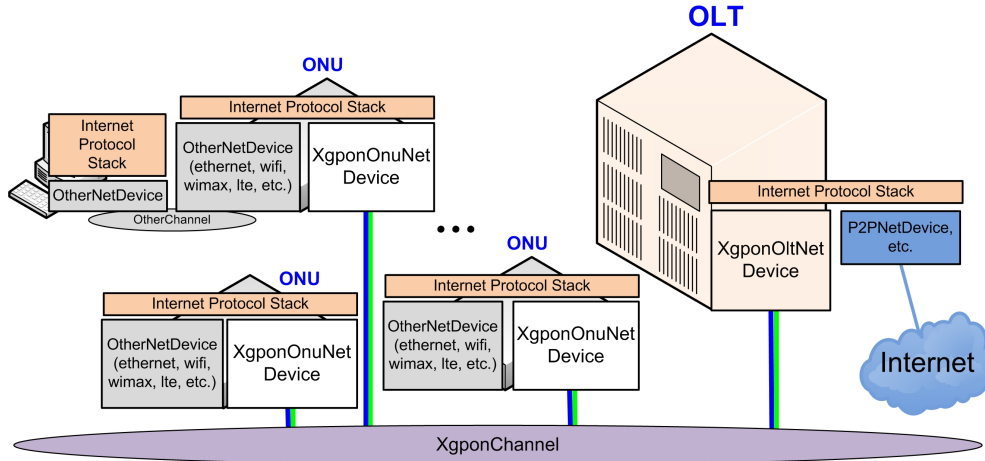
Figure 4.1: Sample XG-PON simulation environment

of a node can be configured in several ways such that different deployment scenarios of XG-PON can easily be configured and evaluated.

The OLT and ONUs are attached to the *XgponChannel* class which represents the physical transmission links in the ODN. As illustrated by Figure 2.1 in Chapter 2, the ODN is a very complex tree composed of optical fibres, splitters/jointers, and reach extenders. To produce the most accurate simulation result, all the components in the XG-PON standard should be simulated. But, as explained in earlier sections, the XG-PON module presented in this chapter simplifies several aspects of XG-PON to minimise the development workload and increase the simulation speed, without a significant impact on the analysis presented in the chapters of this thesis.

Specifically, the *XgponChannel* merely simulates d_{max}, i.e., the logical one-way delay of the channel that is determined by the maximum propagation delay of ODN and various laser on/off delays at the ONU/OLT. d_{max} in the XG-PON simulation module can be configured through the attribute system of ns-3. For a downstream frame from the OLT, *XgponChannel* will pass this frame to each ONU after waiting for d_{max}. *XgponChannel* passes the smart-pointer of this frame to each ONU, which will copy and process the data for itself to avoid unnecessary data copy. As for an upstream PHY burst, *XgponChannel* will pass the corresponding smart-pointer to the OLT, after the appropriate d_{max}. EqD is calculated at the ONU when the upstream burst is produced, based on the $BWmap$ from the OLT; EqD at an ONU is equivalent to the sum of d_{max} and the relative upstream burst delay (with regard to the previously scheduled ONUs in the same upstream frame) perceived at the XGTC layer by the *Framing Engine* of the same ONU.

The XG-PON simulation module also interacts directly with the IP layer with the IP packets being the SDUs, though the XG-PON standard is proposed to carry layer-2 frames of various network technologies (Ethernet, ATM, etc.). This is a reasonable simplification which does not have any significant impact on the performance evaluation
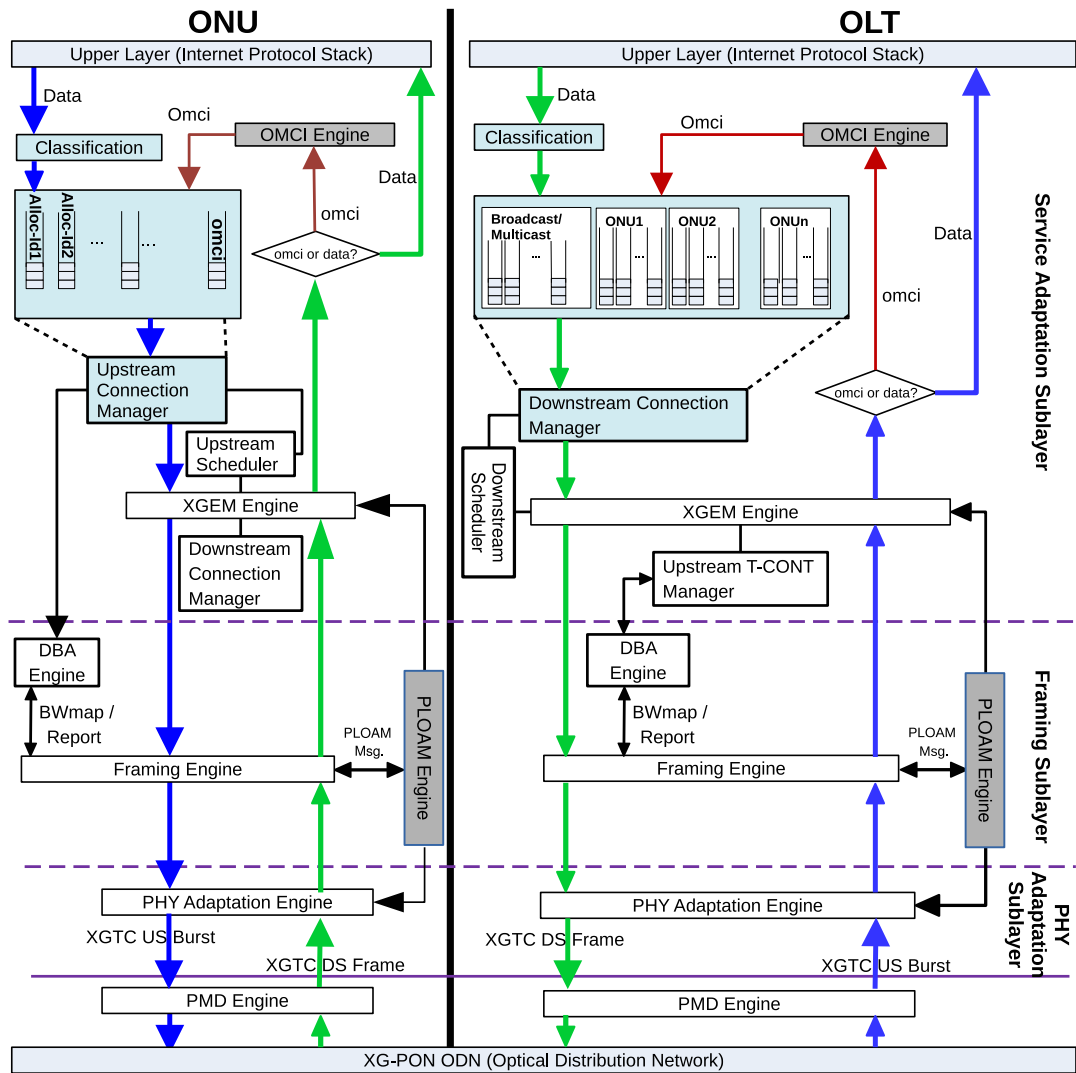
Figure 4.2: Functional block diagram of the XG-PON simulation module

of the QoS policies presented in this thesis, since the focus of this thesis is to study the FTTx networks when connected to technologies comprising IP.

4.3.2 Functional blocks of the XG-PON simulation module

Figure 4.2 illustrates the data transmission paths, both in the downstream and upstream directions within the OLT and ONU, as implemented in the XG-PON simulation module in ns-3. These implementations are based on the XG-PON standard [43] as well as the design principles and decisions presented in sections 4.1 and 4.2, respectively.

4.3.2.1 Downstream traffic from the OLT

As shown by the green arrows in the OLT side of Figure 4.2, when an SDU is received from the upper layer (eg: IP), the SDU should first be mapped to the corresponding connection (XGEM Port) based on the destination IP address and added into the queue for transmitting in the future. Thus, there must be an algorithm for mapping the IP address to the corresponding XGEM Port Id (in the *Classification* box in the figure).

Since the OLT needs to broadcast the downstream XGTC frames every 125 μs, it will periodically ask the OLT's *Framing Engine* to generate an XGTC frame. This engine will first generate an XGTC header since the available space for data in the frame depends on the size of the XGTC header. For the payload of a downstream XGTC frame, the *Framing Engine* resorts to the *XGEM Engine* to get an XGTC payload. This payload comprises of concatenated XGEM frames that can occupy all the available space. As for the SDUs to be encapsulated and transmitted, the *XGEM Engine* lets the *Downstream Scheduler* to decide the downstream connections to be served. This scheduler makes decisions based on *Downstream Connection Manager* which knows the queue length, QoS parameters, and service history of each downstream connection. When carrying out encapsulation, fragmentation will be carried out by XGEM Engine if an SDU is too long for the current downstream transmission opportunity. The XGEM Engine is also responsible for encrypting these SDUs to avoid eavesdropping. The keys used for data encryption are negotiated through PLOAM messages and are maintained by the *Ploam Engine*.

To construct the XGTC header of the frame, the *DBA Engine* is used to generate $BWmap$ which instructs the ONUs on how to share the upstream wavelength. *DBA Engine* makes decisions based on queue occupancy reports, QoS parameters, and service history of the Traffic Containers (T-CONTs). PLOAM messages in the header, are generated by *Ploam Engine*. The downstream frame is sent to the ODN (*XgponChannel*) after passing through *PHY Adaptation Engine* and *PMD Engine*.

4.3.2.2 Downstream traffic at an ONU

As shown by the green arrows in the ONU side of Figure 4.2, when a downstream PHY frame arrives at an ONU, the frame passes through the *PMD* and *PHY Adaptation* Engines which will remove the physical-layer overhead. The *Framing Engine* is then responsible for parsing the resulting downstream XGTC frame.

The PLOAM messages from the XGTC header will be given to the *Ploam Engine*, which will process the messages related with this ONU. The *DBA Engine* is responsible for processing the $BWmap$ in the header for reasons such as scheduling the upstream XGTC bursts if required by the $BWmap$.

As for the payload, the XGEM frames are passed to the *XGEM Engine*. Based on the list of its connections maintained by the *Downstream Connection Manager*, the XGEM frames for this ONU are first extracted. XGEM Engine then carries out de-capsulation, decryption, and reassembly (only if needed[3]). The received SDUs are finally sent to the upper layer (eg: IP).

4.3.2.3 Upstream traffic from an ONU

As illustrated by the blue arrows in the ONU side of Figure 4.2, when an IP packet is received at the ONU, it is first mapped, based on the source IP address, to the corresponding upstream connection, which is maintained by the *Upstream Connection Manager*. The packet is then put into the corresponding queue for transmission in the future.

When it is the time to transmit one upstream XGTC burst (as instructed in the $BWmap$ sent by the OLT in a previous downstream frame), the Framing Engine in the ONU assumes responsibility for producing the XGTC burst. To do this, the *Framing Engine* in an ONU asks its *XGEM Engine* to get an array of XGTC payloads (SDUs), each of which is a concatenation of several XGEM frames belonging to an Alloc-Id. To decide the SDUs to be encapsulated, the *Upstream Scheduler* and the *DBA Engine* in the ONU are also needed since the upstream bandwidth is allocated to each Alloc-Id with the possibility of multiple upstream connections (or Alloc-Ids) belonging to the same ONU. Both the *Upstream Scheduler* and the *DBA Engine* make decisions based on several parameters such as the amount of bandwidth allocated to each Alloc-Id, queue length, QoS parameters, and service history of this T-CONT's upstream connections.

Framing Engine at an ONU also approaches the *DBA Engine* to generate queue occupancy report for the corresponding Alloc-Id, when permitted by the OLT. This report is deduced by the *Upstream Connection Manager* based on the Alloc-Id(s) associated with each ONU. For various purposes, PLOAM messages may be generated by Ploam Engine. When it is allowed by the OLT, one PLOAM message can be put into the header of the XGTC burst, which may contain a range of XGTC SDUs multiplexed when the ONU has several upstream connections/T-CONTs.

The upstream XGTC burst is then passed to the *PHY Adaptation Engine* with the burst profile to be used. After going through the *PMD Engine*, this burst is sent to the ODN (*XgponChannel*).

---

[3]For each downstream connection, the *Downstream Connection Manager* at the ONU should hold the initially received segments for reassembly while the remaining segments are received.

Table 4.1: Major classes of the XG-PON simulation module for ns-3

Classes	Functionality
XgponChannel	Represents the physical medium of ODN
XgponNetDevice	Communicate with both upper layers and *PonChannel*, implements statistics related to OLT and ONU, defines various engines representing protocol stack of XG-PON
XgponDsFrame	Frame transmitted over XG-PON for downstream data
XgponUsBurst	Frame representing upstream burst
XgponXgemFrame	Represent XGEM frame and includes the payload and header
XgponXgemHeader	Represents the XGEM header defined in XG-PON standard
XgponTcont	Represents a T-CONT
XgponTcontOnu	Maintains queue occupancy reports from ONU, QoS parameters and service history of this T-CONT for DBA algorithm. Also holds the received segments for reassembly
XgponOnuConnManager	Contains a list of downstream connections and a list of T-CONTs in each ONU and implements both Downstream and Upstream Connection Managers for the ONU.
XgponOltConnManager	Downstream Connection Manager and Upstream T-CONT Manager for the OLT. Contains broadcast and unicast downstream connections and T-CONTs for upstream connections
XgponPhy	Implements PMD Engine and PHY layer parameters common to OLT and ONUs
XgponOltFramingEngine	Generate the downstream XGTC frames and parse the upstream XGTC bursts in OLT
XgponXgemRoutines	Implements routines common for both OLT and ONU (eg: XGEM frame creation) It also implements encapsulation, decapsulation, fragmentation, reassembly etc
XgponOltDsScheduler	Acts as the OLT Downstream Scheduler shown in Figure 4.2
XgponOltSimpleDsScheduler	A sub-class that follows the round robin (RR) scheme for downstream scheduling
XgponOnuUsScheduler	The ONU upstream scheduler shown in Figure 4.2. Also decides which connection in each ONU to be served in the next transmission opportunity, after payload generation
XgponOnuUsSchedulerRoundRobin	A subclass to serve the T-CONTs of every ONU in a round robin manner
XgponHelper	Helper class for configuring an XG-PON network
XgponConfigDb	Database that holds the information used by *XgponHelper*

4.3.2.4 Upstream traffic at the OLT

As shown by the blue arrows in Figure 4.2, the upstream XGTC burst received from the $XgponChannel$ at the OLT, first, passes through the *PMD* and *PHY Adaptation* Engines. The *Framing Engine* at the OLT is then responsible for parsing the header and the payloads of this burst. If available, the $DBRu$ and the PLOAM message are sent to the *DBA Engine* and the *PLOAM Engine* respectively. As for the XGTC payloads, they are sent to the *XGEM Engine* for de-capsulation and reassembly (if needed). An *Upstream T-CONT Manager* holds the potential segments for reassembly.

As illustrated in Figure 4.2, both OLT and ONU also have an *OMCI Engine* for exchanging OMCI messages that are used for various purposes (ONU management, XGEM Port and T-CONT configuration, etc.).

4.3.3 Major Classes

Major classes used in the XG-PON simulation module are explained here. Some significant sub-modules are also explained in detail, where necessary, as they involve certain design choices. Class names are generally informative and Table 4.1 presents the summary of the major classes. For information on all the classes in the XG-PON simulation module, the reader is referred to the class diagram provided in Appendix A.

Connection Managers: For both the *XgponOnuConnManager* and the *XgponOltConnManager*, two subclasses are implemented, with the relevant data structures organized in different ways for the OLT and the ONU: 1) *XgponOnuConnManagerSpeed* and

XgponOltConnManagerSpeed impose the relationships among XGEM Port-Id,d Alloc-Id, ONU-ID, and IP address of the computer connected to the ONU such that the mapping is faster with the limitation on the number of XGEM Ports that an ONU can have; 2) *XgponOnuConnManagerFlexible* and *XgponOltConnManagerFlexible* do not have such limitations on the number of XGEM Ports, but are much slower. Since millions of packets are processed per second, the first option is recommended for most scenarios.

PMD and PHY Adaptation: *PMD Engine* and *PHY Adaptation Engine* in Figure 4.2 are simplified significantly for simulating XG-PON with reasonable speed. The most important function of the interface here is to tell other classes about the size of a downstream frame or upstream burst.

XgponOltPhyAdapter and *XgponOnuPhyAdapter* are used to implement *PHY Adaptation Engine* for the OLT and ONU, respectively. Instead of simulating their functions (line coding, FEC, scrambling, etc.) step by step, they just pass frames/bursts between the *XgponChannel* and the *Framing Engine* after removing the physical layer header. An implicit assumption is that all frames/bursts can be received correctly. Since the network is generally well planned and FEC has been adopted, a very low frame corruption rate in XG-PON frames and bursts is also a reasonable assumption. In the future, the corruption rate of frames will be simulated based on the distance between OLT and ONU or empirical measurements of XG-PON networks in practical deployments.

DBA: To study different scheduling and DBA schemes, several abstract classes are used in this module for extensibility. The actual schedulers can then inherit these abstractions and implement their specific algorithms.

For example, the *XgponOltDbaEngine* is designed for the OLT *DBA Engine* shown in Figure 4.2. When *XgponOltFramingEngine* generates one downstream XGTC frame, it will resort to the *XgponOltDbaEngine* to generate a $BWmap$. Algorithm 1 presents the pseudo-code of the abstracted bandwidth allocation function (*BW_Allocation_Main()*) which is implemented in the *XgponOltDbaEngine* class of the simulation module. For each upstream frame, *Pick_1st_AllocID_To_Serve ()* (line 2) picks the first $AllocID$ to be provisioned with the first transmission slot in the upstream, *Pick_Next_AllocID_To_Serve ()* (line 11) picks the next $AllocID$ to be served in same upstream frame, if not all the $AllocIDs$ are already served in the given upstream frame (a false output for the if condition in line 8). $GrantSize$ for each $AllocID$ is calculated by the *Calculate_GrantSize ($AllocID_{k,i}$)* function (line 4), based on the queue status report sent $AllocID_{k,i}$ at an earlier upstream frame received at the OLT. At the end of each round of the do-while loop in the algorithm, two conditions are checked (lines 13 and 14) to ensure that, if one of them is breached, the *BW_Allocation_Main()* function can be terminated (indicating the end of $GrantSize$ allocation for the corresponding upstream frame) to return the control back to the *XgponOltFramingEngine* for completing the generation of the current XGTC frame.

Algorithm 1 : BW_Allocation_Main()

This function cycles through all the configured $AllocIDs$ in the XG-PON network and decides the $GrantSize$ to prepare the burst information in the $BWmap$ header section of the downstream frame. The function also assumes that k number of T-CONT types are provisioned in the XG-PON network; $i \epsilon N_k$ where N_k is the number of k T-CONT type $AllocIDs$ in XG-PON. For a simple Round-Robin DBA, suffix k can be ignored in the pseudo-code, as all the $AllocIDs$ have equal priorities; N_k will be equal to N which is the number of ONUs in the XG-PON network.

Pre-defined values: UPSTREAM_FRAME_SIZE is the equivalent amount of bytes for the $125\mu s$-long upstream frame, based on the upstream physical datarate of XG-PON. MAX_ALLOCIDS_IN_CURRENT_UPSTREAM_FRAME is the maximum number of $AllocIDs$ that can be fitted in to a single upstream frame, of size UP-STREAM_FRAME_SIZE.

1: Total_Granted_AllocIDs = Total_GrantSize = 0   ▷ These are local to the current upstream frame
2: $AllocID_{k,i}$ = Pick_1st_AllocID_To_Serve ()
3: **Do**
4:     $GrantSize_{k,i}$ = Calculate_GrantSize ($AllocID_{k,i}$)
5:     Prepare_Burst_Information ($AllocID_{k,i}$, $GrantSize_{k,i}$)
6:     Total_Granted_AllocIDs += 1
7:     Total_GrantSize += $GrantSize_{k,i}$
8:     **if** (Check_All_AllocIDs_Served($AllocID_{k,i}$) == true) **then**
9:       stop bandwidth allocation for this upstream frame
10:     **end if**
11:     $AllocID_{k,i}$ = Pick_Next_AllocID_To_Serve ()
12: **While**
13: {(Total_GrantSize < UPSTREAM_FRAME_SIZE) &
14: Total_Granted_AllocIDs < MAX_ALLOCIDS_IN_CURRENT_UPSTREAM_FRAME)}

For experiments required in this chapter, a simple Round Robin DBA algorithm is implemented in *XgponOltDbaEngineRoundRobin*, which serves fixed amount of bytes (capped by *UPSTREAM_FRAME_SIZE*) for each $AllocID$ in a Round-Robin manner (refer to section 2.4 of Chapter 2 for more details on the scheduler). Round-Robin DBA is implemented by the functions of *XgponOltDbaEngineRoundRobin*, which ensures a continuous tracking of the index of the $AllocID$ served in the current upstream frame, so that at the beginning of every instance of *BW_Allocation_Main()*, the next index of the $AllocID$ (relative to the $AllocID$ served in the previous upstream frame) is fed into the *Pick_First_AllocID_To_Serve()* function. The flow chart for the algorithm can be seen in Figure 4.3

In a similar way (extended implementation of the sub-functions related with the main function, *BW_Allocation_Main()*), several other DBA mechanisms can also be implemented in the XG-PON simulation module, for supporting different types of T-CONTs with many other QoS parameters, while a consistent *BW_Allocation_Main()* function in the parent class.
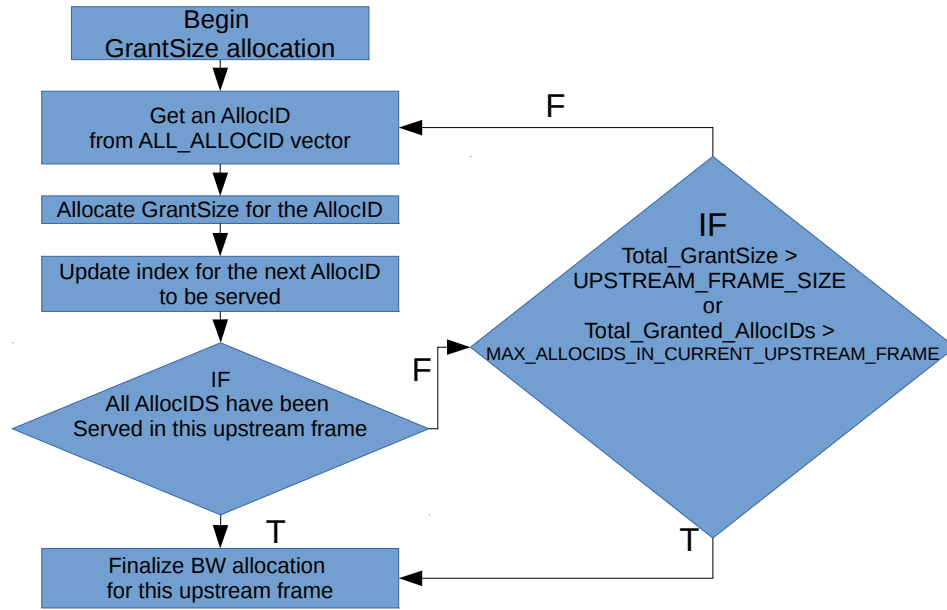
Figure 4.3: Flowchart for the BW_Allocation_Main() in the XG-PON simulation module

XgponOnuDbaEngine acts as the ONU *DBA Engine* shown in Figure 4.2. It is responsible for processing the $BWmap$, to generate the $DBRu$ and schedule the upstream burst.

Helper: For facilitating researchers to configure an XG-PON network with hundreds of ONUs and thousands of connections, a helper is implemented by the *XgponHelper* function. Using the *XgponHelper*, researchers can install the *XgponNetDevice* on nodes and attach them to the *XgponChannel*. They can also configure XGEM Ports and T-CONTs for carrying user traffic. Researchers can also use *XgponHelper* to enable Ascii and Pcap tracing.

Miscellaneous: Further classes of interest in the implementation of the XG-PON module, are listed here.

- *XgponOltPloamEngine* and *XgponOnuPloamEngine* are designed for exchanging PLOAM messages between the OLT and ONU. They also use *XgponLinkInfo* to maintain per-ONU information, such as keys and burst profiles.

- *XgponOltOmciEngine* and *XgponOnuOmciEngine,* are designed for implementing the OMCI channel. For these classes, though, only their interactions with other layers are implemented. Detailed messaging and procedural structure of this engine is left for implementation in the future.

- *XgponOnuUsScheduler* is added within the *XgponTcontOnu* so that T-CONTs of the same ONU may use different scheduling algorithms for their upstream traffic.

- *XgponConfigDb* uses one flag to make sure that *XgponOltConnManagerSpeed*, *XgponOnuConnManagerSpeed,* and *XgponIdAllocatorSpeed* are used together.
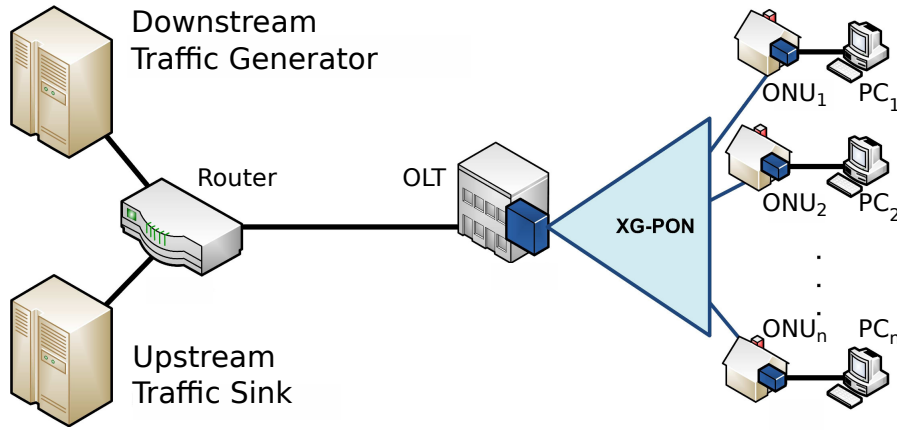
Figure 4.4: Simulated network topology

4.4   Evaluation of the XG-PON simulation module

This section presents the evaluation of the XG-PON simulation module with regard to the functionality validation, performance analysis (simulation speed and memory consumption) and robustness (pressure tests).

Figure 4.4 shows the network topology used for the evaluations. An XG-PON network whose d_{max} is 0.4ms, which is more than the one-way propagation delay in fibre (= 0.3ms) for a refractive index of 1.5 and a physical reach of 60km. For the data rates of XG-PON, the XG-PON1 standard is followed, which is capable of Physical Layer datarates of 10Gbps and 2.5Gbps in the downstream and upstream respectively. There is a total of n ONUs in the XG-PON and a PC is connected to each ONU with a point-to-point link. The PCs act as the customer of XG-PON and play the role of either traffic generators for upstream traffic or sinks for downstream traffic. The delay between each PC and the immediately connected ONU is set to be 2ms, indicating a maximum one-way delay between the user application and the ISP terminal near the user. The OLT is connected to a Router using another point-to-point link, making up a generic core network by an ISP. The delay of this link is set to be 10ms, which is a practical one-way delay between routers at ISP and an OLT placed at the ISP edge of an access network. On the other side of the Router is a Downstream Traffic Generator and an Upstream Traffic Sink, both connected to the Router via individual point-to-point links; a delay of 2ms is used between the Router and the Downstream Traffic Generator (or the Upstream Traffic Sink), to represent a realistic one-way delay between an application and a core router at the ISP. The bandwidth of all the above point-to-point links is set to be 20Gbps so that, in terms of bandwidth, XG-PON is the only bottleneck link in the entire simulation environment. At the application layer, in the Downstream Traffic Generator/PCs, traffic models with uniform inter-packet arrival time distribution are used, to generate the server/user traffic. More specific details of the network traffic will be presented in the respective subsections.
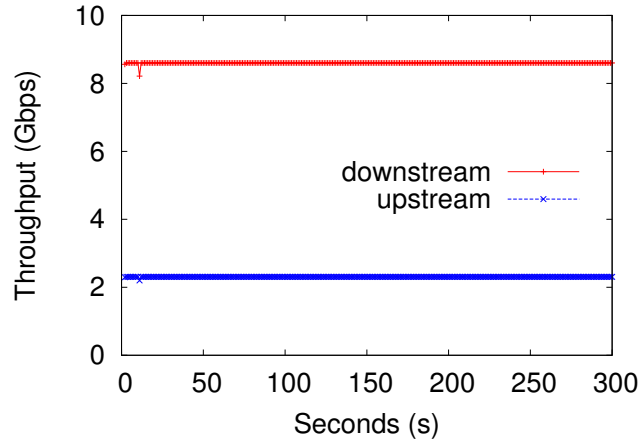
Figure 4.5: Effective bandwidth of the XG-PON simulation module in both directions

4.4.1 Functionality validation

The evaluations in this section focus on validating the generic features of the XG-PON (G.987) standard by the ITU-T, when deployed in realistic network environments. The evaluations 1) quantify the effective bandwidth observed by end-user applications when using XG-PON in the access network, 2) validate the fairness of the implemented Round Robin scheduler, 3) analyse the trade-off between throughput and delay and 4) provide insight into co-existence of the XG-PON simulation module with real-world TCP stacks.

4.4.1.1 Effective bandwidth of XG-PON1

Since the XG-PON module represents the XG-PON1 standard it has a Physical Layer data rate of 10Gbps and 2.5Gbps in the downstream and upstream respectively. Due to the overhead of FEC and the headers of various layers, however, the effective bandwidth observed by an application transmitted across an XG-PON1 is much less. The effective bandwidth is quantified here when large numbers of ONUs are used. The experiment uses 256 ONUs; UDP traffic is used in both downstream and upstream directions, individually, with the overall network load maintained at a higher value than the Physical Layer data rate XG-PON1.

Figure 4.5 plots the total effective bandwidth observed by the applications with time. The overall throughput in downstream is around 8.5Gbps (red line) and upstream is around 2.3Gbps (blue line); these data rates are equivalent to the expected theoretical values, given a packet size of 1024 Bytes and the overheads in XG-PON1 are accounted for. These results also show that the XG-PON module has properly simulated the operation of SDU encapsulation (XGEM and XGTC headers) and other XG-PON1 overheads (FEC, inter-burst gap, etc.).
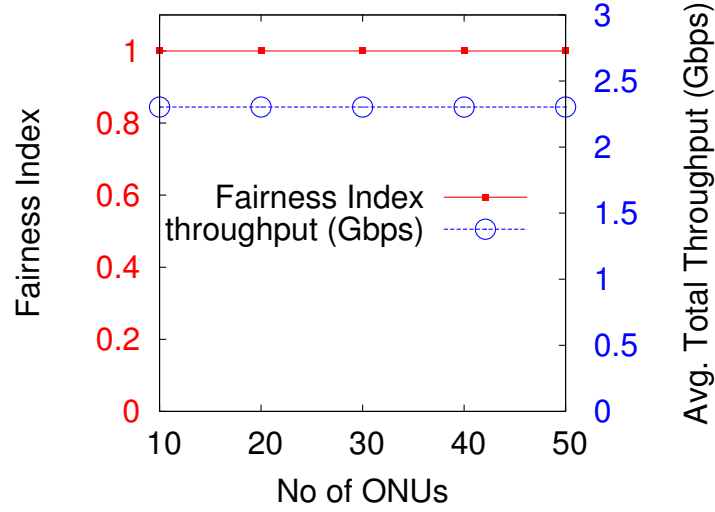
Figure 4.6: Fairness of the Round-Robin DBA algorithm

4.4.1.2 Fairness of the Round-Robin DBA algorithm

The DBA mechanism, which heavily influences the upstream data transfer across XG-PON, is one of the most complex functions of XG-PON. Hence its implementation should be verified. In this experiment, the fairness and the efficiency of the simple Round-Robin (RR) DBA algorithm implemented in the XG-PON simulation module is demonstrated. The DBA is evaluated for five scenarios, each with 10, 20, 30, 40 or 50 ONUs. Each end-user (the PCs connected to ONUs) connected to the ONU generates the same amount of UDP-based deterministic traffic in the upstream, with every ONU connected to a single end-user. The overall network load in the upstream is higher than the upstream data rate (=2.5Gbps) of XG-PON1 to evaluate the efficiency of the DBA when the upstream is (effectively) fully loaded.

Figure 4.6 plots the Jain's fairness index [53] among the end-users (or ONUs) and the overall throughput in the upstream. The red line shows that the fairness index is equal to 1, indicating that the RR DBA algorithm fairly allocates the upstream bandwidth to all the ONUs. The blue line indicates that the Round Robin DBA algorithm is also efficient since the overall throughput is 2.3Gbps in all five scenarios.

4.4.1.3 Trade-off between throughput and delay

The well-known trade-off between throughput and delay seen in large-BDP networks is demonstrated here, since XG-PON is also a large-BDP network due to its Gbps-range bandwidth and millisecond-range delay. That is in a large-BDP network, when a sender is allowed to transmit larger chunks of data at each transmission instance, the client is able to have a larger throughput due to the smaller amount of transmission overhead used per chunk of data; however, since sending larger chunks require many packets to
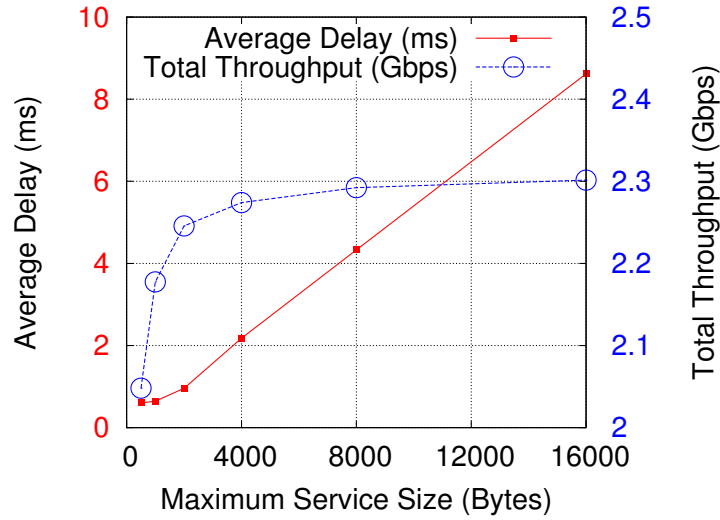
Figure 4.7: Trade-off between throughput and (scheduling) delay in XG-PON upstream

be waiting at the sender buffer, each packet may encounter larger waiting time at the buffer, resulting in higher latency across the large-BDP network.

In XG-PON, this trade-off is expected to be seen between the upstream throughput and scheduling delay achieved by the user(PC) attached to an ONU, when different Maximum Service Size (MSS) values are used by the Round-Robin DBA to serve multiple ONUs. When an ONU gets the opportunity to be served by the RR DBA, the MSS value used by the DBA algorithm serves as the upper bound for the size of the upstream burst (payload) admitted from the ONU queue into the upstream frame. As a result, larger MSS values dictate a smaller number of large payloads to be packed in a single upstream frame, though the burst overhead per payload remain fixed regardless of the values of MSS and the payload size. Hence, larger MSS values results in smaller percentage of burst overhead in each upstream frame, thereby achieving higher throughput per user. However, when multiple ONUs are served in the upstream by the RR DBA, larger MSS values result in higher interval (scheduling delay) between two consecutive upstream bursts transmitted from the same ONU queue, due to the fixed length (125μs) of each upstream frame, resulting in the famous trade-off between the throughput and scheduling delay experienced by each user in the XG-PON upstream.

The experiment, designed to validate the trade-off between throughput and delay, uses 256 ONUs for simulation (using the figure 4.4) with each user (attached to a single ONU) generating the same amount of deterministic upstream UDP traffic in the upstream; overall network load in upstream is higher than the upstream capacity of XG-PON. MSS at the RR DBA is varied with the values of 500B, 1KB, 2KB, 4KB, 8KB, and 16KB.

Figure 4.7 illustrates the impact of MSS on the throughput (measured at the Upstream Traffic Sink in figure 4.4 and accumulated for all the users) and the average (schedul-
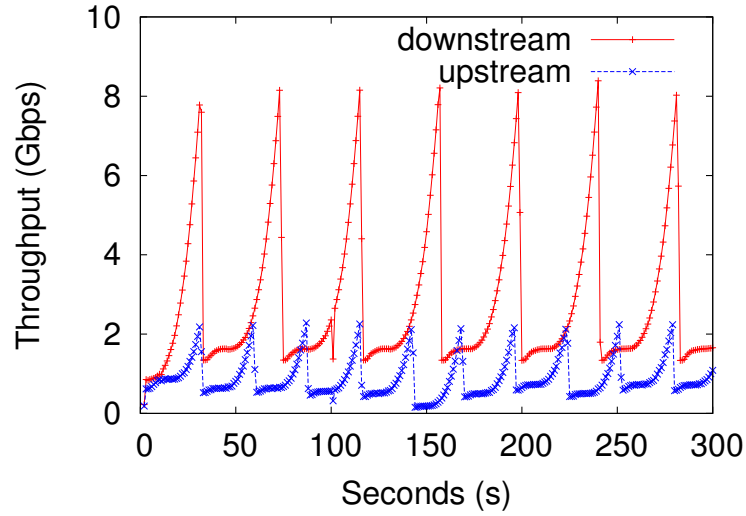
Figure 4.8: Performance of a TCP flow across the XG-PON downstream and upstream

ing) delay, measured for and averaged across all the upstream transmission opportunities provided by the Round Robin DBA for all 256 ONUs. From the figure 4.7, it is clear that both the total upstream throughput (blue line) and the average scheduling delay (red line) increase with MSS, indicating the existence of the well-known trade-off between throughput and delay in XG-PON, which is a large-BDP network.

4.4.1.4 TCP in downstream and upstream directions

The experiments presented here are designed to demonstrate that the XG-PON simulation module is also compatible with realistic TCP models. A single TCP connection is simulated in the downstream and upstream, individually. TCP model from the Network Simulation Cradle (NSC [54]), with CUBIC TCP [33] as the congestion control algorithm is used for the evaluations. NSC is a framework which allows real-world TCP/IP network stacks to be used in a simulation, thus providing highly-accurate models of real-world TCP stacks.

Figure 4.8 shows the throughput vs. time plots for a single TCP connection in the downstream (red line) and the upstream (blue line). When the data rate at the sender is higher than network bandwidth, packets are dropped and sending rate (or throughput) is reduced. The congestion window growth function (not shown here) of Cubic TCP also matched well with the growth pattern of the throughput curves in Figure 4.8, validating the behaviour of and throughput achieved by a single TCP connection in upstream and downstream. This indicates that the XG-PON module is capable of working with the real-world TCP stacks. An extensive study of the TCP and the XG-PON module can be found in the next chapter.

The results pertaining to the functionality clearly demonstrate that the XG-PON module, designed for simulating a 10Gbps (downstream) optical network with hundreds of ONUS, is indeed capable of simulating a standard-compliant XG-PON1 network with known accuracy at the evaluated functionalities.

4.4.2 Performance evaluation

The performance evaluation of the XG-PON simulation module with regard to its simulation speed and memory consumption is presented here. A dedicated computer, running only the system-level processes is used to measure the performance of the simulation module, to avoid interference from other processes. The off-the-shelf computer used is Dell PowerEdge R320 rack server and with the Intel(R) Xeon(R) CPU E5-1410 0 processor of 2.80GHz speed and 10MBytes cache. Note that although this processor has 4 cores, only one of them is used by ns-3 (and the simulation module) for the experiments. The server is installed with 48GBytes total main memory.

To study the simulation performance of the XG-PON simulation module under various scenarios, the number of ONUs and the amount of network traffic the experiments are varied. Number of ONUs (N) are varied with values of 25, 50, 100, 200, 400, 800, and 1000. For each value in N, UDP-based deterministic downstream traffic is generated for a total load of 150Mbps, 300Mbps, 600Mbps, 1.2Gbps, 2.4Gbps, 4.8Gbps, and 9.6Gbps. Due to the overhead of XG-PON physical and XGTC layers, packet drop is observed only when the downstream network load is more than 9.6Gbps. In all the experiments, the upstream network load is set to be one-quarter of the corresponding downstream load. For instance, when the downstream network load is 9.6Gbps, the upstream network load is 2.4Gbps, with observable packet drop in the upstream direction (due to the difference in burst overhead percentage in downstream and upstream).

To evaluate the speed of the XG-PON simulation module, the total amount of time used to complete each 400-second-simulated experiment is measured; the average time consumed for a single simulation second in each scenario is then calculated and plotted in Figure 4.9. For each value of N, the results indicate that the average time consumed per simulation second increases linearly with the network load. This is reasonable since ns-3 is a packet-level network simulator and the number of events increases linearly with the number of packets. With the increase in the number of ONUs, the consumed time also increases at a relatively slower pace. At the extreme case of $N = 1000$ (ONUs) and the downstream network load is 9.6Gbps, the XG-PON module only consumes 160 seconds of duration on average for a simulation second.

The most difficult scenario (ONUs: 1000; Network load: 9.6Gbps) is also simulated in the *debug* mode of ns-3, using *gdb*, which is the standard debugger for the GNU software system. After the simulation enters the steady-state in the *debug* mode, the
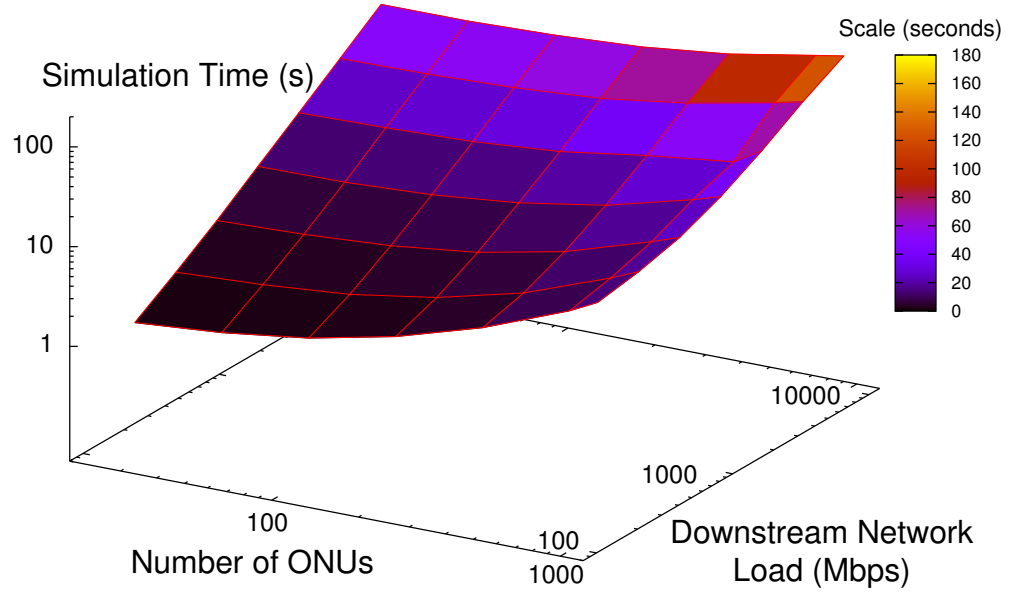
Figure 4.9: Simulation speed of the XG-PON simulation module
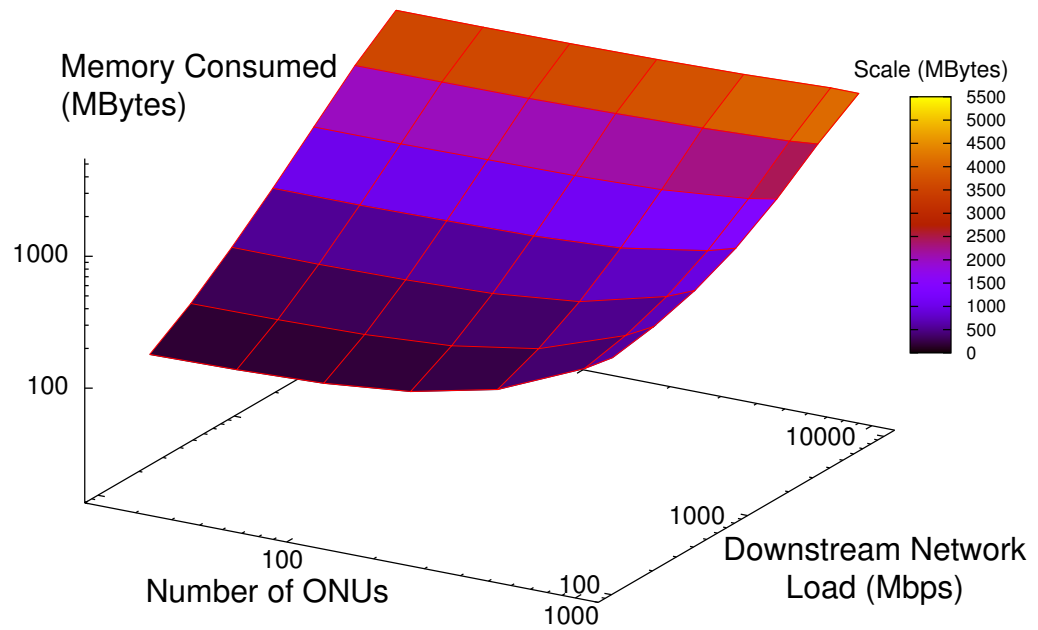
Figure 4.10: Memory consumption of the XG-PON simulation module

experiment is paused (*break* command in *gdb*) at a random time, to check the latest
call stack, and the simulation then continues. The pause-check-continue procedure was
repeated for 126 times during the *debug* mode experiment. In total, code of the XG-
PON simulation module occupied the CPU only for 4 of the 126 times, with the code
of other modules in ns-3 occupying rest of instances (122 times). This clearly indicates
that the XG-PON simulation module has negligible influence on the bottleneck of its
simulation speed while prompting for additional investigation of other associated ns-3
modules (routing, etc.) to further improve simulation speed of the XG-PON module.

In evaluating the memory consumption for the XG-PON simulation module, the same
above experiments were repeated. For each experiment, after starting the simulation,
the simulation is allowed to enter into its steady-state (by waiting for sufficiently long
duration), where the total amount of consumed memory by the complete ns-3 simu-
lator (including the XG-PON simulation module as well) does not increase with time.
Figure 4.10, plotted using the steady-state memory consumption for each combination
of N and downstream network load, indicate that the consumed memory (in steady-
phase) increases linearly with the network load while it increases at a lesser pace with
the number of ONUs. At the extreme case of 1000 ONUs and the downstream network
load is 9.6Gbps, the consumed memory was less than 5GBytes.

Based on the above results, it can be concluded that with the off-the-shelf server, the
XG-PON simulation module can simulate an XG-PON network of up to 1000 ONUs and
10Gbps (downstream), with reasonable speed and moderate memory consumption.

4.4.3   Pressure tests

The results of the additional experiments performed to evaluate the robustness of the
XG-PON simulation module are presented here. The same configurations used for the
earlier performance evaluation experiments are used here with the exception of the
simulation duration, which is set to 4000 seconds in the experiments presented here,
to demonstrate that the XG-PON simulation module can run for sufficiently long pe-
riods (longer than one hour) of time. Note that when there are 1000 ONUs and the
downstream network load is 9.6Gbps, it took more than one week to complete the sim-
ulation. In another set of 49 experiments, a random selection is made for the number
of ONUs and the amount of network traffic, and 500 seconds were simulated in each
experiment. All these simulations ran successfully without any interruption or memory
leak during the experimentation procedure, indicating the robustness of the XG-PON
simulation module in being a simulation platform for large-scale XG-PON-based evalu-
ations.

4.5   Conclusion

This chapter introduced the XG-PON simulation module designed for the ns-3 network simulator. The design and implementation details of the simulation module is presented in detail, to easily understand the principles behind and implementation of the XG-PON module. The evaluation results on performance and functionality demonstrated the high degree of robustness and accuracy of the XG-PON simulation module, which is also reasonably fast and uses a moderate amount of memory. Hence, the XG-PON simulation module for ns-3 presented in this chapter, while being the first XG-PON-standard-compliant simulation platform, is a significant contribution to the wider scientific community. The simulation module also provides the opportunity for any interested researcher to study the performance issues associated with realistic deployment scenarios associated with XG-PON, with quantified confidence.

Chapter 5

Analysis of TCP across the large-BDP network, XG-PON

This chapter presents the evaluations with regard to the influence of TCP flows across an XG-PON network, which is a large-BDP network both in the downstream and upstream directions.

XG-PON is a good example of a large BDP network, due to the very high datarate in the downstream (10Gb/s) and upstream (2Gbps in XG-PON1) directions, indicating a challenging environment for the performance of TCP in both directions. Though the propagation delays are small in both directions, the added and non-uniform scheduling delays among multiple ONUs in the XG-PON, increases both the large-BDP nature of XG-PON in the upstream and any potential challenges presented for the co-existence of TCP flows in the XG-PON.

Though there are several studies for the co-existence of TCP in different types (eg: satellite communication, EPON, etc.) of large-BDP networks (see section 3.3.2 of Chapter 3 for literature review on the challenges for TCP flows in large BDP networks), no study is presented for the challenges for TCP flows across the XG-PON network. Since such challenges are significant for the design of suitable QoS policies for the XG-PON and for the choice of the Transport Layer protocol in the evaluations of the DBAs, this chapter presents evaluations regarding the co-existence of TCP flows across a realistic deployment of XG-PON. TCP is evaluated for the metrics of efficiency, fairness, responsiveness and convergence, using the ns-3-based standard-compliant XG-PON simulation model presented in Chapter 4. Efficiency is evaluated in both the downstream directions using NewReno, CUBIC and H-TCP congestion control algorithms, for a range of RTT values while the rest of the metrics are evaluated only in the upstream direction of XG-PON, using CUBIC TCP algorithm and simplified simulation scenarios. Comparisons and discussions with regard to the performance of UDP flows are presented for quantifying the relative behaviour of TCP across the large-BDP XG-PON network.

The results presented in this chapter indicate that, in terms of efficiency, while single TCP flows do not fully utilise the capacity of XG-PON in both directions, larger number of TCP flows behave similar to UDP and utilise the capacity efficiently due to the small fluctuations in throughput (or congestion window, $CWND$) caused by the congestions control algorithms in TCP. These large fluctuations also cause TCP to behave unpredictably, be it when tested for responsiveness for UDP pulses or evaluated for convergence with newly introduced TCP flows. The results, therefore, prompt for the evaluation of a suitable DBA for the FMC of XG-PON and LTE, first using the UDP traffic before proceeding to evaluate the performance with TCP flows.

This chapter presents these findings, starting with a brief explanation of the challenging interaction between TCP and large-BDP networks in section 5.1. The details of the simulation environment are presented in section 5.2. The results and discussions are presented in section 5.3. The chapter is summarised and concluded in section 5.4.

5.1 TCP in a large-BDP network

BDP is a measure of how much data needs to be put on the link at one time (size of the $CWND$) to fill the pipe. For example, for a cross-continental 10 Gb/s link from Europe to USA that has 100ms RTT the BDP is 10 Gb/s * 0.1 s = 1 Gigabits (120MB). The congestion window has to be huge (120MB). Standard TCP (eg: NewReno), by increasing the congestion window by one every RTT, would reach full sending rate approximately within one hour, if it is assumed that none of the packets are lost. Even though loss rate cannot be ignored in realistic networks, having one hour under no-loss assumption is a huge amount of time to spend on reaching full network datarate[90].

Links with high bandwidth and great latency (large-BDP) cause a situation where TCP is mostly inefficient. Since XG-PON is also a large-BDP network both in the downstream merely due to the very large datarate and in the upstream due to the combined effect of the large datarate and the added scheduling delays (on top of the propagation delay), the problem of TCP inefficiency across the XG-PON cannot be ignored.

5.2 Experimental Environment

This section presents the details of the simulation environment designed to evaluate the challenges faced by the three variants of the congestion control algorithms in TCP. The three variants consist of the most popular TCP congestion control in the Internet in NewReno, and two high-speed TCP variants in CUBIC and H-TCP so that the performance of TCP-flows in XG-PON can be analysed for two common forms of TCP variants (traditional and high-speed variants). A detailed description of all three TCP variants
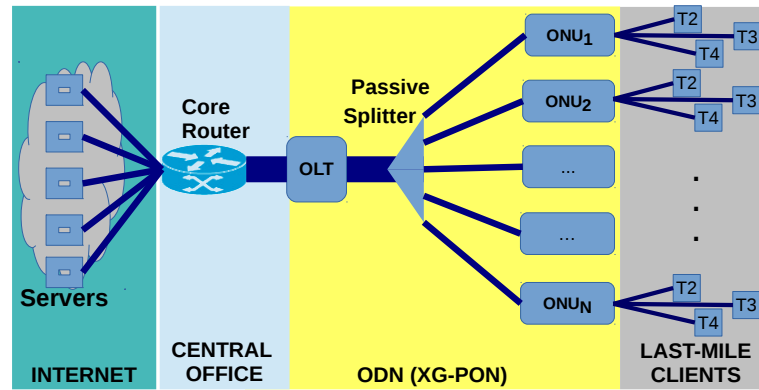
Figure 5.1: Network set-up used for simulations

can be found in section 2.5.2 of Chapter 2. The section first begins with the intro-
duction of the network architecture, which is followed by the details of the evaluation
methodology; evaluation metrics are presented at the end of the section.

5.2.1 Network Architecture

The main goal of the experiments is to analyse the challenges faced by the TCP variants
for applications in realistic XG-PON deployment scenarios. Figure 5.1 represents the
network architecture used in the evaluations in this chapter. XG-PON is placed between
the *Central Office* on one side (near OLT) and the clients on the other (last-mile) side;
the last-mile clients, representing the fixed-broadband users served directly by the XG-
PON, are directly connected to the ONUs; the Core router, directly connected to the
OLT, represents an edge/gateway router in the Central Office of the service provider.
The servers in the Internet are connected directly to the Core Router, representing
application agents or sinks near the service provider.

5.2.2 Evaluation Methodology

The following details are significant to the evaluations presented in this chapter:

- The network architecture in Figure 5.1 is implemented in ns-3, which is the simu-
  lation platform for the evaluations in this chapter. The XG-PON simulation mod-
  ule presented in Chapter 4 is used to configure the XG-PON in the figure.

- To represent realistic TCP stacks, NSC is used at the Transport layer. The linux-
  2.6.26 TCP code of NSC (version 0.5.3) was used to simulate the behaviour of

NewReno, CUBIC and H-TCP. Selective Acknowledgement[1] and Window Scale [2] options were enabled in NSC to effectively utilise the E2E network capacity and to achieve huge values of $CWND$ at the TCP sender, respectively, take advantage of the XG-PON network's large bandwidth in upstream and downstream at a quicker pace, even when a single TCP flow is used. Since NSC was originally designed for ns-2 network simulator, the $NscTcpL4Protocol$ class, which originally interfaced ns-2 with NSC, was modified to fit the interface between ns-3 and NSC.

- The only bottleneck link in Figure 5.1, with regard to bandwidth, is the XG-PON network, so that the challenges faced by the TCP variants across the XG-PON network are identified easily; all other links are configured with higher datarates than the XG-PON downstream or upstream physical datarates.

- All nodes are configured with buffer values higher than the link BDP and all application sources have sufficient (unlimited) amount of packets at the application layer, to transmit data for the full duration of each simulation.

- One way propagation delay ($p2pDelay$) between any two nodes (except between the OLT and the ONU) are manually configured. Thus, the E2E RTT between a server and the client (or vice versa) can be given by,

$$RTT = 6 * p2pDelay + delay_{scheduling}$$

accounting for 6 point-to-point links (excluding OLT-ONU link) in the network architecture $delay_{queuing}$ is the queuing-delay experienced by the transmitted packets at the OLT or ONU buffer. Since only the throughput related evaluations are presented in this chapter, the E2E RTT is calculated based on an assumed $delay_{scheduling}$ value of 2ms, which is a reasonable assumption for 0.4ms of one-way propagation delay for 60km (see equations 2.2 and 2.3) between the OLT and ONU.

- To send traffic in the downstream direction of XG-PON, traffic sources are placed in the servers in the Internet and traffic sinks are placed in the clients in the last-mile; for the upstream direction, traffic sources are placed in the clients and the traffic sources are placed in the servers. The *BulkSendApplication* in ns-3 is used to generate application traffic in the traffic sources, while *OnOffApplication* was used to generate UDP-based traffic; application packet size and Maximum

---

[1]The normal ACK procedure in TCP acknowledges segments only until the highest contiguous segment, even though more segments may have been received at the receiver with non-received segments in the middle of a $CWND$-long segments. Selective Acknowledgement extension[67] allows TCP receiver to send ACKs in blocks so TCP segments are ACK-ed regardless of the non-contiguous nature of received segments, to minimise the E2E delays at the TCP, caused by the duplicate ACKs

[2]Window Scale extension[51] in TCP provides an extended length of 32 bits (from the original 16 bits) along with a scale factor to carry this 32-bit value in the 16-bit Window field of the TCP header, so that very large values of $CWND$ (up to 2^{32} KBytes) can be achieved at the TCP to be able to utilise up to the full capacity of XG-PON, even when using a single TCP flow

Segment Size are fixed at 1024 Bytes and 1500 segments respectively, to minimise segmentation in TCP and XG-PON respectively.

5.2.3 Evaluation Metrics

The evaluations in this chapter are based on the following four metrics:

1. **Efficiency:** Efficiency refers to the utilisation of the effective XG-PON bandwidth in the downstream and upstream directions. A single TCP flow is simulated, in both downstream and upstream directions to estimate the overall efficiency of different TCP congestion control algorithms when subjected to the large BDP nature of XG-PON.

2. **Fairness:** Since the DBA is only employed in the upstream, fairness among the TCP flows from different ONUs are significant when subjected to an unbiased Round-Robin DBA. Fairness is measured using the below equation for the Jain's Fairness Index (FI [53]), based on the throughput achieved per ONU ($throughput_i$) in the upstream of XG-PON.

$$FI = (\sum_{i=1}^{N} throughput_i)^2 / [N * \sum_{i=1}^{N} (throughput_i)^2]$$

   where N is the number of ONUs in the XG-PON network.

3. **Responsiveness:** Here, the TCP flow is tested when short UDP pulses (ON-OFF model) are introduced during the stead-state of the TCP flow. 40% and 70% of the effective upstream capacity of XG-PON are used as pulse datarates during ON duration, to represent different upstream loads of UDP clients in access networks. Pulse-width (ON duration) is varied to represent different lengths of UDP bursts.

4. **Convergence:** Convergence of TCP flows is evaluated by introducing additional TCP flows to an existing steady-state system in the upstream. This is used, in comparison with the convergence experienced by the UDP flow to indicate the impact of DBA on the relative convergence time taken for the TCP flows, in terms of number of TCP flows in the network.

Fairness, responsiveness and convergence are tested only in the upstream direction, where the Round-Robin DBA is used for scheduling upstream transmission opportunities between different ONUs, each with a single buffer to store the traffic in the upstream direction of the XG-PON. XG-PON does not use a DBA in the downstream. This chapter also follows the recommended guidelines by Yee-Ting et.al[60], who stressed the need to perform TCP-based evaluations using different TCP variants for networks which effectively use the congestion control algorithms in TCP (by eliminating hidden bottlenecks in places other than where needed) in a range of network conditions (effi-
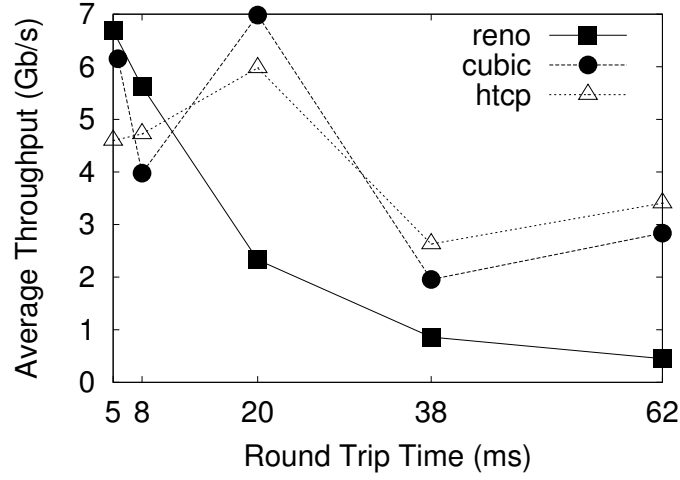
Figure 5.2: Average throughput for all three TCP variants in the downstream

ciency, fairness, responsiveness and convergence), to ensure comprehensive analysis of TCP is performed across large BDP networks.

5.3 Results and Discussion

This section discusses the evaluation results obtained for the interactions between TCP (and UDP) -based traffic in XG-PON, in terms of the metrics, efficiency, fairness, responsiveness and convergence, as explained in section 5.2.3).

5.3.1 Efficiency

Here, a single TCP flow with either the NewReno, CUBIC or H-TCP congestion algorithm, is simulated, both in downstream and upstream, for $p2pDelay$ values of 0.5ms, 1ms, 3ms, 6ms and 10ms (RTT values of 5ms, 8ms, 20ms, 38ms and 62ms respectively). The simulation ran for 200s in each instance and is plotted for the average throughput for each TCP variant, for each RTT value. Average throughput is calculated with from equation $\frac{TotalSentBytes_{t2} - TotalSentBytes_{t1}}{t2 - t1}$, where t1 and t2 are equal to 30s and 180s respectively, corresponding to steady-state behaviour of the congestion control algorithms in TCP between t1 to t2.

In the downstream, as in Figure 5.2, NewReno, CUBIC and H-TCP show poor performance for average throughput as RTT was increased from 5ms to 62ms. Firstly, NewReno shows a linear decrease of average throughput, owing to slow, linear $CWND$ increment, as RTT increases. Secondly, even though both CUBIC and H-TCP are able to perform better than NewReno in higher RTT/BDP values, they are unable to maintain an average throughput value closer to 9,6Gb/s, which is the effective maximum
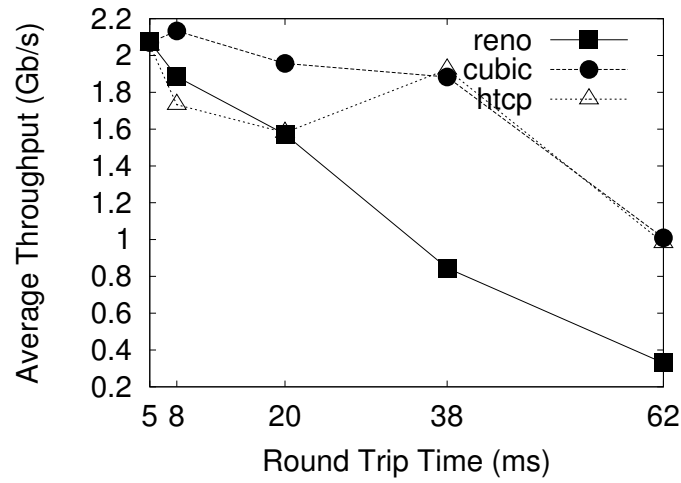
Figure 5.3: Average throughput for all three TCP variants in the upstream
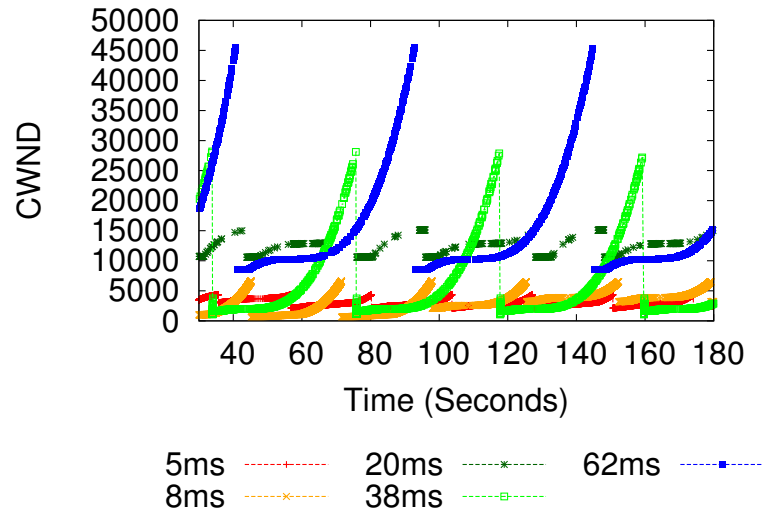
Figure 5.4: CWND growth for CUBIC TCP for all RTT in the downstream
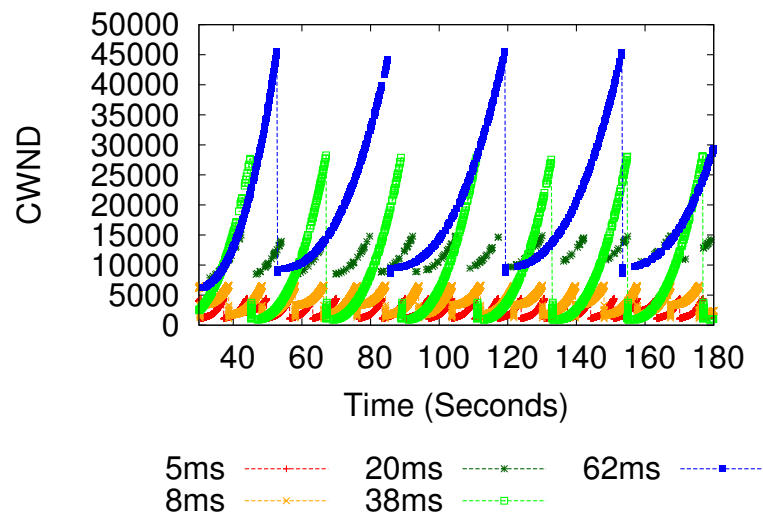
Figure 5.5: CWND growth for H-TCP for all RTT in the downstream
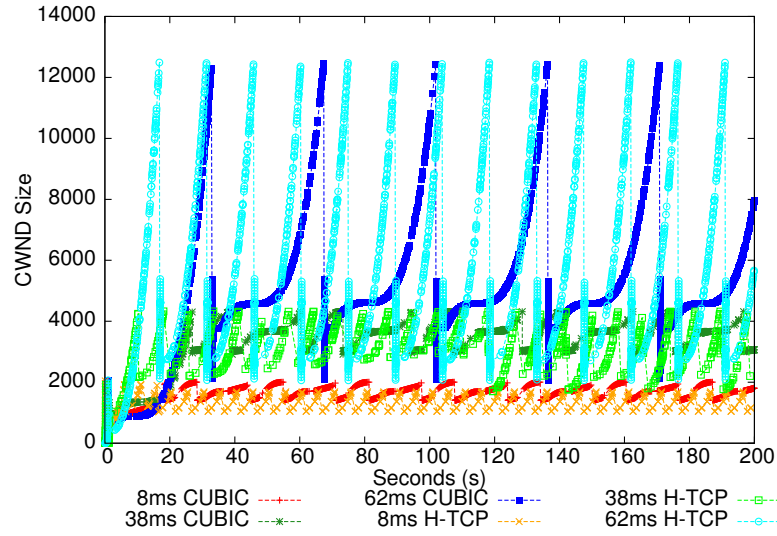
Figure 5.6: CWND growth of CUBIC and H-TCP in upstream for some RTT

throughput achievable by the traffic in XG-PON downstream. Figures 5.4 and 5.5 present the reasons behind such poor performance of CUBIC and H-TCP. As in Figure 5.4, for RTT=5ms and 20ms, since CUBIC is able to demonstrate less variation between the highest and lowest $CWND$ achieved in each congestion epoch, it is able to have very high values for the average throughput, closer to the effective capacity of XG-PON downstream (=9.6Gb/s). The same is observable for H-TCP for RTT=20ms as in Figure 5.5. However, with RTT=8ms and 38ms, both CUBIC and H-TCP experience RTO at the end of every congestion epoch, forcing them to restart their $CWND$ growth from *Slow Start* phase. For the highest RTT of 62ms, both CUBIC and H-TCP achieve higher average throughput than for RTT=38ms, by not experiencing RTO, due to having a comparatively less exponential increment towards the end of each congestion epoch.

Between CUBIC and H-TCP, CUBIC shows higher average throughput, only when its congestion epoch demonstrates an equal concave and convex $CWND$ growth (RTT = 5ms and 20ms), in contrast to a fully exponential growth in H-TCP. But, when the convex region takes most part of each congestion epoch, H-TCP outperforms CUBIC.

Similar to that in the downstream, average throughput performance degrades for each congestion control algorithm in the upstream direction (Figure 5.3). The same explanations of the downstream apply for the $CWND$ growth of NewReno, CUBIC and H-TCP, in the upstream as well. $CWND$ growth plot for upstream of CUBIC and H-TCP for RTT=8, 38 and 62ms is given in Figure 5.6.

When using the UDP traffic in an XG-PON serving a single ONU, average throughput values of 9.6 Gb/s and 2.29 Gb/s were observed in the downstream and upstream directions respectively.
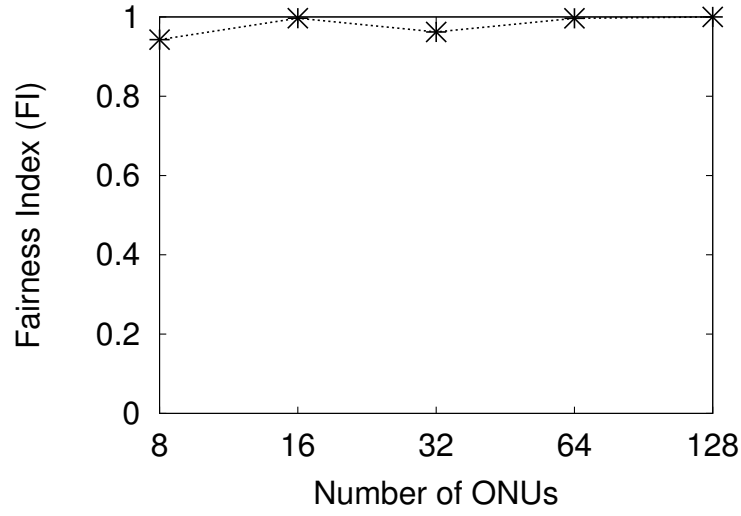
Figure 5.7: Fairness Index (FI) for multiple TCP flows in the upstream
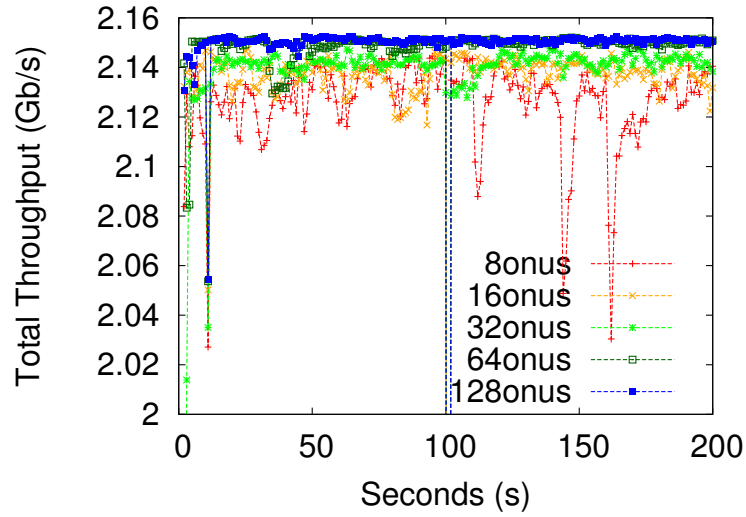
Figure 5.8: Total Throughput for multiple TCP flows in the upstream

5.3.2   Fairness

Fairness of multiple TCP flows was evaluated with and without UDP traffic in XG-PON, to represent the fairness performance of the TCP flows against the unbiased Round-Robin DBA in a realistic mixture of traffic flows.

5.3.2.1   Fairness Among multiple TCP flows with no background UDP traffic

To evaluate fairness among multiple TCP flows, XG-PON is configured with multiple number of ONUs (8, 16, 32, 64 and 128), each with a single upstream TCP flow. Each 200s-long simulation was performed with the CUBIC congestion control algorithm in TCP and an RTT of 62ms. The Jain's Fairness Index for all five scenarios are almost equal to 1, as seen in Figure 5.7).
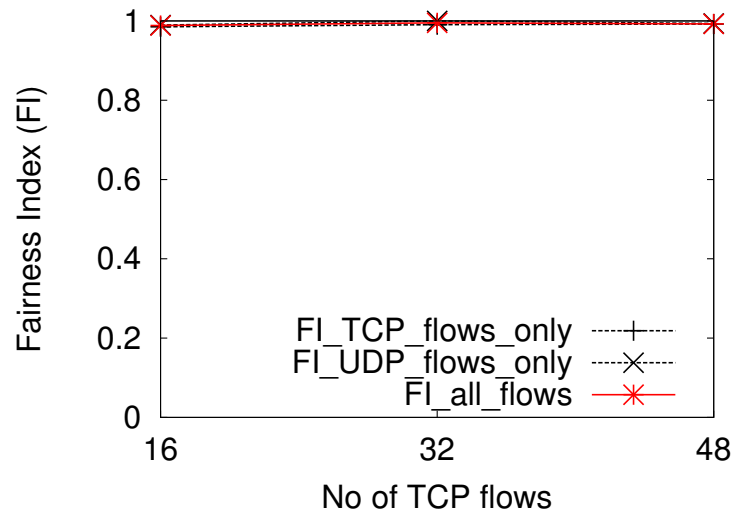
Figure 5.9: Fairness Index (FI) for multiple TCP and UDP flows in the upstream
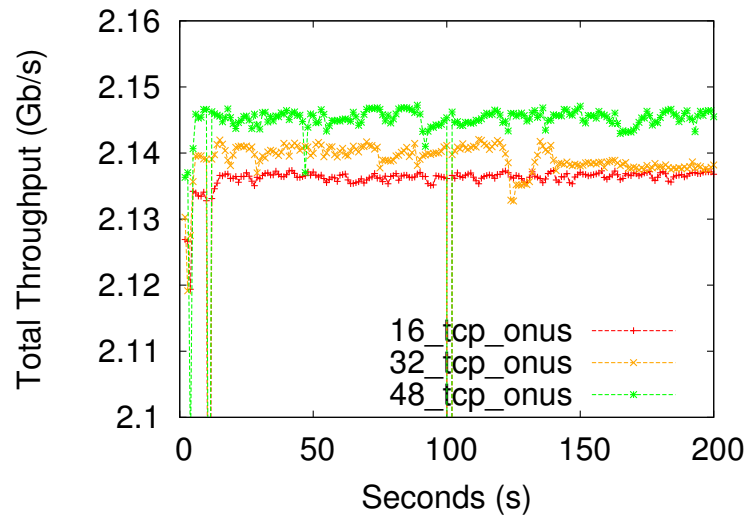
Figure 5.10: Total Throughput for multiple TCP and UDP flows in upstream

As the number of ONUs increases, total throughput through XG-PON has also increased as in Figure 5.8. That is, as the number of TCP flows increases, per-flow throughput becomes smaller, due to the adjustment CUBIC makes to its $CWND$ growth pattern to share the upstream capacity of XG-PON, equally, among all the flows; thus, each flow can only reach a smaller value for the maximum $CWND$, causing fewer fluctuations for the $CWND$ when congestion is experienced per flow. As a result, total throughput in the upstream increases when the XG-PON transmits more simultaneous number of TCP flows.

5.3.2.2   Fairness among multiple TCP flows with background UDP flows

To evaluate how the unbiased Round-Robin DBA would treat different types of traffic, a fixed number of 64 flows is simulated in the 3 scenarios below, with each ONU having either a single TCP or UDP flow:

- S1: 16 TCP flows and 48 UDP flows

- S2: 32 TCP flows and 32 UDP flows

- S3: 48 TCP flows and 16 UDP flows

As seen in Figure 5.9, the Round-Robin DBA is able to provide a very high fairness to all the flows, regardless of the nature of Transport layer protocol used. However, Figure 5.10 shows an improved throughput from S1 to S3, even though the total number of flows through XG-PON remains the same at 64. This is for the same reason as in section 5.3.2.1, where the throughput in XG-PON upstream increases with the increasing number of TCP flows in the upstream.

5.3.3   Responsiveness

In the upstream, XG-PON throughput is governed by a DBA mechanism at a minimum frequency of 125μs. However, a single TCP flow transmitted across XG-PON operates at tens of ms-range latencies, resulting in $CWND$ changes at ms-range frequencies. Due to the difference in frequencies at different orders of magnitude, we expect a complex response from an existing (steady-state) TCP flow when introducing a new UDP pulse in the upstream. The UDP pulse used here is a highly abstracted representation of bursty upstream traffic generated by a fixed-broadband user.

Figures 5.11 and 5.12 evaluate the responsiveness of a single CUBIC TCP flow (RTT = 62ms) in the upstream when continuous short (10s pulse-width) and long (40s pulse-width) UDP pulses are introduced in the upstream, respectively. In both figures, the reaction of the TCP flow is evaluated when introduced with a UDP pulse, whose bandwidth is 40% and 70% of the upstream capacity of XG-PON. For 40%-pulses, the Round-Robin DBA fully grants the required upstream capacity in XG-PON for both short (Figure 5.11a) and long (Figure 5.12a) UDP pulses, while the co-existing TCP flow is throttled at the introduction of the UDP pulse due to accumulated throughput of TCP flow and UDP pulse reaching the upstream capacity of XG-PON. As the UDP pulse is removed, each TCP flow continues to grow its $CWND$ (as seen in its throughput behaviour) based on the congestion control's original equation and throttled only by the upstream capacity of XG-PON.

For 70%-pulse scenario, the TCP flow in short (Figure 5.11b) and long (5.12b) pulse-width scenarios occupies a greater share of the unused (by UDP) XG-PON upstream
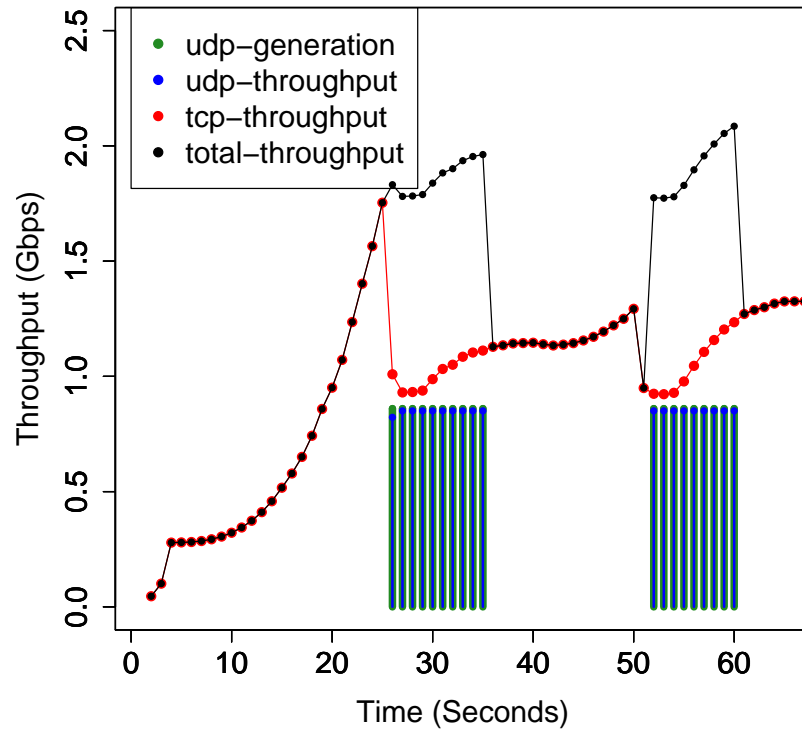
(a) Pulse Height = 40% of XG-PON capacity
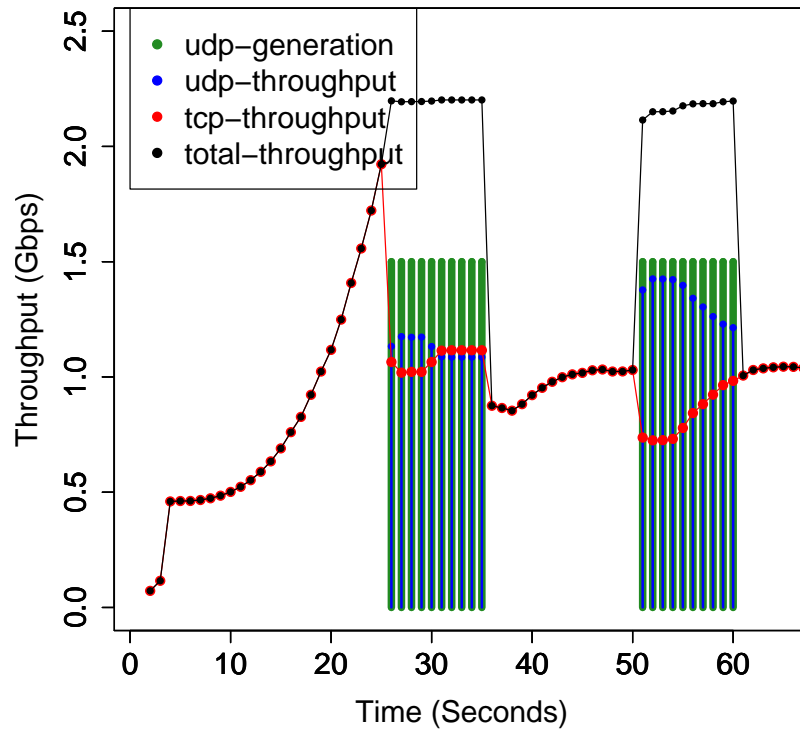
(b) Pulse Height = 70% of XG-PON capacity

Figure 5.11: Responsiveness of a CUBIC TCP flow to a UDP pulse, starting at 25s, pulse-width = 10s and pulse-gap = 15s
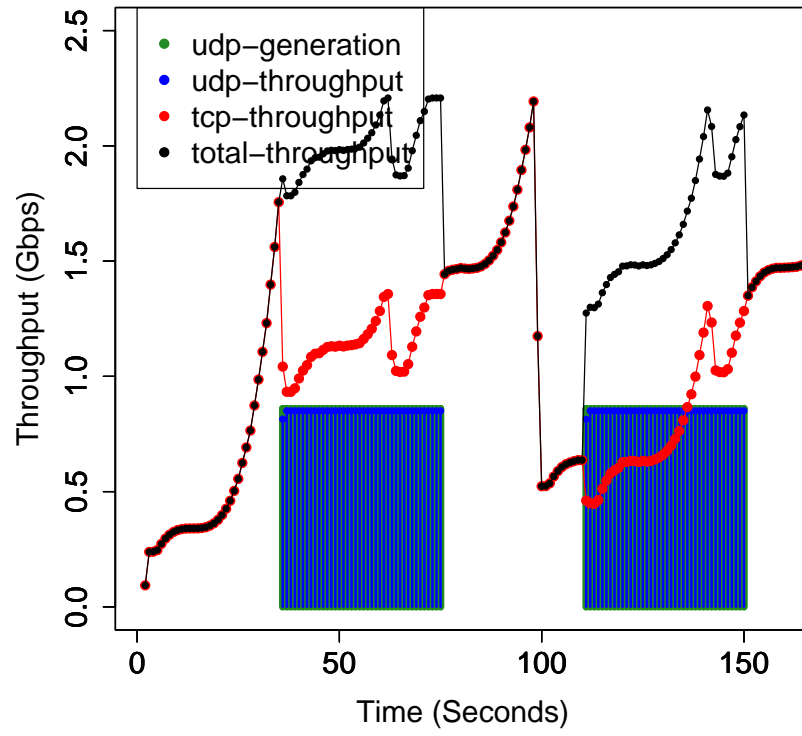
(a) Pulse Height = 40% of XG-PON capacity
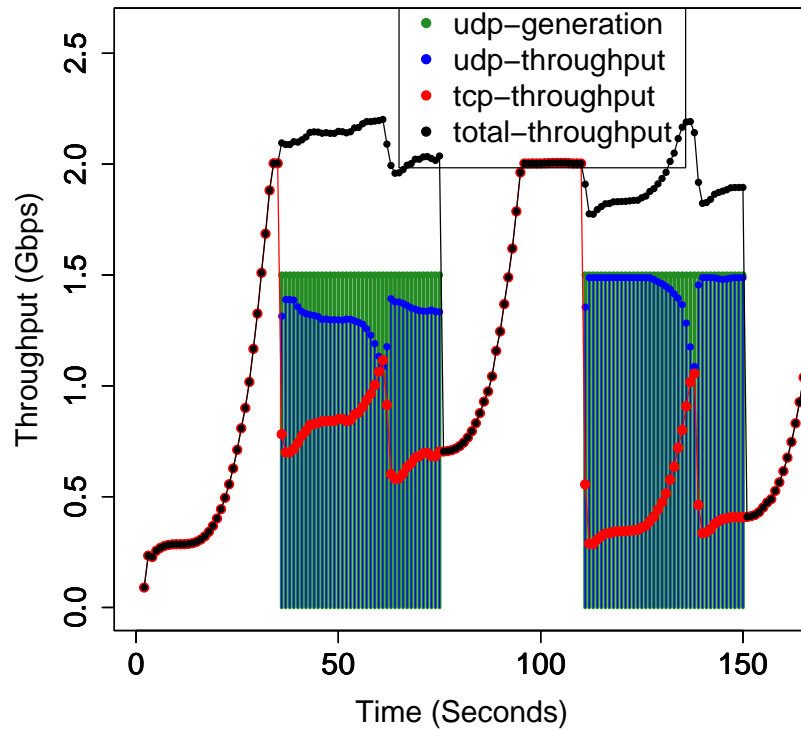
(b) Pulse Height = 70% of XG-PON capacity

Figure 5.12: Responsiveness of a CUBIC TCP flow to a UDP pulse, starting at 40s, pulse-width = 40s and pulse-gap = 40s
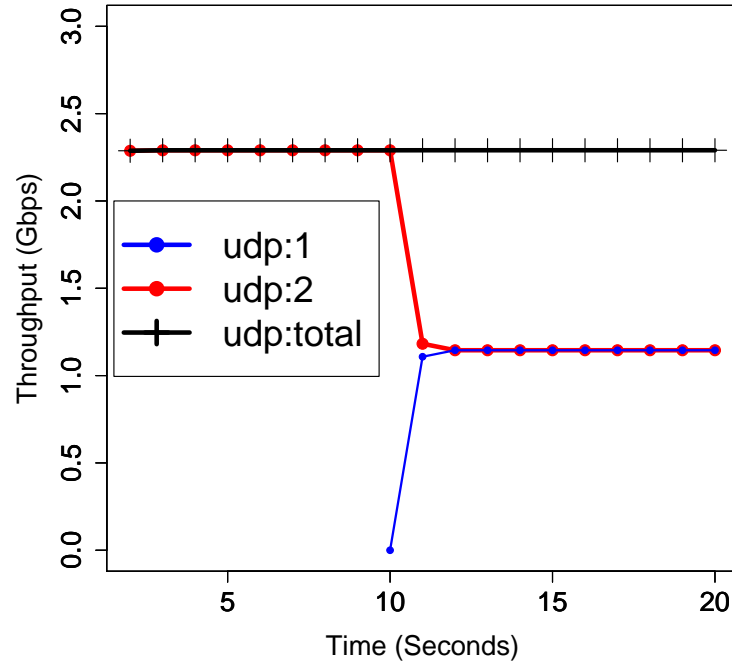
Figure 5.13: Convergence of 2 UDP flows, with the new one introduced at 10.1s

capacity, resulting in UDP pulses receiving less than the requested (70% of) upstream capacity. Once again, as the UDP pulses are removed, the TCP flows behave the same way as in 40%-pulse scenarios.

The results in this section validates the unpredictable nature of the response from an existing TCP flow towards a newly introduced UDP pulse due to: 1) the differences in the order of operational frequencies for the Round-Robin DBA in XG-PON (125-us) and the congestion control algorithm in TCP (few ms of RTT) and 2) the non-synchronised (feedback-less) operation of the DBA in XG-PON and the TCP congestion control algorithm at the client.

5.3.4 Convergence

This section presents the evaluation for the convergence of TCP and UDP flows under two different scenarios.

5.3.4.1 Scenario 1: Convergence of a new flow with an existing flow

As in Figure 5.13, when a new UDP flow is introduced at 10.1 seconds into the simulation of existing UDP flow, a very fast convergence of the two flows is seen due to
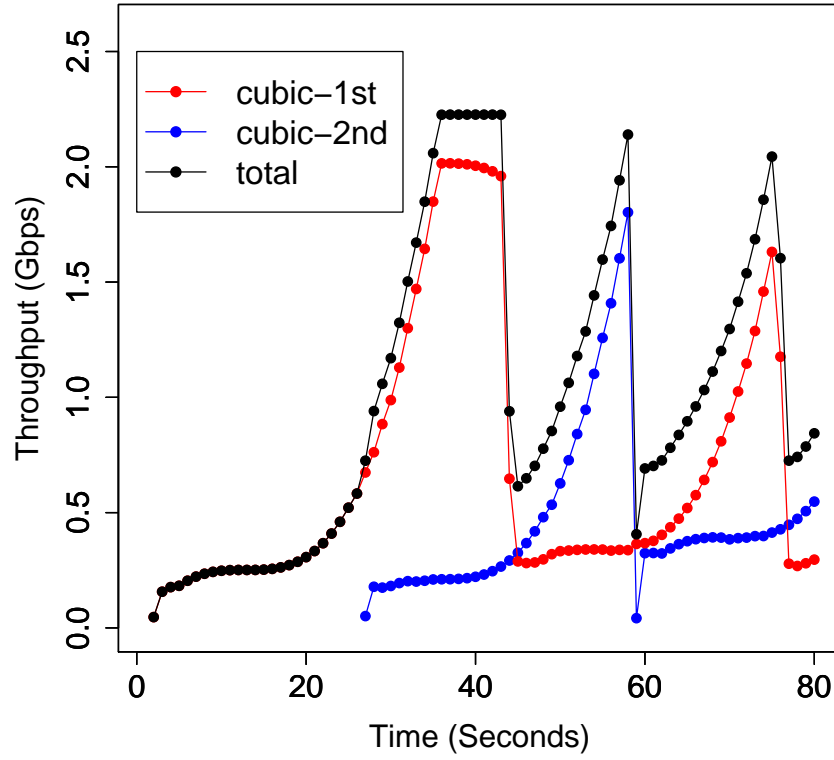
(a) 2nd Flow after 25s
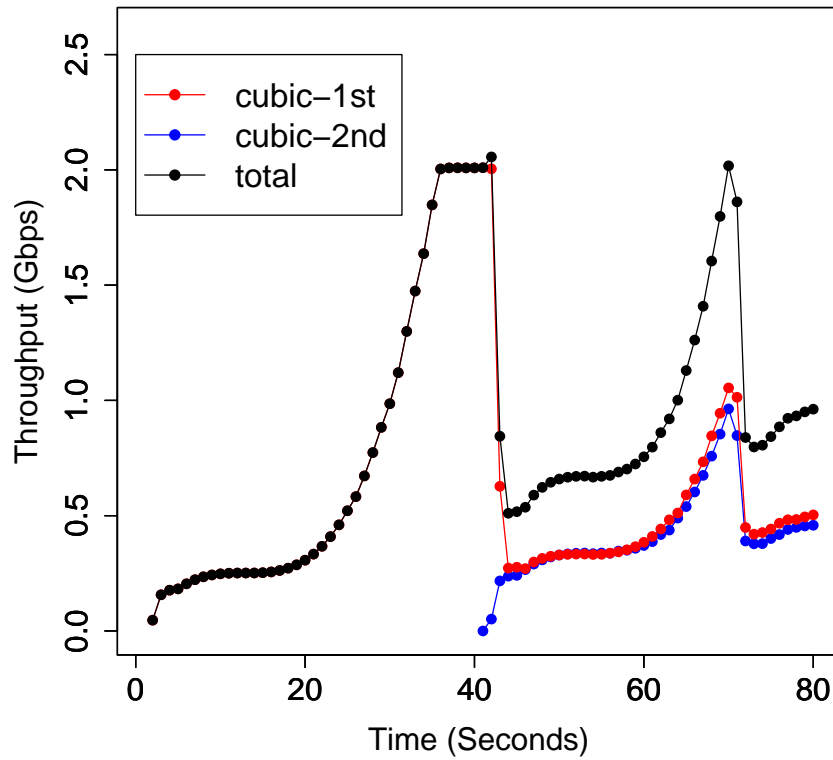
(b) 2nd Flow after 40s

Figure 5.14: Responsiveness of a CUBIC TCP flow to a new CUBIC TCP flow

the influence of the unbiased Round-Robin RR-DBA. The total throughput remains at 2.29Gbps throughout the entire duration of the simulation as expected.

Figure 5.14 presents a similar condition for two CUBIC TCP flows for an RTT of 62ms. Here, the new TCP flow (cubic-2nd) is introduced after 25 seconds into the simulation (Figure 5.14a) of the existing TCP flow (cubic-1st), with the 1st flow in its $exploratory - state$ [3] where the $CWND$ increases exponentially in CUBIC TCP. After the 40s mark, when the total throughput is throttled by the upstream capacity of XG-PON, the cubic-1st flow resets to its $steady - state$ [4] while the cubic-2nd flow continues in its $exploratory - state$, without any interference from the former, until the effective capacity of XG-PON throttles the total throughput (near 60s). After the second congestion, the cubic-1st, at its $exploratory - state$, continue to increase its $CWND$, without any interference from cubic-2nd which is first in its *Slow Start* phase and then in $steady - state$. Maximum throughput achieved by either of the TCP flows, when in the $exploratory - state$ depends on the unused capacity of XG-PON in the upstream.

To test the dependence of these observations on the relative time in introducing the new TCP flow, a second scenario is simulated, with the cubic-2nd introduced at 40 seconds into the cubic-1st in XG-PON and the results presented in Figure 5.14b. Here, cubic-1st immediately resets its $CWND$ followed by a $steady - state$. As cubic-2nd is also in its $steady - state$ after the resetting of $CWND$ of cubic-1st (around 45s), both TCP flows experience a similar increment of their $CWND$, resulting in fair throughput distribution between them in the upstream.

The above results again indicate an unpredictable behaviour for the convergence of two TCP flows when served by the unbiased Round-Robin DBA, in XG-PON upstream.

5.3.4.2   Scenario 2: Convergence of two new flows with existing 14 flows

Convergence of multiple flows is evaluated here for understanding the behavioural performance of the CUBIC TCP flows, when using small $CWND$ fluctuations. Figure 5.15 shows the convergence of 14 multiple CUBIC TCP flows (cubic:1-14) with 2 newly introduced CUBIC TCP flows (cubic:15 and cubic:16) of similar nature; convergence behaviour of UDP flows in Figure 5.16 are presented to indicate the quantify the relative behaviour of TCP flows in similar simulation environment. For both TCP and UDP, per-second-collected and per-10-millisecond-collected datarate plots are presented to understand the relative convergence behaviour and the relative difference in the time taken to converge. As the exact time of introduction of new flows depend on the complexity of simulation environment when datarates are measured per 10-ms, a black arrow indicates the exact point at which the new flows are introduced.

---

[3]When CUBIC TCP is in the convex region of the CUBIC curve in Figure 2.9 in Chapter 2
[4]When CUBIC TCP is in the concave region of the CUBIC curve in Figure 2.9 in Chapter 2
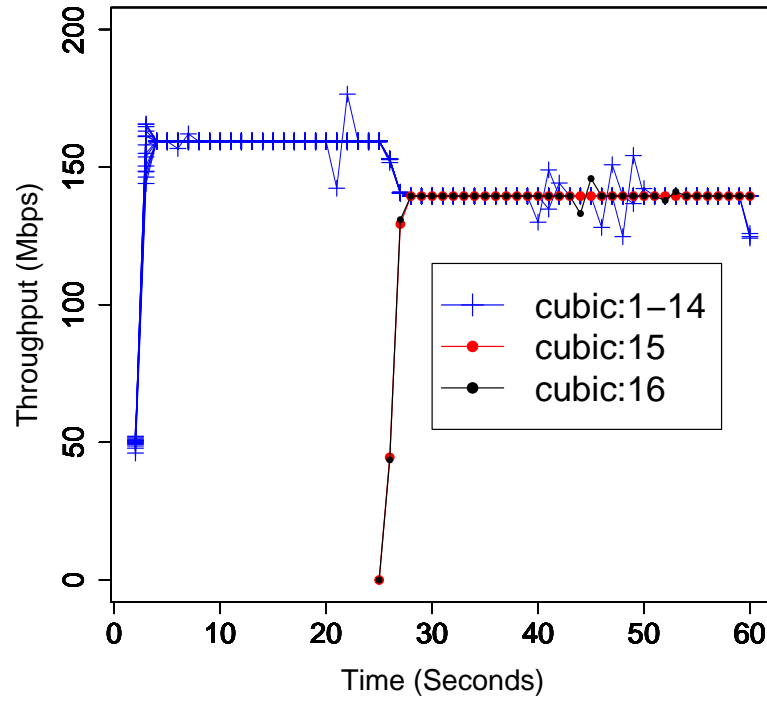
(a) Per-second datarate collection
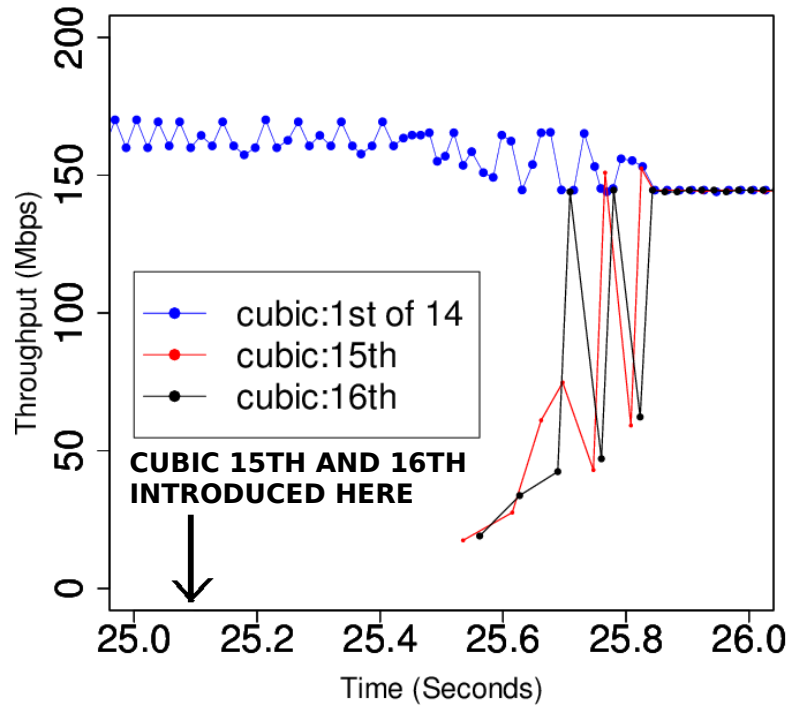
(b) Per 10-millisecond datarate collection

Figure 5.15: Convergence of 14 existing and 2 new TCP flows
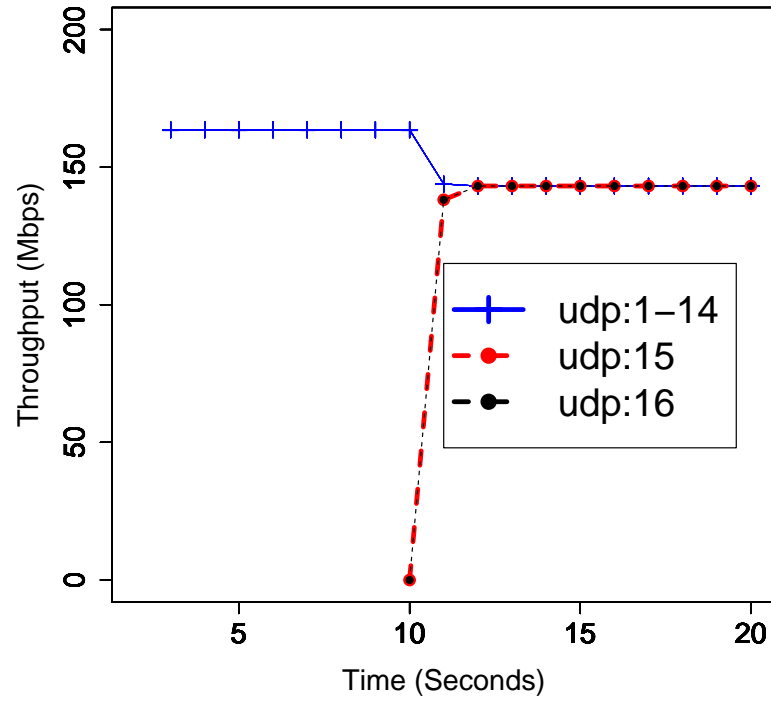
(a) Per-second datarate collection
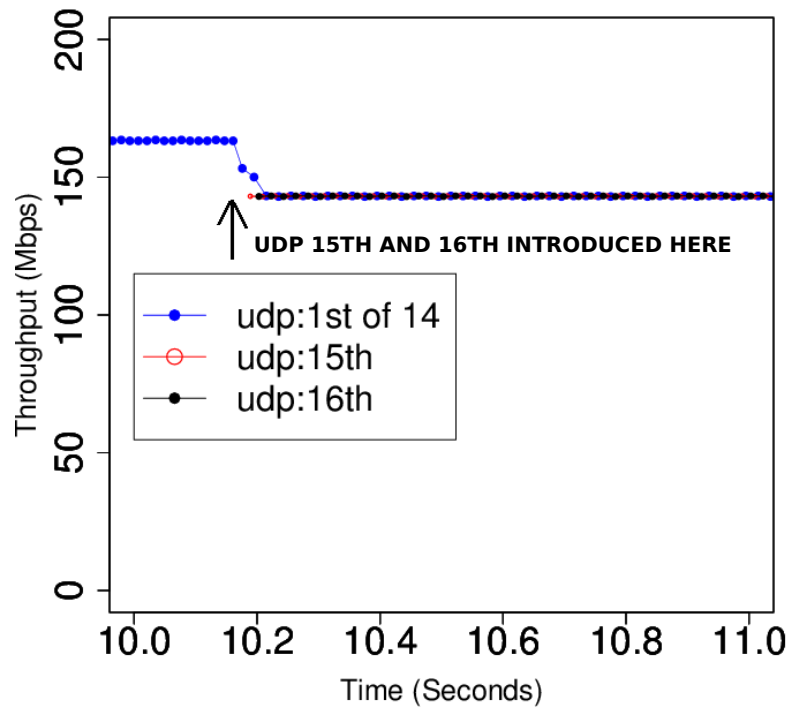
(b) Per 10-millisecond datarate collection

Figure 5.16: Convergence of 14 existing and 2 new UDP flows

As in Figure 5.15a, when two new TCP (CUBIC, RTT = 62ms) flows are introduced, a reasonably acceptable convergence of 16 TCP flows is seen just as for the 16 UDP flows as in Figure 5.16a. A faster and fairer convergence of all of the TCP flow(s) is observed, in the presence of large number of TCP flows, as opposed to only 2 flows in XG-PON upstream (seen in the previous convergence scenario). This is because as the number of TCP flows are larger in the XG-PON upstream, congestion algorithm in each TCP flow experiences throughput congestion earlier in its $CWND$ growth function, resulting in less variation of $CWND$ and prolonged $steady-state$ in the congestion algorithm. Since packet losses in $steady-state$ do not trigger *Slow Start* in a high-speed TCP congestion control algorithm such as CUBIC TCP, convergence is easily predictable when large number of TCP flows are introduced in XG-PON upstream, validating a predictable behaviour for the convergence of multiple TCP flows across the fixed effective bandwidth of XG-PON upstream.

Between the TCP and UDP flows, however, the two new TCP flows took more than 700ms (Figure 5.15b) to converge with the existing 14 flows, as opposed to under 100ms (Figure 5.16b) for the UDP flows.

Overall, the results of convergence indicate that: 1) though the Round-Robin DBA is not able to provide fair treatment for TCP flows with very high fluctuations in the $CWND$, causing unpredictable reactive measures at the congestion control of the high-speed TCP variant CUBIC, the DBA is effective for TCP flows when the fluctuations are small. 2) regardless of the unbiased nature of DBA, a simple convergence behaviour of 16 flows take significantly long durations for the TCP flows when compared with the UDP flows, indicating a larger impact on the convergence of several more TCP flows under more complex scenarios.

5.4   Chapter Summary and Conclusions

This chapter provided a detailed analysis of the performance of TCP flows in the context of downstream and upstream data transmission across a realistically deployed XG-PON. Evaluations are performed with different congestion control algorithms (NewReno, CUBIC and H-TCP) and a range of RTT values where necessary to investigate the combined impact of large BDP nature of XG-PON and the Round Robin DBA mechanism. This chapter investigates the TCP performances using only a simple (Round Robin) DBA in XG-PON, to distinguish the individual impact of large BDP and the DBA on the performance of TCP traffic across XG-PON in the downstream and upstream directions. The evaluations are based on four metrics, namely efficiency, fairness, responsiveness and convergence of the TCP flows. The results focused specifically in the upstream direction, since a DBA is required only in the upstream direction in an XG-PON standard. The summary of the evaluations in this chapter are as follows:

- A single TCP flow in XG-PON is able to achieve, on its own, a larger average throughput, close to the effective bandwidth of XG-PON only at very small RTT (or BDP) values in the downstream and upstream directions. However, the single TCP flow is not able to achieve such high throughput at higher RTT values, even when using high-speed TCP variants like CUBIC and H-TCP, be it in the downstream or upstream direction. A single (TCP) flow in XG-PON upstream is not under the effective influence of TDM (downstream) or the RR DBA (upstream); hence the resulting behaviour is purely due the large BDP nature of XG-PON in either direction. That is, a larger end-to-end RTT results in higher distance between (1) the two TCP epochs and (2) the highest and lowest achievable congestion window values for the TCP flow, resulting low average throughput as the end-to-end RTT increases.

- Similarly (due to the large BDP nature of XG-PON), when a new UDP pulse or TCP flow was introduced to an existing TCP flow in the upstream, a uniform (or predictable) response from the combined responses of the Round-Robin DBA and the TCP flows was not observed for both the existing TCP flow and the UDP pulse. The congestion window growth of and the utilisation of XG-PON by the newly introduced TCP flow depended heavily on the congestion window growth state of the existing TCP flow, indicating that an XG-PON is able to heavily influence the behaviour of the newly introduced TCP and UDP flow behaviours, individually, due to the large-BDP nature of the optical network.

- However, when a larger number of TCP flows were introduced, a higher total throughput was observed in the upstream, as the fluctuations between the values of congestion window of each TCP flow is very small, compared to a single-flow scenario. The results showed a higher bandwidth utilisation by all the TCP flows in the upstream, indicating that the large BDP nature of XG-PON is neutralised when a large number of TCP flows share the total upstream bandwidth of the XG-PON. It is also notable that the RR DBA, which uses a TDMA-based scheduling, not only facilitates a fair sharing (as validated by Jain's Fairness Index) but also refrain from causing throughput degradation, hence an efficient and fair throughput utilisation behaviour in XG-PON upstream for a large number of TCP flows.

- Similarly when larger number of (eg: 14) TCP flows were tested for convergence with 2 more TCP flows in the upstream, fairer throughput and predictable (as in UDP) convergence was observed for the TCP flows, due to having minimal impact from the large-BDP nature of XG-PON on the small congestion window fluctuation of each existing or new TCP flow. Due to the influence of the DBA on the scheduling delay experienced by the delay-sensitive TCP flows, the convergence time of TCP flows was in the range of several hundreds of milliseconds as opposed to few 10s of milliseconds in the corresponding UDP scenario,

Overall, these results indicated an unpredictable response nature of TCP flows across an XG-PON network, especially when a larger range of congestion window fluctuations are involved in the TCP epochs, during simplified simulation environments. However, when the fluctuations are small, the influence of XG-PON as a large-BDP network is minimal on the throughput performance of (multiple) TCP; simultaneously-present large number of TCP flows in the upstream, are also influenced for increase in end-to-end delay and therefore reduction in network utilisation (throughput) due to the scheduling mechanism inherent in XG-PON.

The next chapter will present the design, implementation and evaluation of two DBAs, namely EBU and XGIANT that provide Quality of Service in a stand-alone XG-PON network when using UDP-based upstream traffic, so that efficient QoS-aware DBA mechanisms can be developed for realistic upstream scenarios of serving multiple classes of applications in an XG-PON-based LTE backhaul.

Chapter 6

XGIANT: A new DBA mechanism for providing Quality of Service in a stand-alone XG-PON

This chapter presents the XG-PON-standard-compliant DBA, XGIANT, which is designed by improving the simple QoS framework of GIANT [59] DBA. GIANT (introduced in Chapter 2) was designed to provide QoS-aware scheduling for the GPON standard.

Using the XG-PON simulation module presented in Chapter 4, this chapter presents, in detail, the design, implementation and evaluation of a simple and efficient QoS-based DBA (XGIANT) for the stand-alone XG-PON. To design a standard-compliant and QoS-aware DBA for XG-PON, first this chapter analyses the QoS frameworks of two (X)GPON standard compliant DBAs, namely the GIANT for GPON and EBU for XG-PON; EBU is an efficient but complex improvement over the benchmarked GIANT DBA. The XGIANT DBA, which improves upon the DBA framework in GIANT and fine-tunes the values for key parameters in GIANT to design a standard-compliant and QoS-aware DBA for XG-PON is then presented. XGIANT was developed from the simple DBA framework of GIANT, as opposed to from the complex DBA framework of EBU, to facilitate a potential low complexity implementation of the DBA in a real XG-PON network. XGIANT . Finally, this chapter presents the implementation of XGIANT in the XG-PON module presented in Chapter 4, along with a validated implementation of EBU in the same module to provide comparative evaluations on the performances of both DBAs in terms of mean-queuing delay and throughput. UDP-based application traffic is used to validate the suitability of the QoS policies of XGIANT and EBU DBAs in XG-PON. The evaluation results presented in this chapter indicate that XGIANT is able to provide prioritised treatment with regard to mean queuing-delay and throughput for 3 different T-CONT types (T2, T3 and T4) when the XG-PON upstream is under-loaded, fully-loaded and over-loaded and ensure better mean queuing-delay performance than EBU for a range of upstream loading.

Table 6.1: Summary of GIANT DBA mechanism

T-CONT Type	Bandwidth Type	$GrantSize_{k,i}$ allocation mechanism	Frequency of $GrantSize_{k,i}$ allocation in one allocation cycle
T1	Fixed only	Fixed and constant $GrantSize$	Once, based on $MDR_{1,i}$
T2	Assured only	Polling and reservation method	Once, based on $MDR_{2,i}$
T3	Assured, Non-assured	Polling and reservation method	Once, based on $GDR_{3,i}$ and once based on ($MDR_{3,i}$-$GDR_{3,i}$)
T4	Best Effort only	Polling and reservation method	Once, based on $MDR_{4,i}$
	An allocation cycle = one round of $GrantSize$ allocation to all the $AllocIDs$ in the XG-PON network, up to $AB_{k,i}$.		

The chapter is organised as follows: section 6.1 presents a technical overview and analysis of GIANT and EBU, in terms of designing a simple and efficient DBA for XG-PON; section 6.2 presents the simulation environment, including the network architecture, evaluation metrics and simulation methodology, which is first used to refine the QoS policies of GIANT in section 6.3 and then justify the choice of values in XGIANT for the key parameters of GIANT in section 6.4; a detailed description of the implementation of XGIANT is presented in section 6.5; results and discussion for the performance of XGIANT and EBU are presented in section 6.6, which is followed by the chapter conclusion in section 6.7.

6.1 Related Work

This section presents a technical overview of both GIANT[59] and EBU[35] DBAs from the literature. GIANT DBA algorithm was both the first standard-compliant DBA mechanism and the first physically-implemented DBA algorithm for GPON. EBU DBA is an improved version of GIANT, proposed (with evaluations) for the XG-PON standard. This section discusses the concepts of GIANT and EBU, both of which are built on the same algorithmic principles, to provide differentiated QoS for four classes of traffic in (X)GPON. The details given in this section form the basis for the design and evaluation of XGIANT DBA, which is the core contribution of this chapter.

6.1.1 Technical overview of GIANT and EBU DBAs

GIANT maps four traffic classes to the four T-CONT types in GPON (T1 - T4) as in Table 6.1, with each T-CONT type subjected to a distinct bandwidth allocation strategy.

In order to explain the bandwidth allocation strategy of GIANT, let us assume that GPON serves N ONUs, with each ONU serving all four T-CONT types, to allocate $GrantSize_{k,i}$ to each $AllocID_{k,i}$ based on a Maximum Data Rate (MDR_k), where k ε {T1, T2, T3, T4} and i ε N. Since T1 traffic has a fixed $GrantSize_{1,i}$ allocation and the fixed bandwidth portion in (X)GPON is heavily-dependent on the requirements of

the service providers, GIANT does not specify the provision of MDR_1. $AllocID_{k,i}$ belonging to the other T-CONT types ($k \ \varepsilon$ {T2, T3, T4}) are provisioned $GrantSize_{k,i}$, with an upper bound value of $AB_{k,i}$ such that:

$$AB_{k,i} = \frac{(SI_k * 125\mu s) * MDR_k}{N} \tag{6.1}$$

and

$$MDR_k = 0.67 * C_{GPON} \tag{6.2}$$

where C_{GPON} is the effective upstream capacity of G-PON; service interval, SI_k is the frequency at which the DBAs allocate $MDR_{k,i}$ to each $AllocID_{k,i}$. SI_k also determines the minimum (when the upstream is not congested) frequency at which an $AllocID_{k,i}$ was provisioned with the $GrantSize_{k,i}$. Since T3 serves both assured and non-assured traffic types, the Guaranteed Datarate for T3 (GDR_3) in GIANT is equal to 50% of MDR_3, to provide a guaranteed bandwidth allocation to the assured portion of T3. In equation 6.1, SI_k represents the frequency at which each $AllockID_{k,i}$ is given upstream $GrantSize_{k,i}$ in GIANT. A down counter is used in GIANT, to start with the value of SI_k, reduce by 1 at the end of every upstream frame, and reset to SI_k when a value of 1 is reached.

Different SI_k values for T2 (SI_2) and T4 (SI_4) ensure T2 and T4 are given the highest and lowest priority in the entire GPON network consisting of multiple T2, T3 and T4 T-CONTs. To provide an intermediate priority for T3, GIANT defined two sets of SI_3, for the assured/guaranteed datarate (GDR_3 with SI_{3G}) and non-assured datarate (MDR_3 - GDR_3 with SI_{3M}), where the SI_{3G} is equal to SI_2 and SI_{3M} to SI_4. However, due to the reliance of $GrantSize_{k,i}$ allocation only on the expiry of down counter (when the down counter reaches a value of 1), GIANT proved to be inefficient in $GrantSize$ allocation in different upstream loads across GPON even though GIANT's QoS framework was compliant with the GPON standards.

Han et al.[35] proposed the EBU DBA for XG-PON, using the same framework of GIANT as in Table 6.1 and eliminating the over-reliance of GIANT on the down counter, to result in an XG-PON-standard-compliant DBA, which efficiently utilised the upstream bandwidth of XG-PON among the $AllocID$s of T2, T3, and T4. Since EBU is designed for XG-PON, the equation 6.2 becomes:

$$MDR_k = 0.67 * C_{XG-PON} \tag{6.3}$$

where C_{XG-PON} is the effective upstream capacity of XG-PON (= 2.3Gbps as per evaluations in Chapter 4). EBU ensured that the $GrantSize_{k,i}$ is allocated to an $AllocID_{k,i}$

when there is unutilised space in the upstream frame, within the aggregated limits of $AB_{k,i}$ $(= \Sigma_{i=1}^{N} AB_{k,i})$, at the expense of a complex DBA, due to the calculations performed within EBU to keep track and make use of the differences between each $GrantSize$ and the original queue occupancy report by the corresponding $AllocID$.

EBU also introduced the inter-ONU fairness mechanism which, in a round-robin manner, rotated the ONU whose $AllocID$ is first given $GrantSize$ in each upstream frame; inter-ONU fairness ensured that the effect of congestion in the upstream (the non-allocation of $GrantSize$ to only the last few $AllocID$s) is spread across all the ONUs.

6.1.2 Discussion on the cons and pros of GIANT and EBU in the context of a stand-alone XG-PON

GIANT introduced a simple and effective DBA framework for differentiated QoS provision within GPON. However, GIANT lacked upstream bandwidth utilisation efficiency which is a crucial factor that should not be compromised in providing QoS in GPON. EBU, a GIANT-optimised DBA for XG-PON, focuses on smoothing the bursty nature of XG-PON's traffic, by introducing complex DBA calculations within the DBA, while retaining the impact of down counters by using large values for SI_k. EBU also ignored the evaluation of its mean queuing-delay performance for overloaded XG-PON, even though over-loading of XG-PON is a common condition that could occur in any realistic XG-PON deployment due to the amount of upstream traffic arriving in the T-CONT queues in XG-PON.

In summary, though EBU preserved a substantial amount of DBA principles and algorithmic architecture of GIANT, it used complex optimisations for achieving moderate improvement in mean-delay. This chapter proves that it is possible to achieve a better (than EBU) mean queuing-delay performance for a QoS-aware DBA in XG-PON by introducing simpler-than-EBU optimisations to GIANT, by means of improved QoS policies and thoroughly evaluated parametrisation (from GIANT).

The rest of the sections in this chapter present the details of the design, implementation and evaluation of the resulting GIANT-improved DBA, XGIANT. XGIANT is evaluated, using the XG-PON simulation module presented in chapter 4, to demonstrate the superiority of the queuing-delay performance of XGIANT against EBU, while maintaining priorities between T2, T3 and T4 T-CONT types, for a range of XG-PON upstream loading (under, fully and over -loaded upstream) in XG-PON. In order to facilitate potential implementation targets in a realistic XG-PON network, this chapter focuses in the design of a simple DBA mechanism that can achieve mean queueing delay performance similar to or better than EBU, when serving multiple classes of traffic.
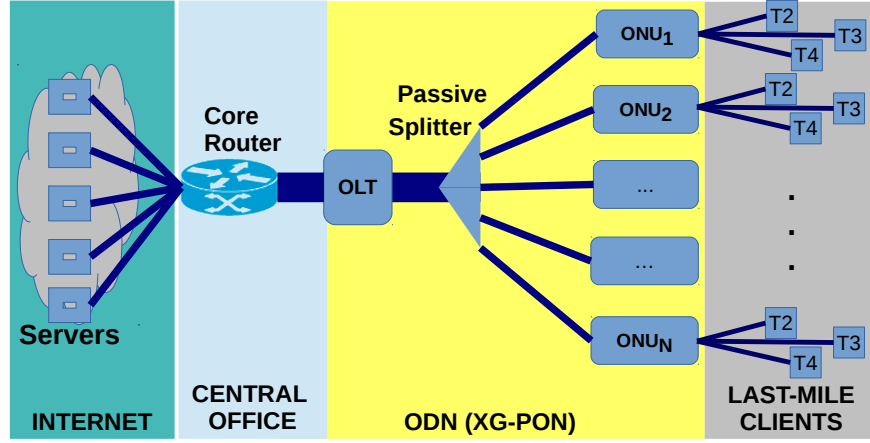
Figure 6.1: Network set-up used for simulations

6.2 Simulation Environment

This section describes the simulation environment used for: 1) the refinement of GI-ANT DBA to design a suitable and improved DBA in the form of XGIANT and 2) to evaluate the performance of XGIANT and EBU in similar networking conditions. The network architecture is presented first, followed by the metrics of evaluations; simulation methodology is presented in the end.

6.2.1 Network Architecture

The network topology in Figure 6.1 is designed to reflect a typical PON access network connected to the Internet, for estimating the mean queuing-delay and throughput of different classes (T-CONTs) of traffic arriving in each ONU. Each ONU is configured to have one flow for each of the 3 T-CONT (T2, T3 and T4) types; T1 traffic, and therefore $AllocID_1$, is not configured in any ONU, as T1 has a fixed allocation in XGIANT, as in GIANT and EBU; a one-way propagation delay of 400us is configured for all the ONU-OLT logical links, to represent a distance of 60km between the OLT and each ONU. Every link except for the OLT-ONU links are configured for very high physical datarates and sufficient MAC and PHY queues, such that XG-PON is the only bottleneck link in the transmission between the users in the last-mile and the servers in the Internet.

6.2.2 Evaluation Metrics

The following metrics are evaluated in the rest of the chapter in refining GIANT for XG-PON and comparing the resulting XGIANT DBA with the EBU DBA.

- Mean queuing-delay: This is a crucial metric of evaluation in the context of DBA design for XG-PON, as the delay and relative priority of the DBAs are demon-

strated through this metric. Mean queuing-delay is evaluated in the experiments of this chapter, only for the packets which are transmitted across the XG-PON in the upstream, after entering and waiting at the T-CONT queues: first the packets entering and exiting each T-CONT queue is accounted for the difference in time (queuing-delay) in the corresponding T-CONT queue; the average of the queuing-delay for all the packets across all the ONUs for each T-CONT will yield the final mean queuing-delay value for each T-CONT type in XG-PON. The total delay across the XG-PON for application traffic can be calculated as explained in section 2.2.4 of Chapter 2).

- Throughput: this metric represents the total amount of data (payload) transmitted across the XG-PON within the total simulation duration, for each of the 3 T-CONT types (T2, T3 and T4). In addition to reflecting average datarate per T-CONT type in XG-PON, the throughput metric also provides the validation regarding the capacity sharing assurances by the DBA towards each T-CONT type.

6.2.3   Simulation Methodology

As the main motivation behind the experiment is to produce a direct comparison of XGIANT and EBU DBAs, the simulation environment in this chapter uses a similar set-up[1] as in the original publication of EBU by Han et al. [35]: 16 ONUs are used in XG-PON, with each scenario being simulated for a duration of 20 seconds ($>10^6$ packets when upstream is fully-loaded); a T-CONT queue value of 15KB was used for each T-CONT type (T2, T3 or T4); packet size was fixed at 1472 Bytes[2] and each T-CONT flow was generated with a fixed inter-packet arrival rate, such that total upstream traffic ratio (= Total Upstream Traffic / Effective capacity of XG-PON in upstream) from all three T-CONT flows in all 16 ONUs is between 0.5 and 1.8.

6.3   Refinements to GIANT

This section presents the refinements XGIANT makes to GIANT, in order to improve the network conditions that may occur in the XG-PON upstream.

1. Utilisation of XG-PON upstream capacity

GIANT allocates bytes to any $AllocID_{k,i}$ only at the expiry of its down counter. When the down counter waits for more than one upstream frame ($SI_k > 1$), there will always

---

[1]In the subsequent chapters, XGIANT and EBU are evaluated for similar, yet different from the original simulation environment proposed by Han et al. to analyse the influence of the DBAs on the upstream traffic under different configurations (number of ONUs, buffer size in the ONUs, etc.) of XG-PON.

[2]maximum packet size at application layer to avoid segmentation of 1500 Bytes-long XGEM frame

be an allocation cycle when the down counter is merely updated (reduced by 1), without providing any $AB_{k,i}$, even though the $AllocID_{k,i}$ has requested upstream transmission slots. Though EBU dynamically re-assigns unused bandwidth reserved from one $AllocID_{k,i}$ to the other within a T-CONT type, the problem of under-utilised upstream frame may not be eliminated across all the T-CONT types as long as an allocation cycle spans across multiple upstream frames.

Decision: XGIANT calculates $AB_{k,i}$ as a percentage of (SI_k * upstream frame size), instead of a percentage of an upstream frame. This ensures that, even when the down counters of $AllocID$s are not expired, sufficient upstream transmission slots are provisioned by XGIANT to fully utilise the upstream capacity in the presence of the available upstream load.

2. $GrantSize$ **provision for lower-priority T-CONT types in an over-loaded XG-PON**

GIANT and EBU provision $GrantSize$ to $AllocID$ only when the allocation cycle has space for the $AllocID$ to transmit data an upstream frame. But, the down counter is updated only when the $AllocID$ is queried in an allocation cycle; hence, the lower priority $AllocID$s, which are not queried at over-loaded upstream conditions, merely update their down counters when they are eventually queried at a subsequent allocation cycle; the query, and the down-counter update for a single allocation cycle will only occur after several upstream frames. This will lead to higher queueing-delays for lower priority $AllocIDs$ when the upstream load increases, resulting in extended starvation of best effort traffic for multiple upstream frame cycles[3].

Decision: When any $AllocID$'s down counter expires, XGIANT allocates transmission slots to the particular $AllocID$ in the next possible upstream frame, without waiting for another down counter expiry. That is the $AllocID$s with expired down counters are remembered for their down counter state until an upstream transmission slot is guaranteed by the DBA.

6.4 Justification for the Choice of Values for Key GIANT Parameters

This section presents the refinements XGIANT makes, after performing extensive evaluations on two key parameters of GIANT (and EBU) and identifying avenues of improvements regarding parametrisation. In provisioning $GrantSize$ to different T-CONT types, both GIANT and EBU used parameters with minimal explanation/evaluation. However, this section clarifies that, by extensive evaluations, two key parameters in the DBAs can have an impact on the priority and queuing-delay performances of the T-CONT types, for different upstream loads in XG-PON.

---

[3]A framing cycle = allocation of $GrantSize$ by the DBA within a single upstream frame
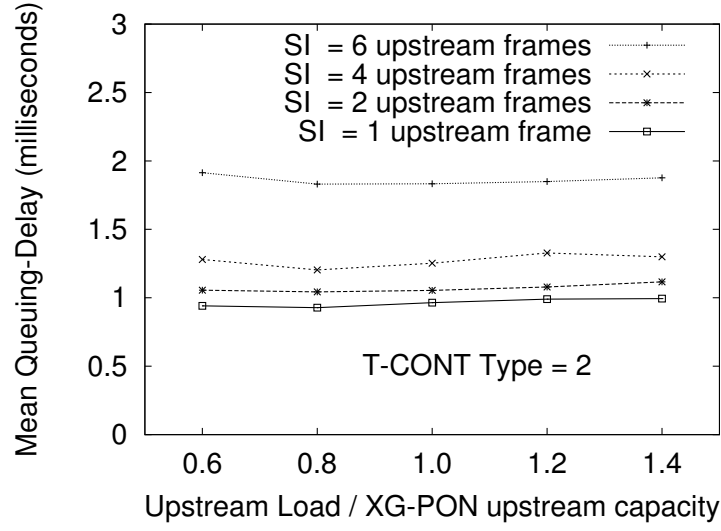
Figure 6.2: Mean queuing-delay in T2 for different values of service intervals (SI_2)
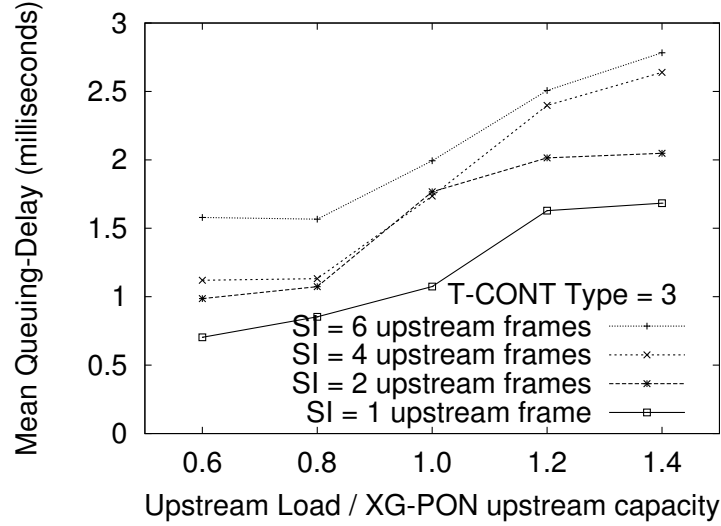
Figure 6.3: Mean queuing-delay in T3 for different values of service intervals (SI_3)

The two key parameters evaluated and presented here are SI_k and $GDR_3 : MDR_3$ ratio. These parameters are chosen for the extensive evaluation here due to the substantial impact the range of values from each of the parameters can have on the performance of the DBA in a stand-alone XG-PON network. The rest of the parameters, such as the ratio of MDR_k to C_{XG-PON} (as in equation 6.3) and the allocation policies will further be evaluated in the following chapter, with an objective of designing a QoS-aware DBA for the integrated network architecture of XG-PON and LTE.

The stand-alone XG-PON network architecture in Figure 6.1 is used for evaluating the impact of SI_k and $GDR_3 : MDR_3$ ratio on the performance of the partially refined GIANT DBA (which is an outcome of section 6.3).
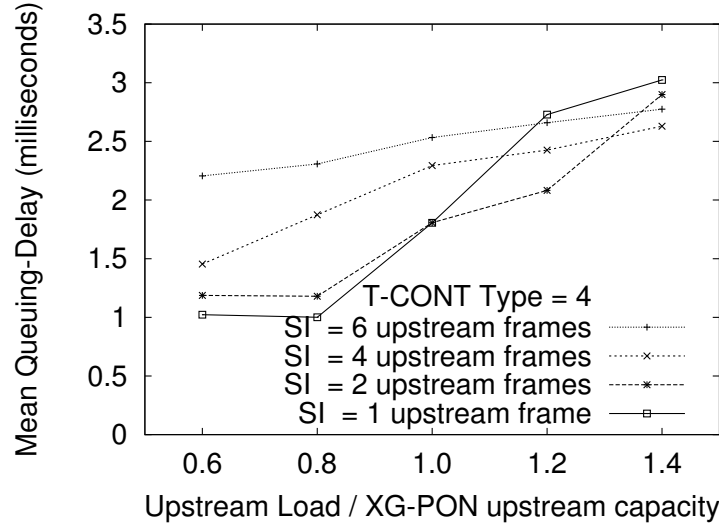
Figure 6.4: Mean queuing-delay in T4 for different values of service intervals (SI_4)

6.4.1 Parameter 1 - SI_k

In order to smooth the burstiness in the arrival of packets in the T-CONT queues, GIANT and EBU used 5 or 10 upstream frames for SI_k. Since the number of upstream frames a packet waits in the T-CONT queue can have a direct impact on the mean queueing-delay of the packet, a range of values (1,2,4 and 6 upstream frames) are chosen for SI_k to determine the impact of these values on the mean queuing-delay. For T3, however, the SI_{3G}:SI_{3M} ratio is kept at 1:2 to provide an isolated discussion on the impact on all the T-CONT types, than on the T3 T-CONT type only.

The results indicate that, for T2 ((Figure 6.2) and T3 (Figure 6.3), the mean queuing-delay is minimum when one upstream frame cycle is used for SI_2 and SI_{3G}; as SImax increases, so does the delay. For T4 (Figure 6.4), when upstream is under-loaded, delay is minimum when SI_4 is lowest (=1); however when upstream is over-loaded, delay increases quickly, as expected, for lower SI_4 values, at the expense of prioritising, first T2 and then T3, traffic.

Decision: SI_2 and SI_{3G} are fixed at 1 (upstream frame), to accommodate each allocation cycle of MDR_2 and GDR_3 within one upstream frame cycle to provide low and prioritised queuing-delay performances for T2 and assured portion of T3. SI_4 is fixed at 2 (upstream frames) to provide slowly increasing mean queuing-delay performance for T4 traffic when XG-PON is over-loaded.

6.4.2 Parameter 2 - Ratio of GDR_3:MDR_3

For the T3 traffic, GIANT and EBU used fixed values for the ratio between the assured (GIR in [59]) and the total (PIR in [59]) portions of the T3 bandwidth. The equivalent
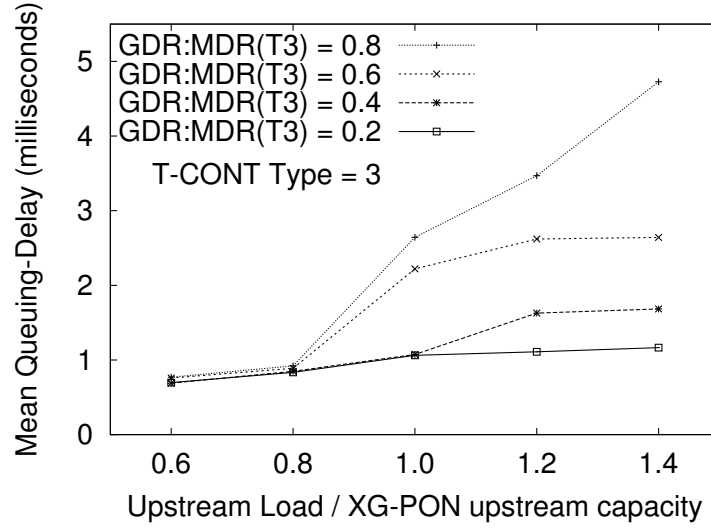
Figure 6.5: Mean queuing-delay in T3 for different $GDR_3 : MDR_3$ ratios of T3
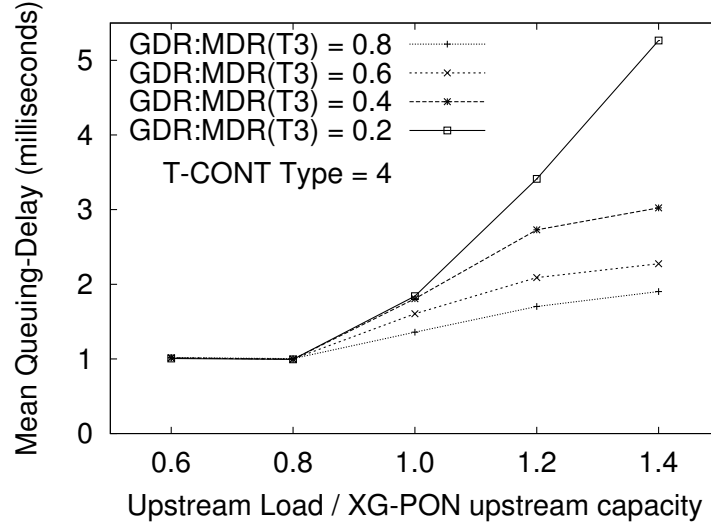
Figure 6.6: Mean queuing-delay in T4 for different $GDR_3 : MDR_3$ ratios of T3

GDR_3:MDR_3 ratio is varied here to evaluate its impact on the mean queuing-delay performance of the DBA.

As the GDR_3:MDR_3 ratio is increased, the mean queuing-delay for overall traffic in T3 increases (Figure 6.5) while the mean queuing-delay for T4 (Figure 6.6) decreases. For T3, since the MDR_3 is fixed for different GDR_3:MDR_3 ratios, increasing the ratio adds more traffic into the assured portion, hence higher mean queuing-delay which accounts for both the assured and non-assured portion of T3. For T4, when the T3 GDR_3:MDR_3 ratio is increased, less (MDR_3 - GDR_3) rate given for T3 every SI_3M, during which time, more T4 traffic is given transfer opportunities, showing less mean-delay for T4. The corresponding results for T2 (not included) showed no difference in mean queuing-delay as the varying ratio of GDR_3:MDR_3 has no impact on T2 traffic, which is served before T3 traffic.

Decision: In order to maintain the mean queuing-delay priority between T3 and T4, the GDR_3:MDR_3 ratio is fixed at 0.4. This ensured that T4 was always treated at a lower priority than T3 traffic at light-loaded, fully-loaded and overloaded upstream conditions.

The resulting DBA is an improved version of GIANT DBA and presented in this chapter, under the name of XGIANT DBA, to provide QoS treatment for different classes of traffic in the XG-PON upstream.

6.5 Implementation of XGIANT DBA

This section presents the implementation details of XGIANT DBA in XG-PON. Chapter 4 presented the pseudo-code and flow-chart for the abstracted bandwidth allocation function (*BW_Allocation_main()*) of the *XgponOltDbaEngine* class, in Algorithm 1 and Figure 4.3 respectively. XGIANT (compared to the Round-Robin DBA in XG-PON) has its implementations in two sub-functions of the *BW_Allocation_main()* function.

First, the *Calculate_GrantSize(AllocID$_{k,i}$)* sub-function is modified so that XGIANT can be implemented in OLT. Algorithm 2 presents the pseudo-code for $GrantSize_{k,i}$ calculation (line 4 of Algorithm 1 in Chapter 4) in XGIANT. Since XGIANT makes no refinements to T1, lines 2-4 follow the same procedure as in GIANT (or EBU). Parameter 1 optimisation of section 6.4.1 is reflected in lines 6 and 11 for MDR_2 and GDR_3 respectively (1 upstream frame cycle eliminate the need for down counter expiry validation) and in lines 16 and 26 for (MDR_3 - GDR_3) and MDR_4 respectively (2 upstream frame cycles). Parameter 2 optimisation in section 6.4.2 for T3 is reflected in lines 11 and 17. *assured_round* for T3 and T4 indicate the time during and immediately after the GDR_3 allocation, respectively, in XGIANT.

Then Check_All_AllocIDs_Served(*AllocID$_{k,i}$*) (line 8 of Algorithm 1 in Chapter 4) is implemented in XGIANT is implemented to accommodate the marginal cases of moving from the last *AllocID* of higher priority T-CONT type to first *AllocID* in the lower priority T-CONT type. Order of *AllocIDs* in each T-CONT type is indexed in a fixed round-robin manner throughout the entire duration of XG-PON operation.

6.6 Results and discussion

This section presents the evaluation of XGIANT DBA in XG-PON, with comparisons to a validated (against the performance of EBU presented in [35]) implementation of EBU DBA for the XG-PON module presented in Chapter 4.

Algorithm 2 : Calculate_GrantSize ($AllocID_{k,i}$) in XGIANT

Here, XGIANT DBA calculates the actual $GrantSize_{k,i}$ to be given to each served $AllocID_{k,i}$, as a function of the requested queue occupancy report (request_size ($AllocID_{k,i}$)) and $AB_{k,i}$. Specifically, XGIANT follows GIANT DBA to provision T1, T2, assured portion of T3, polling of T4, non-assured portion of T3 and T4, in the given order **The following are initialised at the beginning of XG-PON operation:** for k= 1:4 and for every i ϵ N, $AB_{k,i} = \frac{(SI_k*125\mu s)*0.67*C_{XG-PON}}{N}$, as per Equation 6.1, 6.3.

1: TYPE = Get_TCONT_Type ($AllocID_{k,i}$)

2: **if** (TYPE == 1) **then**   ▷ T1 AllocID gets grant
3:     grant_size = $AB_{1,i}$
4: **end if**

5: **if** (TYPE == 2) **then**   ▷ Then T2 AllocID
6:     grant_size = min { request_size($AllocID_{2,i}$), $AB_{2,i}$ }
7:     Shape_Grant_For_Standard(grant_size, $AllocID_{2,i}$)
8: **end if**

9: **if** (TYPE == 3) **then**   ▷ For each T3 AllocID
10:     **if** (assured_round == true) **then**   ▷ Assured portion is given every round
11:         grant_size = min { request_size($AllocID_{3,i}$), 0.5 * 40% of $AB_{3,i}$ }
12:         **if** (grant_size > 0) **then**   ▷ DBRu is given in assured_round, if only it
    $AllocID$ data to transmit
13:             Shape_Grant_For_XGPON_Standard(grant_size, $AllocID_{3,i}$)
14:         **end if**
15:     **else** ▷ non-assured portion given, every 2nd upstream frame (= $SI_{3M,i}$ expiry)
16:         **if** (non-assured_down_counter($AllocID_{3,i}$) ==   expired) **then**
17:             grant_size = min { request_size($AllocID_{3,i}$), 60% of $AB_{3,i}$ }
18:             Shape_Grant_For_XGPON_Standard(grant_size, $AllocID_{3,i}$)
19:             reset non-assured_down_counter($AllocID_{3,i}$)
20:         **end if**
21:     **end if**
22: **end if**

23: **if** (TYPE == 4) **then**
24:     **if** (assured_round == true) **then**
25:         grant_size = 1   ▷ T4 is polled in assured round
26:     **else**   ▷ T4 AllocID is given grant every 2nd upstream frame (= $SI_{4,i}$ expiry)
27:         **if** (down_counter($AllocID_{4,i}$) == expired) **then**
28:             grant_size = min { request_size($AllocID_{4,i}$), $AB_{4,i}$ }
29:             Shape_Grant_For_XGPON_Standard(grant_size, $AllocID_{4,i}$)
30:             reset down_counter($AllocID_{4,i}$);
31:         **end if**
32:     **end if**
33: **end if**
34: **return** grant_size
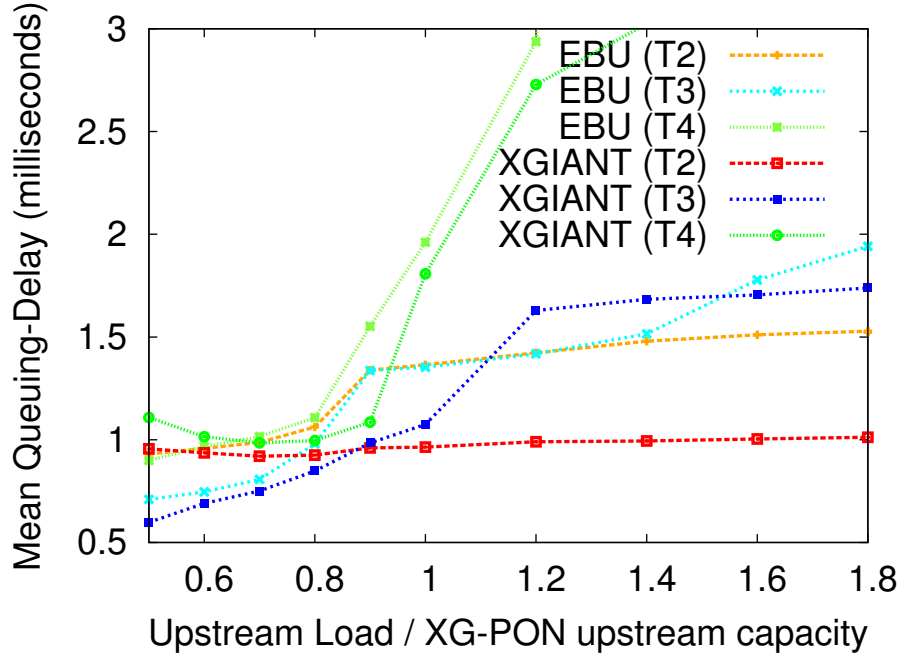
Figure 6.7: Mean Queuing-Delay of T2, T3 & T4 at different upstream load in XG-PON

The results present the evaluations for the metrics mean queuing-delay and throughput, which are explained in section 6.2.2. The comparison behaviour between XGIANT and EBU are only given for the mean queueing-delay which is sufficient to establish the superiority of XGIANT against EBU.

6.6.1 Mean queuing-delay

Figure 6.7 shows the mean queuing-delay performance of XGIANT and EBU for T2, T3, and T4 T-CONT types, across the entire XG-PON network. To provide a fair comparison, the same network set-up and simulation methodology (as briefed in sections 6.2.1 and 6.2.2 respectively) are used for the performance evaluation of each DBA. EBU was implemented with the same algorithm and the inter-ONU fairness policy in its original proposal [35] and simulated using its original values of $AB_{k,i}$ (ABmin$_{T2}$ = ABmin$_{T3}$ = ABsur$_{T3}$ = 7812 Bytes, AB$_{T4}$ = 15624 Bytes), SI_k (SImax$_{T2}$ = 5, SImax$_{T3}$ = SImin$_{T3}$ = SI$_{T4}$ = 10) and GIR:PIR ratio of T3 (=0.5). XGIANT, after refined from GIANT, contained different values for $AB_{k,i}$ ($AB_{2,i}$ = 1564 Bytes, 0.5 * 40% of $AB_{3,i}$ = 624 Bytes, 60% of $AB_{3,i}$ = 1876 Bytes, $AB_{4,i}$ = 3128 Bytes), SI_k (SI_2 = SI_{3G} = 1; SI_{3M} = SI_4 = 2) and GDR_3:PDR_3 ratio of T3 (= 0.4). The results for EBU in Figure 6.7 closely match the results in [35] when XG-PON is loaded for upstream load ratios ($=\frac{Upstream\ Load\ in\ XG-PON}{C_{XG-PON}}$) from 0.5 to 1.0; however, a notable increment of mean queuing-delay can be seen for upstream load ratios between 0.7 and 0.9, for all three T-CONT types. Since the discussions in this section use the mean queuing-delay that is averaged across all the ONUs in a single XG-PON network, XGIANT does not implement

a fairness policy as in EBU.

As the upstream load ratio increases from 0.5 to 1.8, the mean queuing-delay of T2 traffic remains constant and low (1ms) for XGIANT, while in EBU mean delay for T2 shows a sudden hike at 0.9 load ratio (due to the T-CONT queue being utilised fully) and keeps increasing at a slower gradient as load ratio further increases. The mean queuing-delay for T4 in XGIANT also show a significant improvement over that of EBU beyond 0.8 load ratio. This is because, while XGIANT makes use of early expiry of SI, even when the upstream load ratio is greater than 0.9, the impact of the sum of unutilised $AB_{4,i}$ in EBU is minimal under similar scenarios.

For T3, both XGIANT and EBU achieve lower mean-delays, compared to the higher priority T2, when load ratio is less than 1.0; this is because XGIANT gives $GrantSize$ to T3 at a higher frequency (more number of $GrantSize$ allocations) than for T2 within an allocation cycle; EBU, however, provides more $GrantSize$ (in terms of total Bytes) to T3 than to T2, at equal frequencies. This condition can also be considered a violation of strict prioritisation between T2, T3 and T4 as required in XG-PON standard, though the elimination of the condition relies merely in the choice of values for $AB_{2,i}$ and $AB_{3,i}$, which may need to have different set of values when load ratio is under 1.0. However, when load ratio is increased beyond 0.9, T3 of XGIANT and EBU takes its priority between T2 and T4 as intended, with XGIANT achieving lower mean-delay overall.

6.6.2 Throughput

Figure 6.8 shows the breakdown of the throughput in the XG-PON upstream as provisioned by XGIANT for the three T-CONT types (when each T-CONT buffer is loaded with equal amount traffic arrival, which has a uniform distribution for its inter-packet arrival). As desired, at under-loaded conditions (upstream load ratio < 1) all T-CONT types achieve the same throughput, while in fully and over-loaded (upstream load ratio $\leq$ 1) conditions, T4 throughput is quickly sacrificed in favour of T2, while T3 declines more gracefully because of the assured (40%) portion in T3. XGIANT is also able to utilise the upstream capacity up to 2.3 Gbps at fully-loaded and over-loaded conditions, proving the efficient network utilisation capability of the DBA.

6.7 Conclusion

This chapter presented the XGIANT DBA for the stand-alone XG-PON network. XGIANT is an improved version of the benchmarked GIANT DBA, which was the first GPON standard-compatible and physically implemented DBA to provide QoS-based treatment for three effective types of T-CONTs. In presenting the details of XGIANT implemented
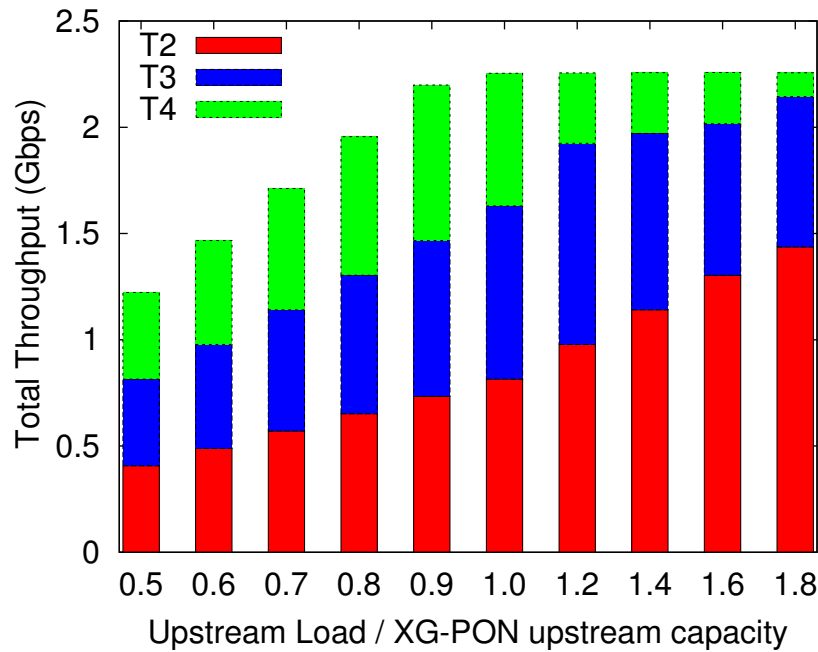
Figure 6.8: Throughput of T2,T3 & T4 in XGIANT at different upstream load in XG-PON

in the XG-PON module for ns-3, this chapter discusses the refinements XGIANT has made to the policies of GIANT and justifies, by evaluation, the choice of values for two key GIANT parameters. When evaluated for performance, XGIANT proved its capability and superiority over EBU, to provide low and prioritised mean queuing-delay for the 3 T-CONT types. The prioritised treatment given by XGIANT for all 3 T-CONT types is validated by the network-wide throughput assurance of XGIANT. Overall, XGIANT proved to be an efficient QoS-aware DBA for XG-PON, while being simpler than its recent counterpart, EBU.

The next chapter will look at evaluating the mean queuing-delay and throughput - based performances of both XGIANT and EBU DBAs, in order to establish the suitability of these DBAs in providing differentiated QoS for 3 different classes of aggregated upstream traffic in LTE when using the XG-PON backhaul.

Chapter 7

Design, implementation and evaluation of suitable DBAs for the dedicated XG-PON backhaul in LTE

This chapter presents the design, implementation and evaluation details of two DBA mechanisms, which are designed, based on the performance evaluations of the XGIANT (Chapter 6) and EBU DBAs, to provide suitable QoS treatment for different classes of upstream traffic in LTE across the dedicated XG-PON backhaul.

Deficit XGIANT (XGIANT-D) and Proportional XGIANT (XGIANT-P) are designed by improving the QoS policies of XGIANT and EBU, both of which, when evaluated, failed to ensure the required prioritisations in terms of mean queuing-delay for three different classes (conversational voice, peer-to-peer video and best-effort Internet) of upstream LTE traffic across the XG-PON backhaul. The improved QoS policies in XGIANT-D and XGIANT-P DBAs: ensure the required prioritisations among the three classes of traffic; smooth the burstiness of eNB-aggregated voice and video traffic; introduce two efficient bandwidth allocation policies (hence two flavours of suitable DBAs) for the best-effort Internet traffic; simplify the bandwidth allocation policies for all the classes of traffic; and use an efficient fairness policy, as opposed to the QoS policies in XGIANT (and EBU), so that suitable DBAs are feasible for the FMC of XG-PON and LTE.

In presenting the performance evaluations of XGIANT-D and XGIANT-P DBAs, first, the chapter presents a brief recollection of XGIANT and EBU DBAs in section 7.1, to discuss the suitability of the QoS policies when serving realistic upstream application traffic (of different classes) of LTE in the dedicated XG-PON backhaul. Section 7.2 presents the simulation environment implemented in the ns-3 network simulator, in terms of the integrated network architecture of XG-PON and LTE, the QoS metric conversion from LTE to XG-PON, the configuration of LTE network and the generation of realistic application traffic in the LTE upstream. Section 7.3 presents the evaluations of XGIANT

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE

7.1 Suitability of XGIANT and EBU DBAs for the
dedicated XG-PON backhaul in LTE

and EBU, both of which indicate their inability to ensure prioritised and fair mean
queuing-delay performance for the three types of realistically-generated upstream LTE
traffic in the under-loaded and marginally over-loaded XG-PON backhaul. Section 7.4
presents the improvements in and the implementation details (in ns-3) regarding the
XGIANT-D and XGIANT-P DBAs. Section 7.5 presents the evaluations with regard to the
improved QoS policies in XGIANT-D and XGIANT-P, both of which ensure prioritised
and fair mean queuing-delay performances for all three upstream application traffic of
LTE at the same traffic loading scenarios in the XG-PON backhaul. Section 7.6 presents
the performance evaluation of all the DBAs, for the additional metrics of queuing-
jitter across XG-PON and differences in datarate and traffic burstiness across XG-PON
and LTE, to demonstrate the relative (to XGIANT and EBU) impact of the improved
QoS policies in XGIANT-D and XGIANT-P. Section 7.7 summarises the contributions to
conclude the chapter.

7.1 Suitability of XGIANT and EBU DBAs for the dedicated XG-PON backhaul in LTE

This section presents a technical discussion on the adaptability and suitability of the
QoS policies of EBU and XGIANT DBAs, in the context of XG-PON backhaul for LTE
upstream, as both DBAs are designed to provide QoS only in a stand-alone XG-PON
serving fixed-broadband users.

EBU is a recently proposed DBA to provide differentiated QoS for 3 different classes of
traffic (T-CONTs) in the stand-alone XG-PON. The design, implementation and evalu-
ation of XGIANT DBA, presented in Chapter 6 also presents a second, but simpler and
more efficient DBA than EBU for XG-PON to provide differentiated QoS for the same 3
classes of traffic. Both EBU and XGIANT DBAs were designed for an XG-PON network
serving fixed broadband users; hence the DBA evaluations used deterministic traffic
patterns. Since the aggregated traffic pattern (packet arrival distribution and bursti-
ness) in LTE is highly subjective to realistic network environments, the performance of
both EBU and XGIANT is unpredictable in the context of serving aggregated upstream
traffic in LTE across an XG-PON backhaul.

As explained in Chapter 6, EBU and XGIANT are primarily based on the combined effect
of equations 6.1 and 6.3, resulting in the following equation for maximum possible
$GrantSize_{k,i}$ allocation ($=AB_{k,i}$) for an $AllocID_{k,i}$:

$$AB_{k,i} = \frac{(SI_k * 125\mu s) * Ratio_k * C_{XG-PON}}{N_k} \qquad (7.1)$$

where $Ratio_k$ is the relative (against effective upstream capacity of XG-PON, C_XG-

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE

*7.1 Suitability of XGIANT and EBU DBAs for the
dedicated XG-PON backhaul in LTE*

PON) proportion of the maximum datarate (MDR_k) assigned to each T-CONT type k (= 2, 3 and 4); $i \, \epsilon \, N_k = N$ with N_k and N are the number of ONUs and T-CONT type k AllocIDs, respectively, in the entire XG-PON network.

Both EBU and XGIANT also preserved the original traffic classification in the GIANT DBA, with T2 serving assured bandwidth type traffic, T3 serving assured and non-assured traffic and T4 serving Best Effort type traffic.

Major differences between XGIANT and EBU lie in the values chosen for two variables: 1) SI_k, to vary the frequency at which an $AllocID$ is attempted[1] for $GrantSize$ provisioning and the portion and 2) $Ratio_k$ which determined the portion of C_{XG-PON} allocated to each T-CONT type (T2, T3 and T3) in the network. Though XGIANT presented better mean queueing-delay performance, EBU had a better DBA framework by having a fairness policy, namely the inter-ONU fairness, which rotated, in a round-robin manner, the ONU, whose $AllocID$ is first given $GrantSize$ in each allocation cycle[2]; inter-ONU fairness (detailed explanation is provided later in section 7.4.5) ensured that the effect of congestion in the upstream (the non-allocation of $GrantSize$ to only the last few $AllocID$s) is spread across all the ONUs

However, there is little dynamism in the QoS framework of EBU and XGIANT to account for the unpredictable nature of (aggregated) upstream traffic in LTE backhaul, due to the following reasons:

- The priorities between the 3 T-CONT types are governed by a strict order of $GrantSize_{k,i}$ allocation among AllocIDs of different T-CONT types, with AllocIDs of same T-CONT type receiving no traffic prioritisation among them.

- $Ratio_k$ is fixed throughout the whole operational duration of XG-PON and equal for all 3 T-CONT types, resulting in a fixed upper-bound $GrantSize_{k,i}$, regardless of the T-CONT type and queue occupancy of the AllocID. The fixed values chosen by the DBAs (= 0.67) also creates two more challenges: 1) both T3 and T4 T-CONT types may individually suffer under-utilisation of XG-PON upstream capacity when all AllocIDs of other T-CONT types have no instantaneous queue occupancies; 2) best-effort (T4) traffic is completely starved of upstream utilisation in XG-PON when XG-PON upstream is over-loaded by the combined traffic of T2 and T3 AllocIDs.

- Both DBAs maintain a complex order of visiting AllocIDs within an allocation cy-

---

[1]An $AllocID$ is actually provisioned with $GrantSize$ at the expiry of down counter only if there is more space within an upstream frame and the upstream frame. However, since XGIANT and EBU uses more than one upstream frame cycles for an allocation cycle of some T-CONT types, the chances of an $AllocID$ not getting a $GrantSize$ at the expiry of down counter is not negligible

[2]An allocation cycle = one round of $GrantSize$ allocation to all the $AllocIDs$ in the XG-PON network, up to $AB_{k,i}$. When XG-PON upstream is not congested for capacity and every AllocID having non-zero T-CONT queue occupancy, EBU or XGIANT would actually serve each T2 AllocID once, each T3 AllocID twice and each T4 $AllocID$ once to provide upstream transmission opportunity up to $AB_{k,i}$

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE   *7.2 Simulation Environment*

cle, due to the (attempted) serving of T2 AllocIDs first, followed by the assured portion of T3 AllocIDs, queue occupancy reports of T4 AllocIDs and non-assured portion of T3 AllocIDs respectively before finally serving the T4 AllocIDs. This also creates ambiguous prioritisation between the T2 and T3 when XG-PON is under-loaded (mean queuing-delay behaviour in Figure 6.7 in Chapter 6 for upstream load ratio < 0.8, where upstream load ratio $= \frac{Total\ Upstream\ Load}{C_{XG-PON}}$).

Due to the above reasons, this chapter predicts that the DBAs (EBU and XGIANT) designed for fixed-broadband user traffic may not be effective in terms of queueing-delay and capacity utilisation performances when employed in a dedicated XG-PON backhaul serving realistic upstream traffic of LTE, and proceeds to evaluate the hypothesis in the following section.

7.2 Simulation Environment

This section presents the details of the simulation environment that is used for evaluating the suitability and adaptability of the QoS principles of XGIANT and EBU DBAs in serving upstream traffic in LTE with XG-PON backhaul. Due to the strong real-world presence of XG-PON in backhaul transportation and LTE in wireless last-mile, the simulation environment presented here strives for standard-compliance and realistic characterisation of the entities in the simulation environment so that the results and discussion can directly be attributed to real-world challenges.

This section first presents the converged network architecture of XG-PON and LTE used in the evaluations in this chapter, followed by the details of upstream application models in LTE and the QoS metric conversion scheme used in the integrated network architecture respectively; details regarding the number of UEs and eNBs in the LTE network are presented thereafter; the section is concluded with the explanation of two simulation scenarios used in this chapter.

7.2.1 Integrated Network Architecture of XG-PON and LTE

The converged network architecture of XG-PON and LTE shown in Figure 7.1 represents the architecture that is implemented in ns-3, for the evaluations presented in this chapter. The *independent* architecture also follows the justification provided in section 3.2.1 of Chapter 3) with regard to choosing a suitable network architecture for the FMC of XG-PON and LTE.

In Figure 7.1, XG-PON is placed between the Gateway and the eNB; OLT is placed close to the Gateway and ONU, close to the eNB. Gateway represents a data-plane entity in

7. Design, implementation and evaluation of
   suitable DBAs for the dedicated XG-PON
   backhaul in LTE   *7.2 Simulation Environment*
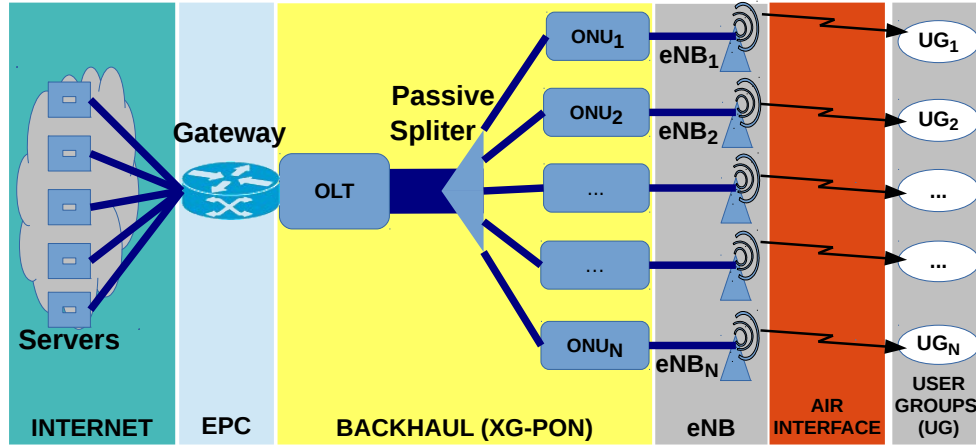
Figure 7.1: Integrated Network Architecture of XG-PON and LTE

the Core Network of LTE, since the evaluations of this chapter deal only with the data-plane traffic in LTE. Ethernet is assumed to bridge the interfaces of Gateway-OLT and ONU-eNB, since both XG-PON and LTE standards have support for Ethernet to connect external entities. Use of existing Ethernet interface for bridging the entities of different technologies also enables the integrated network architecture to serve a variety of wired and wireless technologies in the last-mile, while allowing for independent evolution paths of the individual technologies (eg: from XG-PON to XGSPON and LTE to LTE-A) only to require minimal changes at the interface for integration of the evolved versions of the technologies (eg: integration of XGSPON and LTE-A)

7.2.2 Application Modelling in LTE Upstream

The following three types of upstream application traffic in LTE are used in the evaluations presented in this chapter. Realistic traffic models are used to generate traffic in ns-3 representing real-world voice, video and best-effort applications.

- **Voice:** Several authors[88, 100, 50] recommend a two-state (ON-OFF) model for VoIP, with exponentially distributed ON and OFF durations. During ON state, fixed-size voice packets are transmitted at a constant bit rate (CBR). To represent a High Definition (HD) VoIP application in a UE, during the ON state, data is generated at a CBR of 64 kbps with a packet size of 160 Bytes to achieve 20ms inter-frame gap as in common voice codecs such as G.722[48]. Based on the values by Zaki et al.[100], duration of ON and OFF states are configured to have exponential distributions with a mean of 0.35 and 0.65 seconds respectively.

- **Video:** The PPBP application model [7] is used to generate video traffic. Based on the values used by Singh et al. [88], a Hurst parameter value of 0.9 is used for bursty traffic model for video. A mean-datarate of 300kbps[69] and a packet size of 795 Bytes[63] is used, to represent HD peer-to-peer video conferencing.

7. Design, implementation and evaluation of
   suitable DBAs for the dedicated XG-PON
   backhaul in LTE   *7.2 Simulation Environment*

- **Best-Effort:** An accurate traffic model for best-effort traffic in mobile networks
  is not common in the literature, especially for bandwidth-hungry applications
  in LTE upstream. However, upstream best-effort traffic can be attributed to web-
  browsing by a mobile user, with behavioural closeness to users of fixed broadband
  networks. Hence, best-effort traffic is modelled as a self-similar and long-range
  dependent traffic, using PPBP application model in ns-3, with a Hurst parameter
  of 0.5 [88]. 2Mbps (chosen by means of evaluation) is used for the datarate of
  BE traffic from a UE, to allow each UE to achieve very high instantaneous/burst
  datarate.

7.2.3 QoS Metrics Conversion Scheme in the Integrated Network Architecture

XG-PON and LTE (as explained in sections 2.2.3 and 2.3.1 of Chapter 2 respectively)
have defined different sets of QoS metrics to represent traffic classification across the
respective standards. However, in the XG-PON backhaul for LTE, it is necessary to have
a QoS metric conversion scheme to preserve the classification of the traffic differentia-
tion across the integrated network architecture of the two standards. From the many
QoS metric conversion schemes described in section 3.4.1 of Chapter 3, the following
standard-complying and static QoS-metrics conversion scheme is implemented in ns-3
for the integrated network architecture presented earlier in section 7.2.1.

- In an upstream data transfer from UE to *Gateway*, the UE imprints each packet
  with a QCI value as per the QoS policies in LTE and transmits the packets in the
  upstream using the associated radio bearer.

- Once the packets arrive at the eNB, a discrete Differentiated Services Code Point
  (DSCP) value is added in the internal IP packet header, for each QCI. The packet
  is then sent towards the ONU, over the Ethernet link.

- ONU receives these packets and associates each DSCP value to a unique T-CONT
  type, before collecting the packets at the queues dedicated for each T-CONT type.

- Within an ONU, an $AllocID$s and T-CONT types share fixed relationship. There-
  fore, when the OLT provisions $GrantSize$ for the $AllocIDs$, the ONUs may trans-
  mit the packets from the respective T-CONT (type) queues in the upstream.

- When the *Gateway* receives the packets from the OLT via the Ethernet link, the
  reciprocal of the above conversion between QCI, DSCP and T-CONT type is used
  to retrieve the original QCI of the packet at the *Gateway*.

Using this conversion scheme, a DBA in the XG-PON backhaul for LTE can preserve the
QoS classification of the upstream traffic types in LTE when the aggregated traffic is
transmitted from the eNB towards the Gateway. For example, since XGIANT and EBU

Table 7.1: QoS metric conversion scheme for the FMC architecture of XG-PON and LTE

Application Type	EPS Bearer type	LTE QCI	DSCP	T-CONT Type
LTE Signalling	non-GBR	1	CS7 - 56	T1
Conversational voice	GBR	2	EF - 46	T2
Peer-to-peer video	GBR	4	CS4 - 32	T3
Best-effort Internet	non-GBR	9	BE - 0	T4

define only four traffic classes in XG-PON and this chapter is interested in analysing only three application types (voice, video and best-effort) in LTE, the above static QoS metrics conversion mechanism is sufficient to provide distinct QoS between all the three upstream application types in LTE. Table 7.1 shows the static QoS-metrics conversion scheme used in the evaluations of this chapter. Since both XGIANT and EBU uses fixed $GrantSize$ allocation for T1, a sample proposal is provided with the LTE signalling traffic between eNB and *Gateway* taking advantage of T_1 T-CONT type in XG-PON, though T_1/signalling traffic is not part of the evaluations in this chapter.

If there are more classes of traffic in LTE upstream than in XG-PON, the above QoS metric conversion scheme will need to be converted into a dynamic QoS conversion scheme between T-CONTs and QCIs, to guarantee the QoS requirements of 3GPP-TS-23.203 in LTE.

7.2.4   Number of UEs per eNB

A ratio of 2:2:1 as used in [100, 14] is chosen for the number of UEs generating voice:video:best-effort traffic, while assuming that each UE would use only one application at any given time. The total number of UEs attached to an eNB is selected uniformly at random in the range of 105 - 145 (mean = 125), with a step of 5 in between, so that the best-effort users are exact integers. These numbers ensure that the 75th percentile of the total instantaneously aggregated datarate per eNB does not exceed the upstream capacity ($\sim$70 Mbps, by means of detailed evaluations performed in ns-3) of the eNB in the LTE module for ns-3.

All UEs are placed randomly around the attached eNB, within a radius of 4km, while eNBs (each with a single cell) are placed in a straight line, with 6km of inter-eNB distance, to avoid interference between UEs of adjacent eNBs. All UEs remain fixed to their position throughout the simulation, as the objective in the experiments is to evaluate the per-eNB-aggregated application traffic in the upstream. PFS (as explained in section 2.3 of Chapter 2) in ns-3 is used in the LTE air interface to ensure balanced upstream fairness and throughput resource allocation for all the UEs in each eNB.

7. Design, implementation and evaluation of
   suitable DBAs for the dedicated XG-PON
   backhaul in LTE   *7.2 Simulation Environment*

Table 7.2: Analysis of datarate for the eNB-aggregated traffic

Scenario	Mean Datarate(Mbps) voice:video:best-effort	Mean Datarate (%) voice:video:best-effort	Total Mean Ratio Datarate(Mbps)	Load
52-eNB	95 - 277 - 1288	5.7 - 16.7 - 77.6	1660	0.74
80-eNB	146 - 422 - 1967	5.8 - 16.6 - 77.6	2534	1.13

7.2.5 Simulation Scenarios

To realise different upstream traffic loading in XG-PON, two scenarios are modelled in the LTE network, such that the XG-PON upstream is:

- under-loaded and

- marginally over-loaded

in terms of the instantaneous mean datarate of total upstream traffic in LTE. While the under-loaded scenario represents a scenario of XG-PON having sufficient capacity for upstream traffic in LTE, the marginally over-loaded scenario represents a scenario of (mean) total upstream traffic in LTE fully occupying the upstream capacity of XG-PON. Hence, the under-loaded and marginally over-loaded scenarios represent real-world scenarios experienced by the service providers when using XG-PON backhaul with growing number (or increasing amount) of LTE users (or upstream traffic in LTE).

The LTE module in ns-3 is configured with 52 eNBs and 80 eNBs (with each eNB connected to a single ONU) to represent a sample scenario of under-loaded and marginally over-loaded scenario in XG-PON. Table 7.2 shows the sum of mean datarate across all the eNBs in each scenario. In both scenarios, voice traffic has the smallest value for the total mean instantaneous datarate, followed by the video traffic; best-effort has the highest amount and percentage of traffic (77.6%) even though a ratio of 2:2:1 was maintained between the number of users generating the traffic types, in each eNB. As expected, the aggregated mean datarate value of each application type is increased in 80-eNB while the ratios among them remain the same.

However, the total instantaneous (per-millisecond) datarate of each application in LTE backhaul, takes a range of values, in both scenarios. This results in the XG-PON backhaul experiencing a range of total upstream load ratios. Figure 7.2 shows this behaviour for a single run in the 52-eNB scenario; the mean, minimum and maximum values in the y-axis indicate the mean, 25th and 75th percentile, respectively, of the total (aggregated for all the UEs) instantaneous mean datarate per eNB, per application; the x-axis represents the Cumulative Distribution Function (CDF) with regard to the number of eNBs in LTE. A similar behaviour was observed for the 80-eNB scenario as well. These values indicate that realistic traffic in LTE may also result in such instantaneously bursty behaviour, requiring the QoS policies of the DBAs in XG-PON to be sufficiently adaptable/dynamic towards the unpredictable nature of traffic arrival in XG-PON to

7. Design, implementation and evaluation of
   suitable DBAs for the dedicated XG-PON
   backhaul in LTE
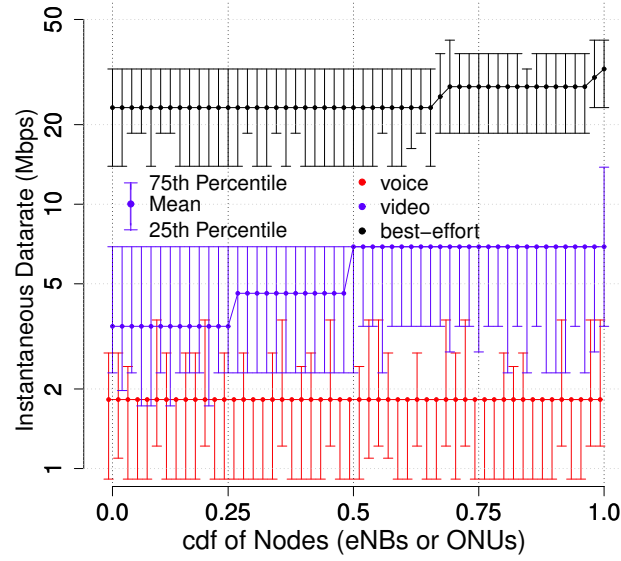
7.3 Experiments

Figure 7.2: Per-eNB instantaneous datarate for a single random run in 52-eNB scenario

provide effective QoS treatment in XG-PON backhaul. The experiments presented in the following section will evaluate the performance of XGIANT and EBU DBAs under such bursty conditions.

7.3 Experiments

This section presents the evaluation metrics and the performance evaluation of XGIANT and EBU DBAs, for the experiment environment presented earlier in section 7.2. The performance evaluation is geared towards understanding the suitability of XGIANT and EBU DBAs, when serving three different classes of upstream traffic in the XG-PON backhaul of LTE.

7.3.1 Evaluation Metrics

The mean queuing-delay is a crucial metric of evaluation for a DBA in XG-PON. That is since the propagation delay of XG-PON upstream is constant and if assumed to be similar for any ONU, the mean queuing-delay metric would give a clear indication of the mean delays experienced by the application types LTE across the XG-PON backhaul. From the mean queueing-delay metric, the total delay across the XG-PON backhaul for an application traffic in LTE can be calculated as explained in section 2.2.4 of Chapter 2).

The mean queueing-delay metric is also the same metric used in section 6.2.2 of Chapter 6, where the calculation of mean queuing-delay in each T-CONT queue only accounts for queuing-delay of the packets being transmitted in the upstream, after entering the T-CONT queues. However, since each eNB is configured to transmit realistic application traffic models in this chapter, each eNB-aggregated application traffic may have a different traffic arrival behaviour resulting in different instantaneous datarates across different eNBs (as in Figure 7.2). The following two additional metrics can also be derived from the mean queuing-delay to provide further insight into the performances of XGIANT and EBU DBAs in XG-PON backhaul for LTE:

- **Strict Priority:** Priority is a significant part of providing QoS treatment to different types of traffic entering XG-PON from LTE, in the upstream. As this chapter is concerned with the QoS treatment for per-eNB-aggregated traffic in LTE across the XG-PON, an assumption of equal propagation between each ONU and the OLT would yield a direct relationship between the mean queuing-delay in the T-CONT queues in an ONU and the effective priority assured by the DBA towards each application type aggregated in the corresponding eNB. That is, the higher the mean queuing-delay experienced by the packets of a T-CONT queue in an ONU, the lower the priority given to the application type of the eNB, since higher priority applications can be expected to require a lower delay in LTE backhaul, as in the LTE air interface (refer Table 2.1). A strict priority in the LTE backhaul is when the DBA in XG-PON assures smaller mean queuing-delay for every $AllocID$ of a higher priority T-CONT type in the entire XG-PON compared to the mean queuing-delay values obtained for any $AllocID$ of a lower priority T-CONT type.

- **Fairness:** Due to the uncoordinated variation of aggregated application traffic datarate among different eNBs, it is inevitable to have different mean queuing-delay values for $AllocID$s of same T-CONT type across different ONU. However, T-CONT queue size is small and similar across different ONUs, fairness in delay across different eNBs can be inferred from the range of mean queuing-delay values experienced by the $AllocID$s of the same T-CONT type in different ONUs. A fairer treatment by the DBA for an eNB-aggregated application type in LTE is when all the $AllocIDs$ of a T-CONT type experience a smaller range of mean queuing-delay values compared to all the $AllocIDs$ of a different T-CONT type.

7.3.2   Evaluation of XGIANT and EBU in the XG-PON backhaul in LTE

With the simulation environment prepared and the evaluation metrics defined, this section presents the performance evaluations of XGIANT and EBU in the context of an XG-PON backhaul for LTE. Each experiment presented in this section (and chapter) is simulated for a total duration of 20 simulation seconds, and repeated for 10 times, each with a different seed number, in order to achieve mutually-exclusive randomness

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE
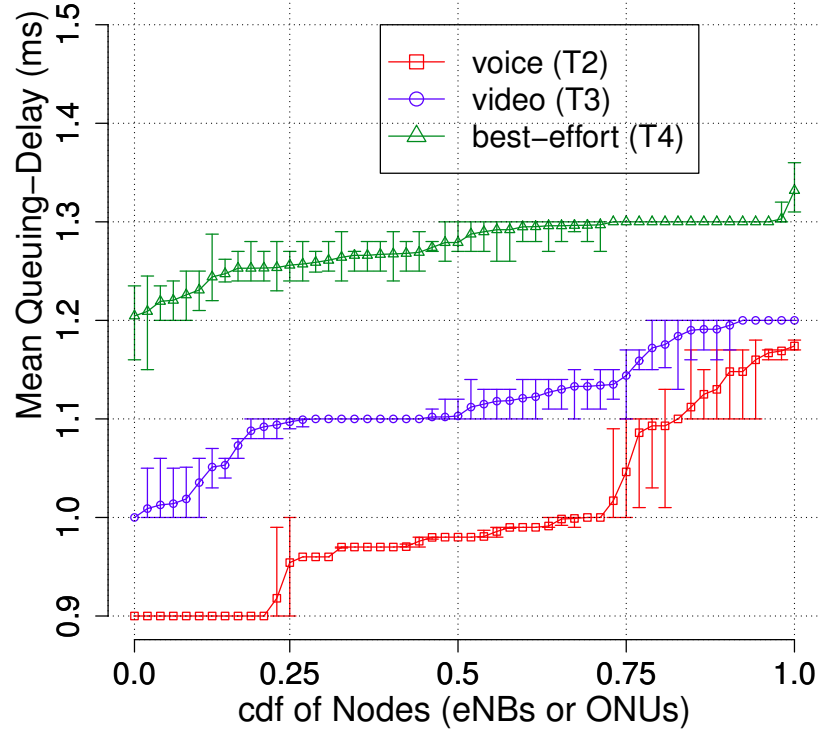
7.3 Experiments
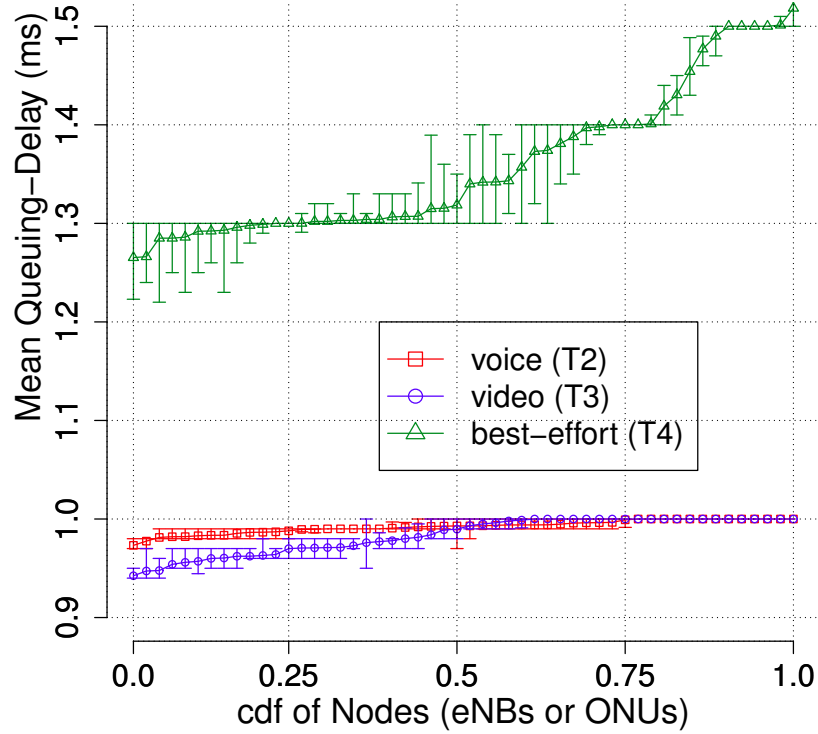
(a) XGIANT

(b) EBU

Figure 7.3: Mean queuing-delay in 52-eNB scenario

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE
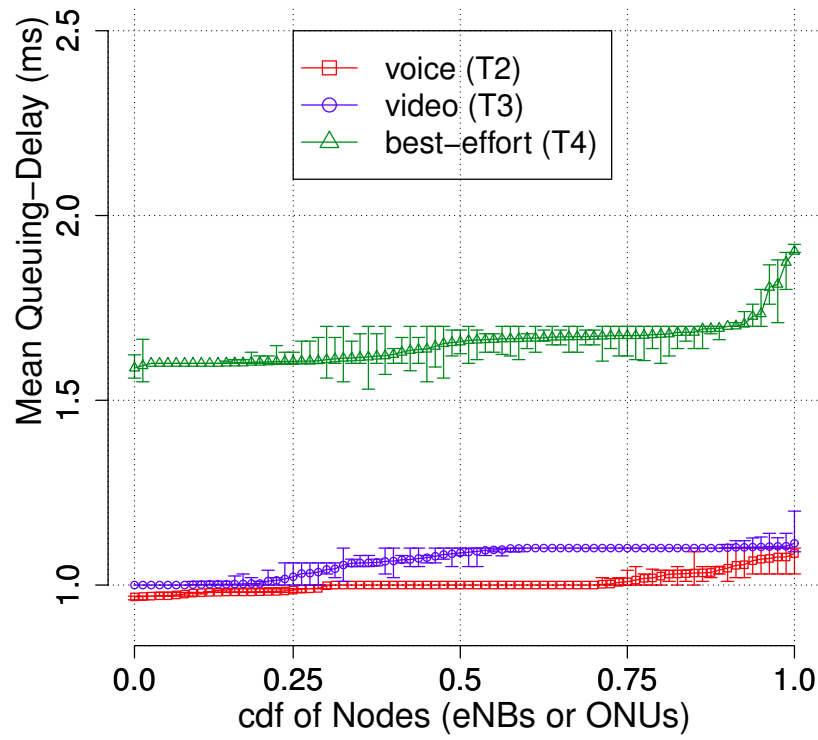
7.3 Experiments

(a) XGIANT
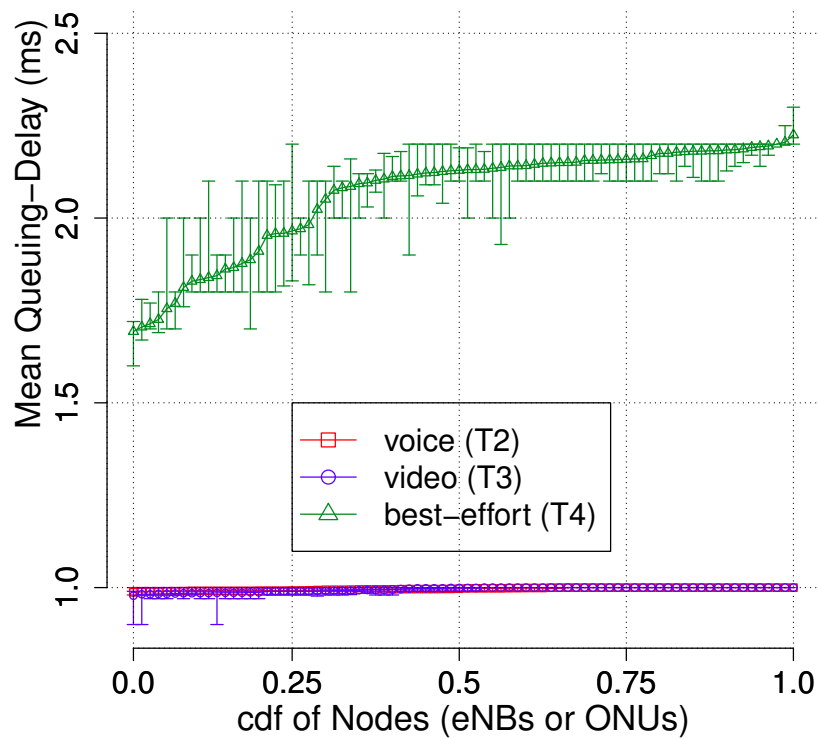
(b) EBU

Figure 7.4: Mean queuing-delay in 80-eNB scenario

7. Design, implementation and evaluation of
   suitable DBAs for the dedicated XG-PON
   backhaul in LTE   *7.3 Experiments*

in the generation of application traffic and the placement of UEs around the eNBs for each experiment; three T-CONT types are used in each ONU, with one for each application type of the eNB; a value of 5KB is used for each T-CONT queue size to provide fair comparison between different AllocIDs of each and different T-CONT types; as in previous chapters, the propagation delay between each ONU-OLT pair is 0.4ms representing 60km distance between each ONU and the OLT.

Implementation of XGIANT and EBU in ns-3 follows the details given in Chapter 6 and [35] respectively. From equation 7.1, $N = N_k$ is set to 52 or 80 as per the scenario. For allocation cycle duration XGIANT uses the fine-tuned values from Chapter 6 ($SI_2 = SI_{3G} = 1$, $SI_{3M} = SI_4 = 2$) and EBU uses its original values in [35] ($SI_2 = SI_{3G} = 5$, $SI_{3M} = SI_4 = 10$). XGIANT employs no fairness policy to rotate first served $AllocID$ in an allocation cycle, while EBU uses its inter-ONU fairness policy, which begins each allocation cycles with the $GrantSize$ provisioning for the $AllocID$ from the next ONU (in round-robin fashion) in XG-PON

Figures 7.3 and 7.4 indicate the mean queuing-delay performance of XGIANT and EBU DBAs in the 52-eNB and 80-eNB scenarios respectively. All plots indicate both the average (points in the plots) and range (error bars in the plots) of the mean queuing-delays from the 10 runs of each experiment.

In both scenarios both XGIANT and EBU fail to provide strict priority among the three T-CONT types. For example, in the 52-eNB scenario XGIANT fails to provide lower mean queuing-delay for some of the T2 $AllocIDs$s compared to the minimum values obtained by any of the T3 $AllocIDs$ (Figure 7.3a, at x = 0.7 - 1.0) in the 52-eNB scenario; EBU fails to provide lower mean queuing-delay for most of the T2 $AllocIDs$, again when compared to the minimum mean queuing-delay values of the T3 $AllocIDs$ (Figure 7.3b, at x = 0.0 - 0.5).

These failures are mainly due to the use of the expiring service intervals and relative ratios of MDR_2 and MDR_3, in the context of highly varying instantaneous datarates between eNB and ONU for the aggregated applications; the realistically generated and aggregated application datarates arriving in the upstream of XG-PON presents a range of values for the $DBRu$ requests from each $AllocID_{k,i}$; the DBA distribute the per-upstream frame capacity ($= \frac{Capacity_{XG-PON}}{125us}$) accordingly. Even though both XGIANT and EBU provisions all the T2 $AllocIDs$ before any T3 $AllocID$, within a window of 2 allocation cycles , T3 $AllocIDs$ receive more upstream transmission opportunities due to using GDR_3. That is, within 2 allocation cycles (= 2 frame cycles[3] in XGIANT and 10 frame cycles in EBU), when down counters are expired, $AllocID_{3,i}$ can receive 3 $GrantSize$ provisions compared to only 2 for the $AllocID_{2,i}$.

Because aggregated video traffic is more bursty than aggregated voice traffic, this re-

---

[3]A framing cycle = allocation of $GrantSize$ within a single upstream frame

7. Design, implementation and evaluation of
    suitable DBAs for the dedicated XG-PON
    backhaul in LTE

7.4 Improving the XGIANT DBA for XG-PON in LTE
Backhaul

sults in some of the T3 $AllocIDs$ being transmitted at a higher frequency than some
T2 $AllocIDs$. Hence, the video traffic from some of the eNBs receives lower mean
queuing-delays than the voice traffic from the same or different eNBs. Less number of
T3 $AllocIDs$ receive lower delays in the 80-eNB scenario (Figure 7.4), compared to
the 52-eNB scenario (Figure 7.3) because of the AB per $AllocID$ (= AB_k * SI_k / N_k
)being lower when N_k increase from 52 to 80.

In terms of fairness, for T2 (voice) and T3 (video), XGIANT shows a wider range of
mean queuing-delay values than EBU, in the 52-eNB and 80-eNB scenarios, indicat-
ing a higher degree of unfairness (range of values) than EBU in both scenarios. This
is expected as XGIANT has no fairness policy as opposed to EBU. For T4 (best-effort)
EBU provides a wider range of values than XGIANT in both scenarios, indicating EBU's
inability to maintain fairness, even with the inter-ONU fairness policy. Overall, even
though EBU employs a reasonably good fairness policy, these results indicate inad-
equate fairness assurance by the DBAs for each T-CONT type, in the context of the
XG-PON backhaul for LTE.

In summary, these observations indicate that the QoS and fairness policies governing
the $GrantSize$ allocation in both XGIANT and EBU, which were originally designed for
the standalone XG-PON, are unable to provide fairness and strict priority for the mean
queuing-delays when serving the bursty voice, video and best-effort traffic with highly-
varying instantaneous upstream load ratios. This behaviour disqualifies both XGIANT
and EBU from being XG-PON-standard-compliant in LTE backhaul (refer section 2.2.3
for details on the XG-PON-standard-compliance of a DBA). Both XGIANT and EBU also
fail to provide reasonable fairness for all the T-CONT traffic types simultaneously, in
under-loaded and marginally over-loaded scenarios of XG-PON upstream loading.

This is because both XGIANT and EBU do not have highly adaptive QoS and fairness
policies fine-tuned for: (1) the unpredictable burstiness behaviour of traffic generated
using realistic application models in ns-3 and (2) the uncoordinated aggregation of
each traffic across different eNBs in LTE, causing uncoordinated traffic loading in XG-
PON among and between each T-CONT type. The following section presents the de-
sign and implementation of improved QoS policies in XGIANT, to ensure better mean
queuing-delay (and throughput) performances for the eNB-aggregated voice, video and
best-effort applications across the XG-PON backhaul in LTE.

7.4   Improving the XGIANT DBA for XG-PON in LTE Backhaul

This section presents the design and implementation details of two XG-PON-standard-
compliant DBAs, namely the Deficit XGIANT or XGIANT-D and Proportional XGIANT
or XGIANT-P, which, when compared to XGIANT and EBU, are capable of providing
improved, strictly-prioritised and fairer mean queuing-delay throughput among the

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE

*7.4 Improving the XGIANT DBA for XG-PON in LTE
Backhaul*

three aggregated application types in LTE backhaul. XGIANT is chosen as the base DBA for designing a custom made DBA suitable for the XG-PON backhaul in LTE, due to the relative superiority of XGIANT over EBU in the mean-queuing delay performance in XG-PON (see section 6.6.1 of Chapter 6).

XGIANT-D and XGIANT-P improve XGIANT in the aspects of choosing a fine-tuned relative MDR ratio among T-CONT types by means of exhaustive evaluations, removing the use of down counters, customising the QoS policies governing the $GrantSize$ allocation for $AllocID$s of each T-CONT type, simplifying the order of serving $AllocID$s of different T-CONT types in an allocation cycle and introducing an intra-TCONT-type fairness policy. Specifically, regarding the $GrantSize$ allocation for $AllocID$s, both XGIANT-D and -P use the same improvements over XGIANT for T2 and T3; the difference between the improved DBAs is only in how T4 $AllocID$s are provisioned $GrantSize$. XGIANT-D follows the concepts of deficit round robin scheduler to calculate a deficit grant every frame to produce deficit-based $GrantSize$ allocation for the T4 $AllocID$s every other upstream frame. XGIANT-P produces a proportionally weighted grant request every upstream frame, based on the concepts of weighted round robin scheduler, for each T4 $AllocID$, to compete against the actual queue occupancy report when provisioning the $GrantSize$ for the T4 $AllocID$s.

7.4.1 Strict priority among T2, T3 and T4

Observations in section 7.3.2 indicated that the failure of XGIANT and EBU to provide strict priority was mainly due to some of the T3 $AllocIDs$ experiencing lower mean queueing-delay values than the lowest mean queuing-delay values seen in the T2 $AllocIDs$. Exhaustive evaluations of the DBA, performed by varying several parameters of the XGIANT DBA to understand the impact of the parameters on the overall performance of the DBA, indicated that different $Ratio_k$ values (based on equations 7.1) for T2 and T3 provided different relative mean-queuing-delay values between the $AllocIDs$ of T2 and T3. Recalling from section 7.1, $Ratio_k$ defines the maximum amount of $GrantSize$ that an $AllocID$ of type k can receive in an allocation cycle: the absolute values of $Ratio_k$ control the mean queuing-delay experienced by each $AllocID$ in XG-PON; the relative values of $Ratio_2$, $Ratio_3$ and $Ratio_4$ control the maximum mean queueing-delay values between $AllocIDs$ of different T-CONT types. Hence, a careful tuning of the $Ratio_k$ values will result in the strict priority between T-CONT types, 2, 3 and 4.

In order to find a more reasonable relative $Ratio_k$ value between the two T-CONT types, the stand-alone XG-PON network architecture in Figure 6.1 and simulation environment in section 6.2 of Chapter 6. Such a simple XG-PON network setup also results in a simpler (than the integrated XG-PON and LTE) network and an unbiased nature of traffic loading in the simulation environment, to enable extensive evaluations for a

wide range of values for $Ratio_k$. Each associated result presented here shows the mean queuing-delay performance of XGIANT for under-loaded, fully loaded and over-loaded XG-PON, represented by upstream load ratios equal to 0.8, 1.0 and 1.4 respectively. The discussions and the conclusions in this section, for upstream load ratios 0.8 and 1.4, will also be applicable for prioritisation in upstream load ratios less than 0.8 and more than 1.4, respectively, due to the direct impact of $Ratio_k$ values on the maximum mean queuing-delay values. The performance of EBU is presented merely for comparative purposes.

The following three rules are individually followed, when choosing different $Ratio_k$ for T2, T3 and T4.

- **Rule 1: (Figure 7.5a)** $Ratio_4 = Ratio_3 = Ratio_2$

  Here, XGIANT clearly prioritises T3 over T2, when the network is light-loaded or fully loaded. This is due to the DBAs provisioning any T3 $AllocID$ (3 counts of allocation) at a higher frequency than any T2 $AllocID$ (2 counts of allocation) within two consecutive upstream frames. For the over-loaded scenario, T2 and T3 show equal priority due to the network congestion reducing the $GrantSize$ allocation frequency for most T3 $AllocID$s.

- **Rule 2: (Figure 7.5b)** $Ratio_3 = Ratio_4 = 0.5 * Ratio_2$

  This rule mitigates the prioritisation ambiguity observed in Rule 1. When any T3 $AllocID$ is provisioned with smaller $GrantSize$ than any T2 $AllocID$ ($Ratio_3 < Ratio_2$), T3 $AllocID$s are given lower priority than T2 $AllocID$s, except in the case of under-loaded (0.8 load ratio) scenario, where the intra-TCONT-type fairness in XGIANT exploits the unused bandwidth for the benefit of the T3 $AllocID$s.

- **Rule 3: (Figure 7.5c)** $Ratio_3 = 0.5 * Ratio_2, with\ Ratio_4 = 2 * Ratio_2$

  Here, though T2 $AllocID$s are prioritised over T3 $AllocID$s, the T4 $AllocID$s experience a mistaken priority. This is because of the aggregated maximum datarate of T2 and T3 less than the effective capacity of XG-PON, resulting in a guaranteed $GrantSize$ for all T4 $AllocID$s, all three load ratios presented in the figure. Ratios such as 70-35-140 and 80-40-120 are avoided due to them resulting in highly over-provisioned XG-PON despite the ratios resulting in strict priority.

Following is the summary of observations from the above three rules:

- To obtain a higher priority for T2 than T3, $Ratio_2$ should be more than $Ratio_3$

- To avoid T3 being prioritised over T2 at light loads, $Ratio_4$ must be more than $Ratio_3$

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE

*7.4 Improving the XGIANT DBA for XG-PON in LTE
Backhaul*

(a) Rule 1

(b) Rule 2

(c) Rule 3
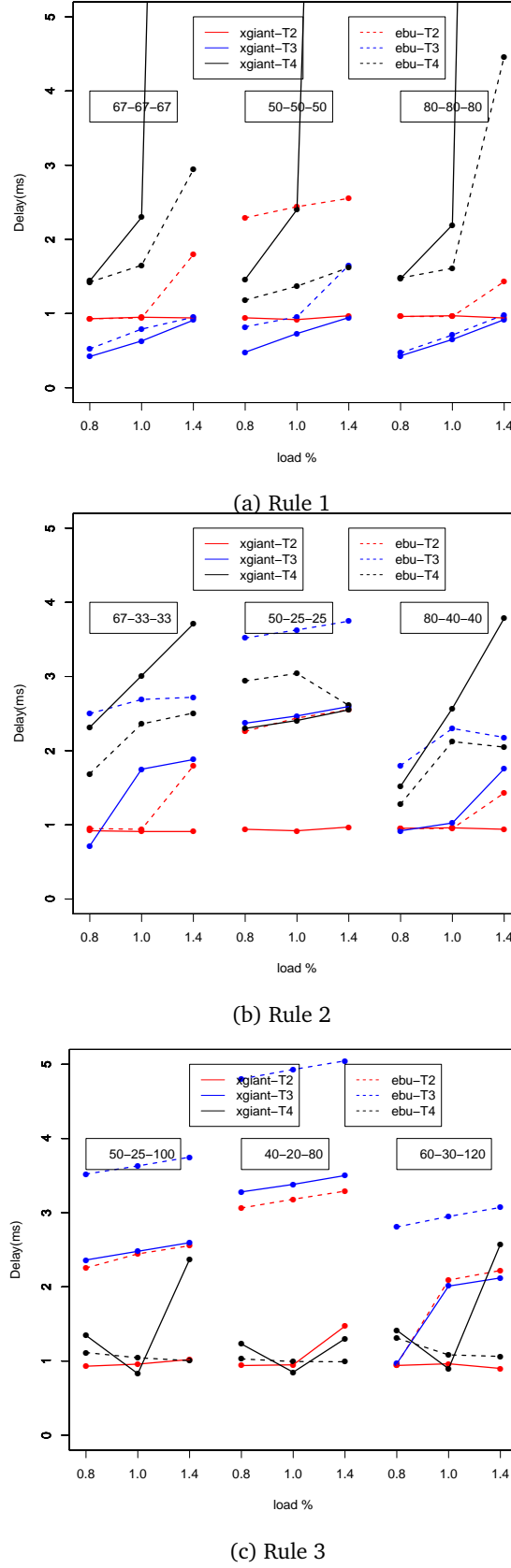
Figure 7.5: XGIANT Vs EBU for different $Ratio_k$ values for T2, T3 and T4

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE

7.4 Improving the XGIANT DBA for XG-PON in LTE
Backhaul
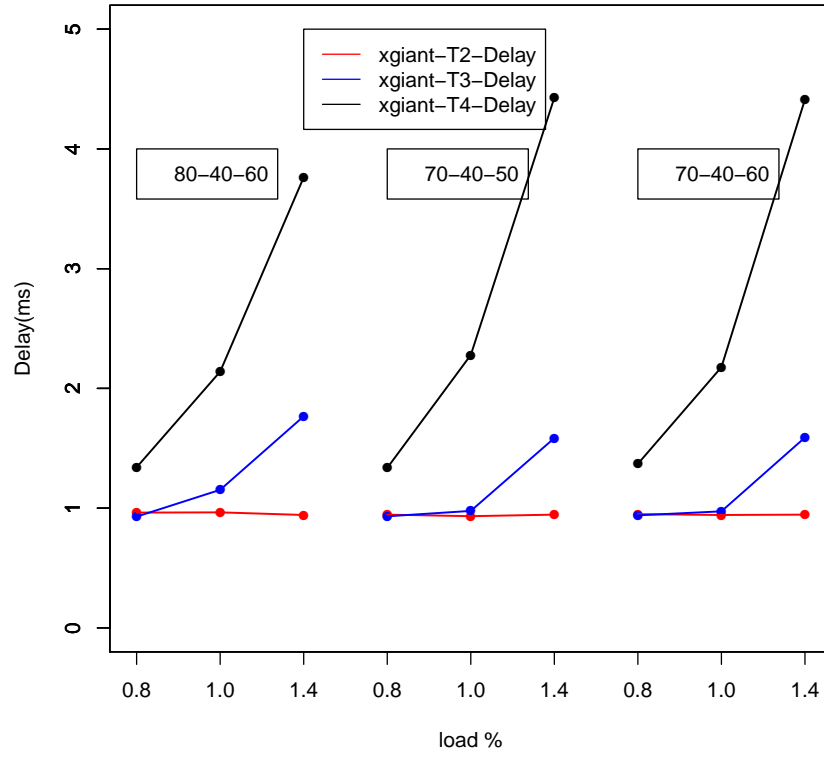
(a) Mean-delay
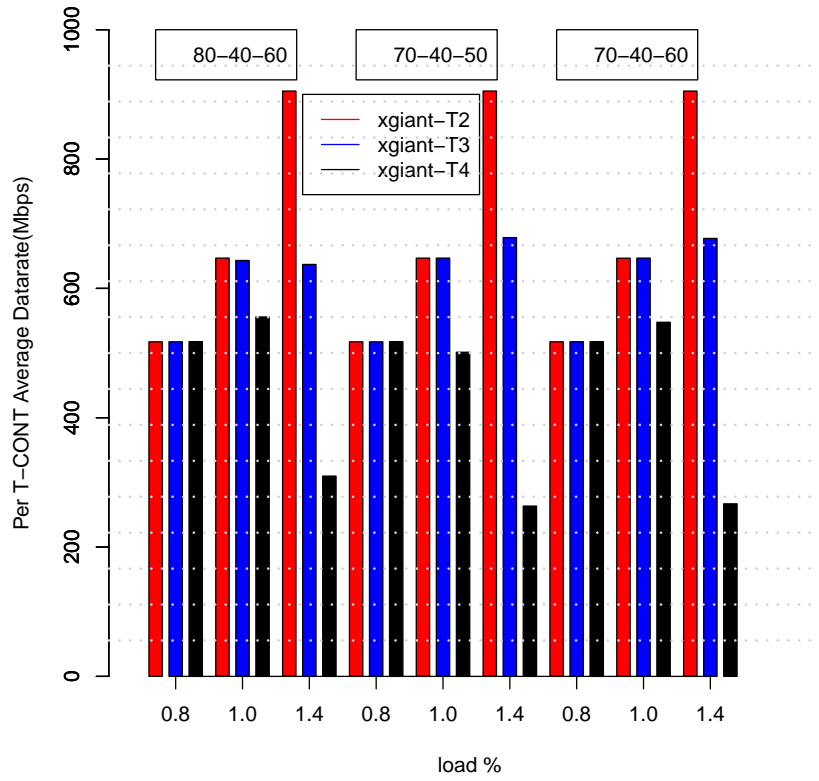
(b) Datarate

Figure 7.6: XGIANT behaviour for improved $Ratio_k$ values for T2, T3 and T4

7. Design, implementation and evaluation of
    suitable DBAs for the dedicated XG-PON
    backhaul in LTE

7.4 Improving the XGIANT DBA for XG-PON in LTE
Backhaul

Figure 7.6a presents the mean queuing-delay (and datarate) performance for three sets of relative $Ratio_k$ values, which follow these summarised rules. Corresponding throughput behaviour can also be seen in Figure 7.6b. The ratio of 70:40:60 is chosen for $Ratio_2 : Ratio_3 : Ratio_4$ in XGIANT-D and XGIANT-P, as this ratio gives both strict priority among all the T-CONT types and moderate over-provisioning factor (= $Ratio_2 + Ratio_3 + Ratio_4$), for under, fully and over -loaded XG-PON upstream. The over-provisioning factor is a crucial factor in LTE backhaul, when serving highly bursty upstream traffic, such that a very high value can add DBA-induced burstiness in the XG-PON backhaul for LTE and a very low value can result in a large range of mean queuing-delay values (hence large jitter) for $AllocIDs$ of each T-CONT type.

7.4.2 Smoothing the burstiness in the under-provisioned T3 AllocIDs

After the relative $Ratio_k$ values are fine-tuned for strict priority of mean queueing-delay between T2 and T3 $AllocIDs$, the resulting ratio, though it smooths the burstiness of voice traffic by using over-provisioned $Ratio_2$ (=0.7) value, reserves a mere 40% of the upstream capacity of XG-PON to T3 $AllocIDs$, by assigning $Ratio_3$ to 0.4. An additional assured and non-assured classification within T3 divides this 40% into even smaller portion per upstream frame when allocating $GrantSize$ for T3 $AllocIDs$ (lines 11 and 17 of Algorithm 2 in Chapter 6).

A closer look at traffic proportions in LTE upstream (Figure 7.2) indicates that the voice traffic occupies a very small percentage of the total upstream traffic in LTE compared to video traffic. Since T3 $AllocIDs$ are attempted for $GrantSize$ provisioning after all the T2 $AllocIDs$ are provisioned with $GrantSize$, the unused T2 bandwidth after serving all the T2 $AllocID$ in an upstream frame ($Unused\_T2\_BW = \sum_{i=1}^{N} AB_{2,i} - \sum_{i=1}^{N} GrantSize_{2,i}$), can be taken advantage of, by the T3 $AllocIDs$.

By eliminating the need for a down counter in T3, a strictly alternating pattern is introduced in XGIANT-D and XGIANT-P, during every other upstream, to differentiate between the assured and non-assured portion of T3. This results in the following alternating maximum allocation bytes ($AB_{3a,i}$ or $AB_{3b,i}$) for each T3 $AllocID$ every other upstream:

$$AB_{3a,i} = (Unused\_T2\_BW + \lambda_3 * Ratio_3 * C_{XG-PON})/N_k \qquad (7.2)$$

$$AB_{3b,i} = (Unused\_T2\_BW + (1 - \lambda_3) * Ratio_3 * C_{XG-PON})/N_k \qquad (7.3)$$

where λ_{T_3} = ratio of assured and non assured portion in T3 = 0.4 in XGIANT.

Thus, the eNB-aggregated video traffic's burstiness is smoothed by the exploitation of $Unused\_T2\_BW$ in the upstream for T3 $AllocIDs$.

7.4.3 Improving T4 $GrantSize$ allocation

The $GrantSize$ provision of T3 $AllocIDs$ using alternating $AB_{3a,i}$ or $AB_{3b,i}$ values every other upstream, gives two highly varying, yet discrete set of values for the total unused bandwidth in XG-PON upstream after provisioning for all the T2 and T3 $AllocIDs$ (= tot_unused_BW). Taking advantage of this adaptive nature of tot_unused_BW values, the following two independent DBA policies are proposed for T4 $GrantSize$ provision, hence two variations of improved XGIANT in XGIANT-D and XGIANT-P. Algorithms 3 and 4 gives the manner XGIANT-D and XGIANT-P, assigns $GrantSize$ for only the T4 $AllocIDs$, respectively, by replacing the equivalent XGIANT implementation (lines 23 - 33) in Algorithm 2 given in Chapter 6. ODD_UPSTREAM_FRAME is a boolean value which alternates its state at the beginning of BW_Allocation_Main(), every upstream frame.

Algorithm 3 : Calculate_GrantSize ($AllocID_{4,i}$) in XGIANT-D

This code forms lines 23-33 of Algorithm 2 in Chapter 6. Before initialisation of this code:
- *tot_unused_BW   (=   UPSTREAM_FRAME_SIZE   - $\sum_{i=1}^{N} GrantSize_{2,i}$ - $\sum_{i=1}^{\overline{N}} GrantSize_{3,i}$) is calculated once per upstream frame.*
- *ODD_UPSTREAM_FRAME is a boolean value, flipped for its state at the beginning of BW_Allocation_Main(), every upstream frame*

1: **if** ODD_UPSTREAM_FRAME == true **then**   ▷ ($AB_{3a,i}$ allocated upstream frame)
2:     $threshold_d = \frac{tot\_unused\_BW}{no\_of\_T4\_unserved}$
3:     deficit ($AllocID_{4,i}$) = max{0, (request_size ($AllocID_{4,i}$) - $threshold_d$)}
4:     tot_deficit_T4 += deficit ($AllocID_{4,i}$)
5: **else**   ▷ ($AB_{3b,i}$ allocated upstream frame)
6:     **if** (deficit ($AllocID_{4,i}$) > 0) **then**
7:         $threshold_d = \frac{tot\_unused\_BW + tot\_deficit\_T4}{no\_of\_T4\_unserved}$
8:     **else**
9:         $threshold_d = \max\{0, \frac{tot\_unused\_BW - tot\_deficit\_T4}{no\_of\_T4\_unserved}\}$
10:     **end if**
11: **end if**
12: no_of_T4_unserved = no_of_T4_unserved - 1
13: grant_size = min {request_size ($AllocID_{4,i}$), $threshold_d$}

- Deficit policy in XGIANT-D (Algorithm 3): A dynamic threshold ($threshold_d$) is introduced here to vary the bandwidth allocated to each best-effort $AllocID$. During every odd upstream frame (when ODD_UPSTREAM_FRAME is true), the $threshold_d$ merely relies on the tot_unused_BW and total number of unserved T4 $AllocIDs$ (no_of_T4_unserved), resulting in a deficit ($deficit(AllocID_{4,i})$), due to the burstiness of the best-effort traffic. As in a deficit round robin scheduler, the deficit is used in the subsequent upstream frame (if ODD_UPSTREAM_FRAME = false), to dynamically adjust the $threshold_d$, so that best-effort $AllocIDs$ with highly bursty traffic are given more bandwidth than the non-bursty ones (line 7 and 9 in Algorithm 3).

Algorithm 4 : Calculate_GrantSize ($AllocID_{4,i}$) in XGIANT-P

This code forms lines 23-33 of Algorithm 2 in Chapter 6. Before initialisation of this code:
- *tot_unused_BW (= UPSTREAM_FRAME_SIZE - $\sum_{i=1}^{N} GrantSize_{2,i}$ - $\sum_{i=1}^{N} GrantSize_{3,i}$) is calculated once per upstream frame.*
- *ODD_UPSTREAM_FRAME is a boolean value, flipped for its state at the beginning of BW_Allocation_Main(), every upstream frame*

1: **if** ($AllocID_{4,i}$ == first T4 AllocID served in this upstream frame) **then**
2:     BF= max{ 1, $\sqrt{tot\_T4\_Request/tot\_unused\_BW}$ }
3: **end if**
4: burst_request = $\frac{BF*tot\_unused\_BW*request\_size(AllocID_{4,i})}{tot\_T4\_Request}$
5: grant_size ($AllocID_{4,i}$) = min{request_size ($AllocID_{4,i}$), burst_request }

- Proportional policy in XGIANT-P (Algorithm 4): Due to the adaptive nature of the deficit policy in XGIANT-D with regard to the $threshold_d$, the $AllocID$s served at the end of every upstream frame are favoured for more $GrantSize$ that the ones served at the beginning due to reducing no_of_T4_unserved. Hence, in the proportional policy, a dynamic Burst Factor (BF) is introduced to indicate the burstiness of all the best-effort $AllocID$s in every upstream frame. BF ensures that the $GrantSize$ allocation to each $AllocID_{4,i}$ in every upstream frame is impacted by a weighted DBRu (line 4 in Algorithm 4), as in a weighted round robin scheduler (see section 2.4 of Chapter 2 for details on schedulers in communication networks).

7.4.4 Simplifying the allocation cycle procedure

Both XGIANT and EBU followed the same procedure of serving $AllocID$s in an allocation cycle, following the original proposal of the same in GIANT. In an allocation cycles, both XGIANT and EBU, first served all the T2 $AllocID$s, then served all the T3 $AllocID$s for assured portion, served all the T4 $AllocID$s for queue occupancy reports, served all the T3 $AllocID$s again for non-assured portion and finally served all the T4 $AllocID$s for $GrantSize$ allocation.

However, with the introduction of strictly alternating $GrantSize$ allocation procedure every other upstream frame and the elimination of down counters for T3 and T4 in the earlier improvements (sections 7.4.2 and 7.4.3), the procedure for visiting $AllocID$s in an allocation cycle for XGIANT-D and XGIANT-P are simplified. That is both XGIANT-D and XGIANT-P, in an allocation cycle, first serves all the T2 $AllocID$s, then serves all the T3 $AllocID$s (either based on $AB_{3a,i}$ for assured portion or $AB_{3b,i}$ for non-assured portion) and finally serve T4 $AllocID$s based on the deficit or proportional policy of XGIANT-D or XGIANT-P respectively. Hence, while T1 and T2 $GrantSize$ allocation policies in XGIANT (lines 2-4 and 5-8 respectively in Algorithm 2) remain

---

Algorithm 5 : Calculate_GrantSize($AllocID_{k,i}$) in XGIANT-D/P

This algorithm is a direct replacement of lines 9-33 of Algorithm 2 in Chapter 6.

 1: **if** (TYPE == 3) **then**
 2:     **if** (ODD_UPSTREAM_FRAME == true) **then**
 3:         grant_size = min { request_size($AllocID_{3,i}$), $AB_{3a,i}$ }
 4:     **else**
 5:         grant_size = min { request_size($AllocID_{3,i}$), $AB_{3b,i}$ }
 6:     **end if**
 7:     Shape_Grant_For_Standard(grant_size,$AllocID_{3,i}$)
 8: **end if**
 9: **if** (TYPE == 4) **then**
10:     Execute Algorithm 3 for XGIANT-D or Algorithm 4 for XGIANT-P
11:     Shape_Grant_For_Standard(grant_size,$AllocID_{4,i}$)
12: **end if**
13: **return** grant_size

---

similar in XGIANT-D/P, T3 and T4 allocation policies (equivalent to lines 9-22 and 23-33 respectively in Algorithm 2) are changed as in Algorithm 5.

7.4.5 Fairness

Due to the unpredictable bursty nature of aggregated voice, video and best-effort applications and the round-robin $GrantSize$ allocation within each T-CONT type, unfair allocation of $GrantSize$ to $AllocID$s of similar T-CONT types is inevitable. Considering the independent nature of each traffic profile, an intra-TCONT-type fairness policy is proposed in XGIANT-D and XGIANT-P, as opposed to the inter-ONU fairness in EBU, to provide fair $GrantSize$ allocation to all $AllocID$s of T2, T3 and T4, individually. In intra-TCONT-type fairness, the first served $AllocID$ of each T-CONT type in the entire XG-PON network is altered in a round-robin manner, independent of the rest of the T_k. For example, when in a given upstream frame, the 1st $AllocID$ of T1, 8th of T2, 3rd of T3 and 15th of T4 are served first, in the subsequent upstream frame, 2nd T-CONT of T1, 9th of T2, 4th of T3 and 16th of T4 are served first.

Algorithm 6 represents the pseudo-code of the function Check_All_AllocIDs_Served ($AllocID_{k,i}$) (line 8 of Algorithm 1 in Chapter 4) which implements the intra-TCONT-type fairness in XGIANT-D and XGIANT-P. Here, two variables are used for each T-CONT type to keep track of the first and last served $AllocID$s in the T-CONT type. Each index is first initialised to a default value at the beginning of XG-PON operation, with:

$$For\ k = 1:4,\ first\_served\_t_k\_index = last\_served\_t_k\_index = k$$

Within the Algorithm, first_served_t_k_index and last_served_t_k_index (for k = 1:4) are reassigned without inter-T-CONT-type dependency of the indexes so that for each

---

Algorithm 6 : Check_All_AllocIDs_Served ($AllocID_{k,i}$) in XGIANT-D and XGIANT-P

This function selects the index for 1st $AllocID$ in the next T-CONT type to be served in the current upstream frame. A $true$ return of this function terminates the BW_Allocation_main() (Algorithm 1) while a $false$ return continues with assigning $GrantSize$ to rest of the $AllocIDs$ in the same upstream frame

At XG-PON initialisation: For k=1:4, last_served_t_k_index = first_served_t_k_index = k, since index of AllocIDs served is from 1 to ALL_ALLOCID.Size(); ALL_ALLOCID vector contains all the AllocIDs configured in the XG-PON network, in the order of adding ONUs in the network.

 1: TYPE = Get_TCONT_Type ($AllocID_{k,i}$);
 2: last_served_t_k_index + = 4   ▷ AllocIDs of same T-CONT type are situated 4 indexes away when each ONU is configured with an AllocID for each of the four T-CONT types (T1, T2, T3 or T4)

 3: **if** (last_served_t_k_index > ALL_ALLOCID.Size()) **then**
 4:     For k=1:4, last_served_t_k_index = k;   ▷ reset if the index is out of scope
 5: **end if**

 6: **if** (last_served_t_k_index == first_served_t_k_index) **then**
 7:     **if** (TYPE ≠ 4) **then**   ▷ If T-CONT type == T1/T2/T3
 8:         TYPE += 1   ▷ select the next T-CONT type
 9:         first_served_t_k_index = last_served_t_k_index   ▷ update index for fairness
10:     **else**   ▷ After all T4 AllocIDs are served
11:         For k=1:4, last_served_t_k_index = k   ▷ indexes are reset
12:         **return** true   ▷ and BW_Allocation_Main() is allowed to exit
13:     **end if**
14: **end if**
15: **return** false

T-CONT type, last_served_t_k_index from this upstream frame is used in the next upstream frame, at the Pick_Next_AllocID_To_Serve() function (line 11 of Algorithm 1 of Chapter 4) to maintain intra-TCONT-type fairness.

7.5   Queuing-delay Performance of XGIANT-D and XGIANT-P

This section presents the mean queuing-delay performance of XGIANT-D and XGIANT-P for the under-loaded (52-eNBs) and marginally over-loaded (80-eNBs) scenarios explained in section 7.2.5. Discussions with regard to the strict priority and fairness metrics are also presented based on the mean queuing-delay performance of the DBAs (see section 7.3.1).

Figures 7.7 and 7.8 show the mean queuing-delay performances of XGIANT-D and XGIANT-P for the 52-eNB and 80-eNB scenarios respectively. Again, each of the 20-seconds-long experiment is repeated for the same 10 seeds as in section 7.3.2, with
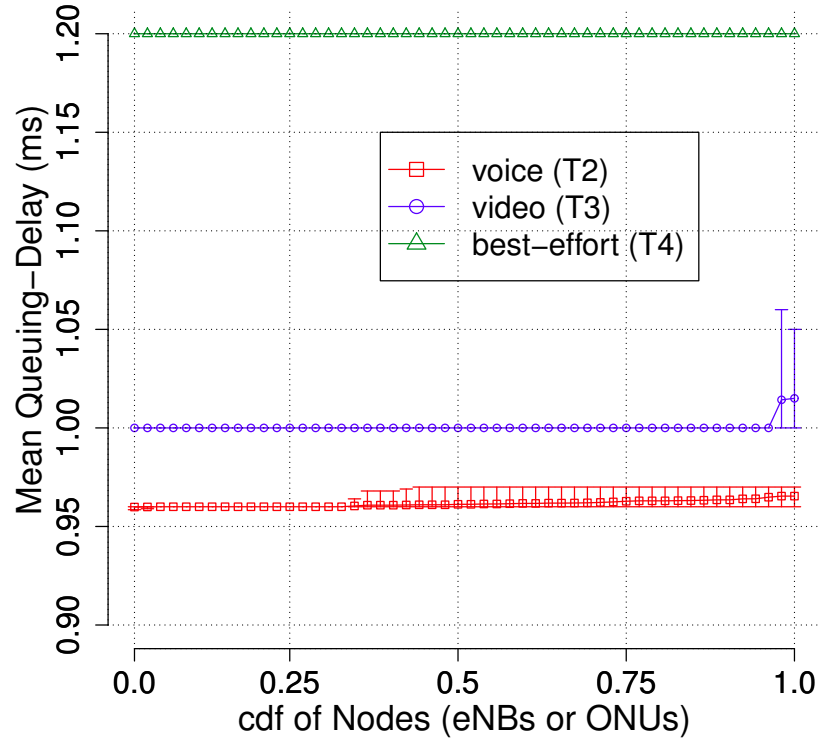
(a) XGIANT-D
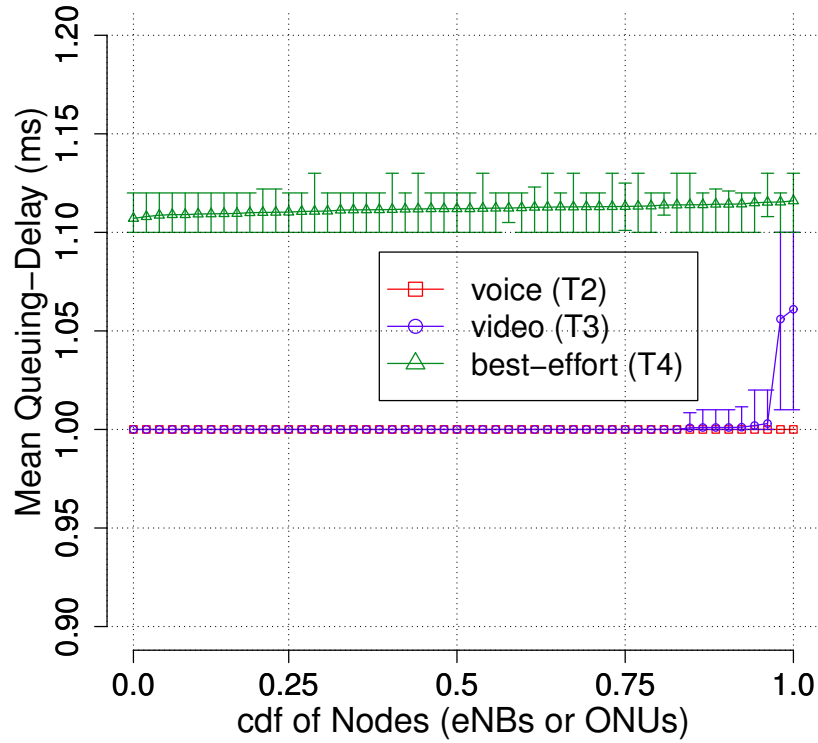
(b) XGIANT-P

Figure 7.7: Mean queuing-delay in 52-eNB scenario

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE

7.5 Queuing-delay Performance of XGIANT-D and
XGIANT-P
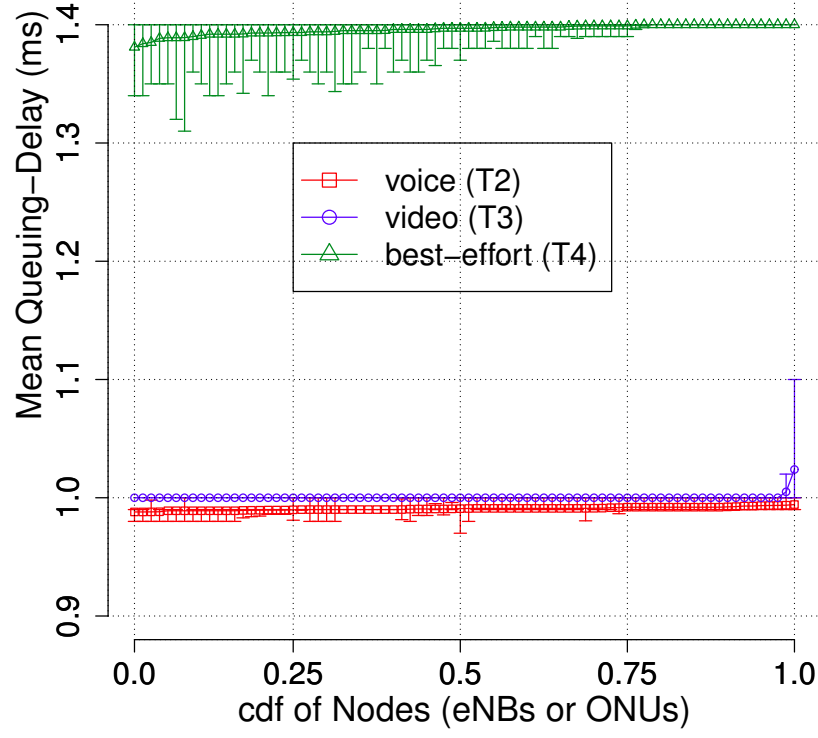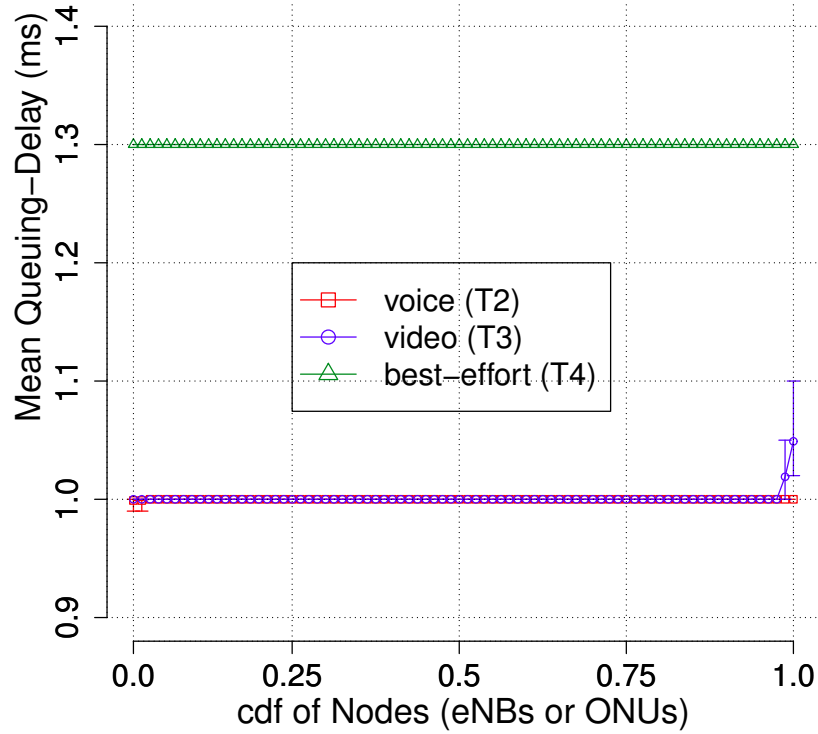
(a) XGIANT-D

(b) XGIANT-P

Figure 7.8: Mean queuing-delay in 80-eNB scenario

each ONU having 5KB for each of its T-CONT queues (T2, T3 and T4) and the ONU-OLT propagation delay set to 0.4ms. Each value in the y-axis indicates the average of mean queuing-delays observed for the 3 T-CONT types, while the error bars indicate the range of the mean queuing-delays over all the 10 runs; the x-axis represent the CDF with regard to the number of eNBs in each scenario.

All four figures show that both XGIANT-D and -P respect the priority between all the $AllocIDs$ of all the T-CONT types, due to the relative over-provision of MDR_{T2} against MDR_{T3} and the dual stage $GrantSize$ allocation for T3 (BW_{g1} or BW_{g2}). Between the two DBAs, XGIANT-D DBA provides lowest mean queuing-delay for voice (T2), while compromising on the mean queuing-delay for best-effort (T4) traffic. XGIANT-P provides lower mean queuing-delays for best-effort (T4), in both the scenarios, while maintaining (joint-)lowest delay for voice (T2) and video (T3) $AllocIDs$. This is because XGIANT-D can never over-provision per-upstream-frame bandwidth due to tot_unused_BW calculated every upstream frame. As a result, while the T4 $AllocIDs$ receive smaller $GrantSize$ in XGIANT-D than in XGIANT-P resulting in T4 $AllocIDs$ receiving higher-than-XGIANT-P mean queuing-delays while T2 $AllocIDs$ receiving lower-than-XGIANT-P mean queuing-delays. Use of the dynamic BF in XGIANT-P results in possible over-provision and therefore larger $GrantSize$ for T4 $AllocIDs$ when upstream load ratio is greater than 1.0; therefore, for both 52 and 80-eNB scenarios, XGIANT-P provides lower mean queuing-delay for best-effort (T4) than XGIANT-D, when XG-PON is instantaneously over-loaded (upstream load ratio > 1.0)

Compared to XGIANT-D, XGIANT-P provides better fairness in mean queuing-delay between all the $AllocIDs$ in each T-CONT type individually, due to its controlled over-provision of $grant\_sieze$ to the T4 $AllocIDs$. XGIANT-P also provides consistent mean-queuing delays for all the T2 $AllocIDs$ and T4 $AllocIDs$ across all the seeds, thereby indicating its robustness against mean queuing-delay variation (or jitter) across per-eNB load variation in the 10 seeds.

Comparisons of these mean queuing-delay values against those of XGIANT and EBU show that both XGIANT-D and XGIANT-P perform better in terms of: 1) strict priority, by ensuring all the lower priority $AllocIDs$ are experiencing higher mean queueing-delay value than any of the higher priority $AllocIDs$, in all the 10 seeds and 2) fairness, by showing a very small range of mean queuing-delay values for all the $AllocIDs$ in each of the T-CONT type due to employing an improved fairness policy (intra-TCONT-type).

Overall, XGIANT-D and XGIANT-P ensure a maximum mean queuing-delay of ∼1ms for voice and video and a higher but < 1.5ms for best-effort, when T-CONT queue values are equal to half the Bandwidth Delay Product (=5KB). These values prove the ability of the improved DBAs to satisfy the literature-recommended two-way delays in the LTE backhaul (10ms in [75], 5ms in [44] and 10ms in [83]) while ensuring strict priorities between and fairness for all 3 aggregated applications.
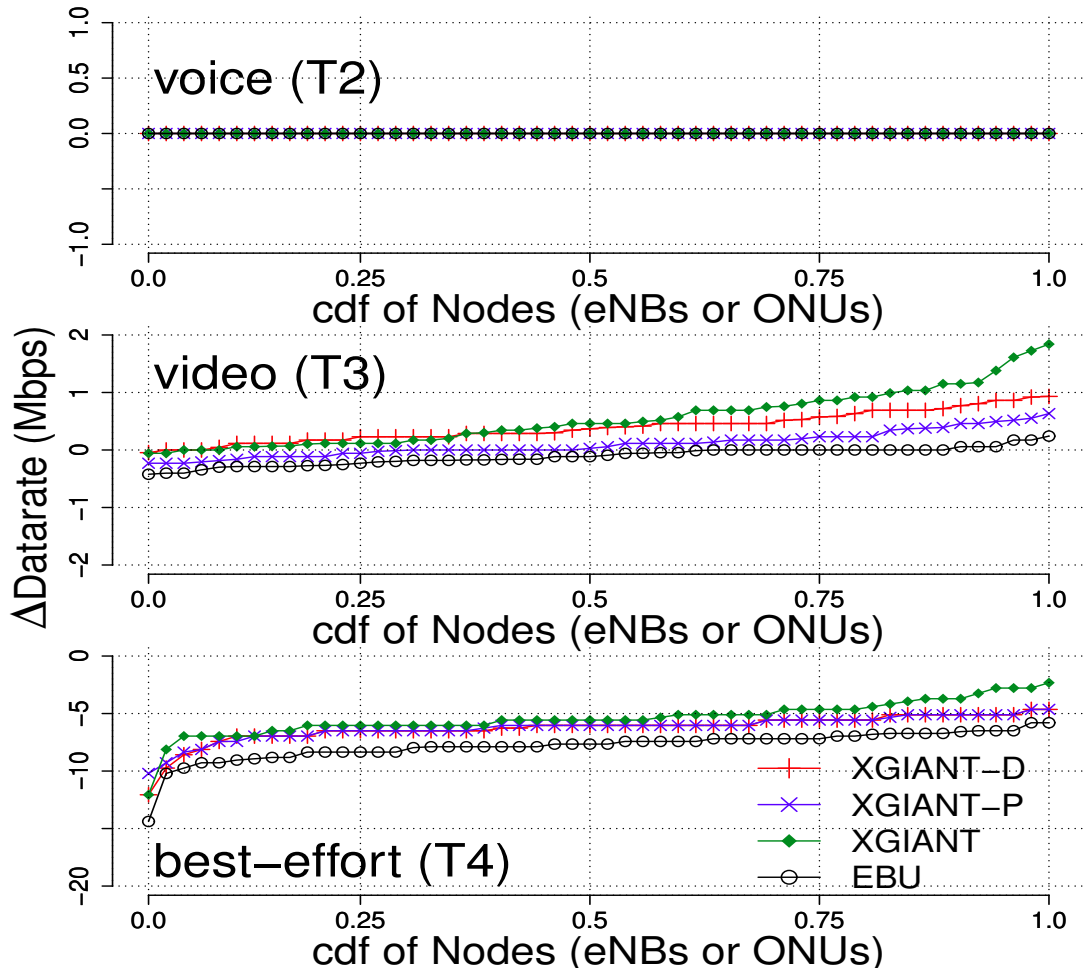
Figure 7.9: Difference in mean datarate in 52-eNB scenario

7.6 Additional Metrics of Evaluation

Earlier sections established the suitability of the improved DBAs, XGIANT-D/P, and the adaptability of their QoS policies in the XG-PON backhaul in LTE for the three application types in terms of mean queueing-delay performance. To further validate the behavioural pattern of the improved DBAs as well as that of XGIANT and EBU in the converged network, this section presents the evaluation of three additional metrics, namely the mean datarate, queueing-jitter and burstiness of each eNB-aggregated LTE application across the XG-PON backhaul.

7.6.1 Datarate assurance by the DBAs

To present the datarate (throughput) performance of XGIANT-D and XGIANT-P in the XG-PON backhaul of LTE, the differences in mean instantaneous (per-millisecond measured) datarate for aggregated applications in the 52-eNB (Figure 7.9) and 80-eNB scenarios (Figure 7.10) are plotted.

7. Design, implementation and evaluation of
   suitable DBAs for the dedicated XG-PON
   backhaul in LTE   *7.6 Additional Metrics of Evaluation*
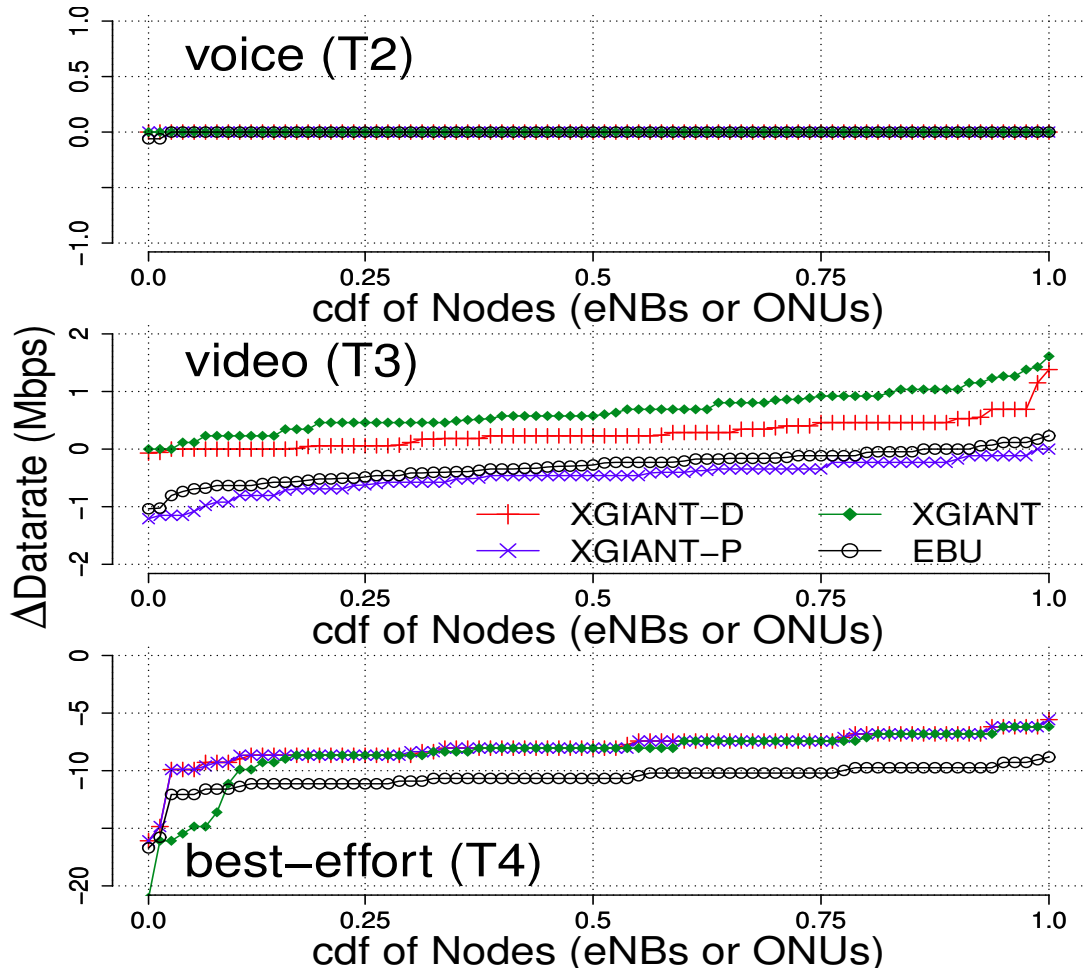
Figure 7.10: Difference in mean datarate in 80-eNB scenario

ΔDatarate in the figures indicates the differences in instantaneous (per-millisecond measured) mean datarate between traffic transmitted across XG-PON and that aggregated at the eNB. A zero value indicates that the DBA provisions the exact eNB-aggregated mean instantaneous datarate across XG-PON; a negative value indicates packet loss and positive value shows expedited datarate in XG-PON due to the extended holding of packets in T_k buffers. Each CDF line represents all the values from all 10 runs, as each value in ΔDatarate is a possible value associated with each DBA.

Figures 7.9 and 7.10 show that while all DBAs ensure lossless bandwidth provision for voice, video traffic shows a mix of expedited and lost datarate and best-effort suffers the highest loss of datarate from all DBAs. Expedited video traffic by all DBAs is due to dual-stage $GrantSize$ allocation: XGIANT shows heavy expediting of traffic due to having a very high value for MDR_3 (=0.67*$Capacity_{XG-PON}$) while EBU shows heavy loss in video datarate due to its extended holding of T3 traffic in the small 5KB T-CONT queue for 5 or 10 upstream frames, resulting in more packet drops between subsequent $GrantSize$ allocation cycles. Between the two optimised DBAs, XGIANT-D
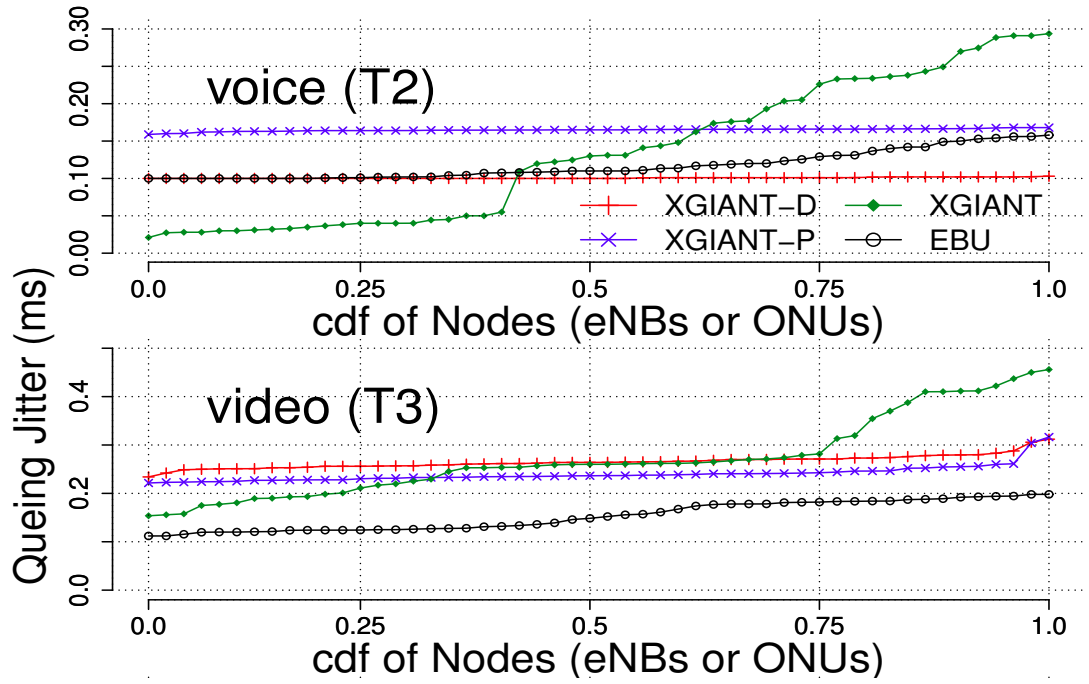
Figure 7.11: Queuing-jitter in 52-eNB scenario

shows no loss of video traffic in both scenarios while XGIANT-P shows a certain degree of voice traffic loss across XG-PON due to the absence of or use of T4 over-provision in the DBAs, respectively.

For best-effort, EBU shows highest datarate drop (packet loss) in both the scenarios, again because of EBU provisioning $GrantSize$ every 10th upstream frame; XGIANT-D and XGIANT-P are able to outperform XGIANT with lowest loss in the 80-eNB scenario, though moderately lossy in the 52-eNB scenario, due to employing adaptive QoS policies for T4 $GrantSize$ allocation.

Overall, XGIANT-D and XGIANT-P also ensure fairer mean datarate loss (CDF plot with less range in y axis) for the entire LTE backhaul, validating the combined influence of the improved QoS and fairness policies.

7.6.2 Queuing-jitter

Due to the real-time nature of voice and video application types, the evaluation of queuing-jitter will provide a good understanding of the E2E jitter the real-time aggregated application can experience in LTE backhaul. Since every transmitted packet in XG-PON upstream is measured for T-CONT queuing-delay, the inter-quartile range (difference between 75th and 25th percentile) of queuing-delay of the packets in an $AllocID$'s T-CONT buffer is used for the representation of queuing-jitter in LTE backhaul.

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE   7.6 Additional Metrics of Evaluation
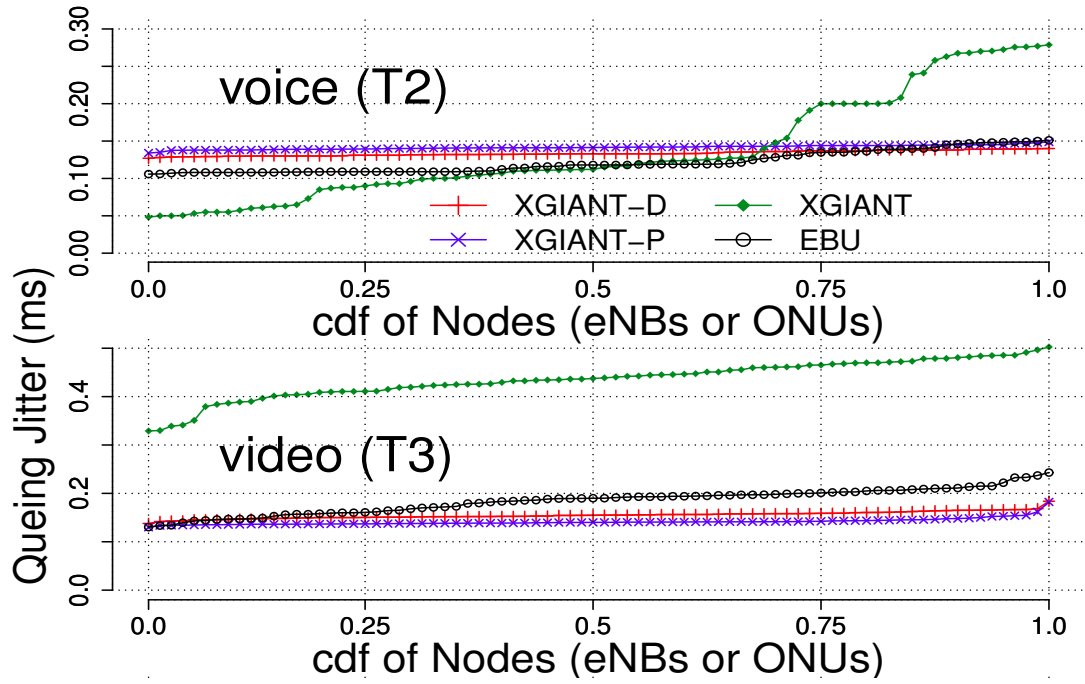
Figure 7.12: Queuing-jitter in 80-eNB scenario

Figure 7.11 and 7.12 show the queuing-jitter introduced by all the DBAs for the real-time applications, voice and video. Two aspects of consideration in the figures are 1) the abstract value of the queuing-jitter by a DBA; for a given application, a lower value of queuing-jitter would indicate a lower queuing-jitter experienced by each eNB and 2) the range of queuing-jitter values for a given application by a DBA; a horizontal line in the CDF plot would, therefore, indicate the ability of the DBA to provide a uniform queuing-jitter across the different eNBs for the same application type in LTE backhaul.

In both scenarios, all DBAs except XGIANT provides uniform queuing-jitter for both voice and video traffic. Lack of a fairness policy in XGIANT provides a very large range of values for the queuing-jitter. EBU is able to provide smoothed queueing-jitter values equivalent to that of XGIANT-D and XGIANT-P by employing long allocation cycles than XGIANT and having a fairness policy; specifically, the video traffic is benefited for queuing-jitter from the combined policies of EBU than from any other DBA in under-loaded (52-eNB) scenario.

Among all the DBAs, XGIANT-D and XGIANT-P provide more uniform queuing-jitter values across the entire LTE backhaul, for both voice and video applications due to the impact of the improved QoS policies on the intra-TCONT-type fairness policy; EBU however provides lowest queuing-jitter values in some scenarios (for video in 52-eNB scenario and voice in 80-eNB scenario) because of its longer allocations cycles (5, 10 upstream frame) taking advantage of the instantaneous burstiness of the applications.

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE

7.6 Additional Metrics of Evaluation
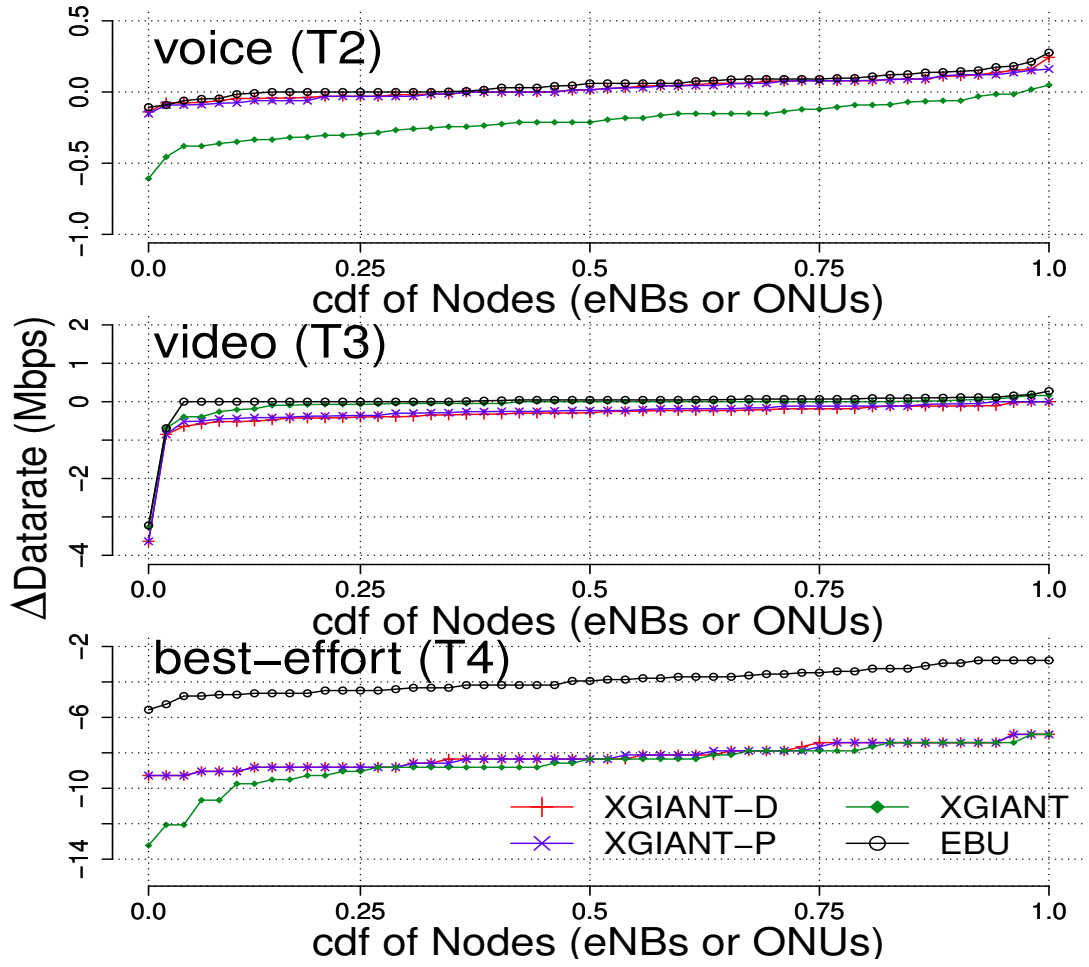
Figure 7.13: Difference in datarate burstiness in 52-eNB scenario

7.6.3 Burstiness of Datarate

Burstiness in datarate is the difference between the highest and lowest values obtained in an instantaneous datarate. Due to the provision of very large Ethernet link-rate between eNB and ONU in accommodating non-throttling upstream bandwidth (compared to the upstream capacity of each eNB in ns-3), the measured instantaneous datarate values range from 0 to very large values such as 200Mbps for eNB-ONU link. Hence inclusion of both the bottom and top extreme values of instantaneous datarate can suppress the actual impact of instantaneous datarate (burstiness) on the mean datarate over longer time frames. Hence, the inter-quartile range of the per-millisecond-measured instantaneous datarate is used to represent the burstiness of traffic between eNB-ONU and ONU-OLT links, individually.

Figures 7.13 and 7.14 show the difference in burstiness caused by the DBAs. That is the ΔDatarate in the figures show the difference in burstiness across XG-PON as compared to the burstiness of the applications experienced in entering the T-CONT buffers (or

7. Design, implementation and evaluation of
suitable DBAs for the dedicated XG-PON
backhaul in LTE

7.6 Additional Metrics of Evaluation
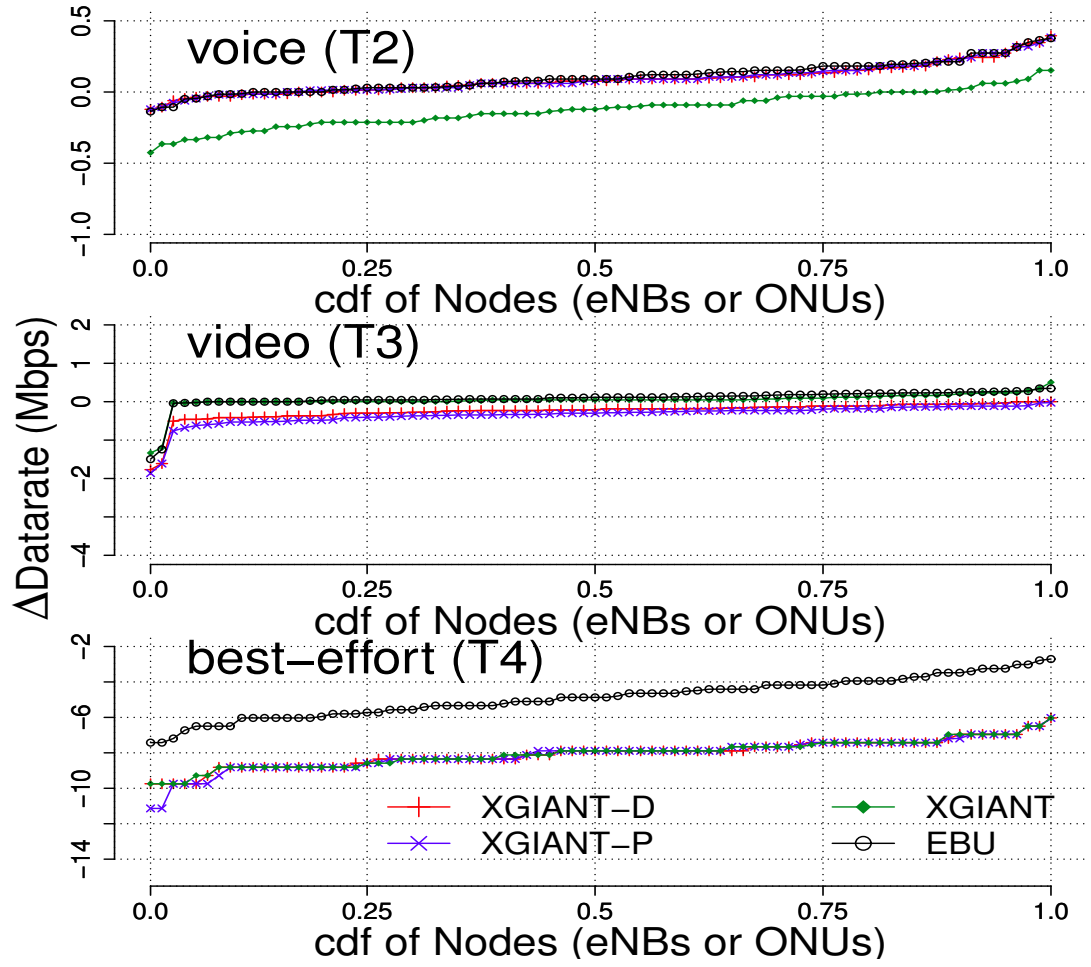
Figure 7.14: Difference in datarate burstiness in 80-eNB scenario

uploaded by the eNB). A positive value therefore indicates added burstiness in the XG-PON-based LTE backhaul; a negative value indicates smoothed burstiness and a zero value indicating the burstiness of the application being unaltered across XG-PON.

In both scenarios, XGIANT provides the best burstiness for voice smoothing among all the DBAs due to having smaller (than EBU) allocation cycles, though at the expense of having a highly unfair queueing delay as seen for the voice traffic in Figures 7.11 and 7.11. For the video traffic however, the dual-stage bandwidth allocation has resulted in burstiness remaining unaltered between XGIANT and its improved versions. Though the improved DBAs fail to be equivalent to XGIANT in terms of burstiness smoothing for voice traffic, they come close to the behaviour of XGIANT for voice and best-effort traffic in the 80-eNB scenario, indicating the impact of the improved QoS policies in marginally over-loaded XG-PON upstream scenarios.

7. Design, implementation and evaluation of
   suitable DBAs for the dedicated XG-PON
   backhaul in LTE

7.7 Conclusion

7.7 Conclusion

This chapter presented the design, implementation and evaluation of two suitable DBAs which provide differentiated QoS treatment for different classes of upstream LTE traffic in the XG-PON backhaul. After providing the implementation details of the standards-compliant converged network architecture for and QoS-metric conversion scheme between XG-PON and LTE, the chapter presented the performance evaluation of XGIANT and EBU DBAs (which were originally designed for the stand-alone XG-PON) in the XG-PON backhaul of LTE, where both DBAs indicated their inability to provide prioritised and fair mean queuing-delay for the under-loaded and marginally fully-loaded upstream conditions in the XG-PON-based LTE backhaul. By improving the QoS policies of XGIANT (and EBU), this chapter demonstrated that the resulting DBAs, namely XGIANT-D and XGIANT-P, are able to provide prioritised and fair queuing-delay performances for the same loading conditions in the XG-PON-based LTE backhaul for all three types of upstream application traffic in LTE. XGIANT-D and XGIANT-P also showed improved performance in the converged network architecture in terms of queuing-jitter, datarate and burstiness (in datarate).

Since this chapter presented the performance evaluation of all four DBAs when serving UDP-based application traffic in LTE, and TCP is the dominant Transport protocol in the Internet, the next chapter presents the performance evaluation of the TCP-based upstream application traffic of LTE across the XG-PON backhaul, when served by each of the four DBAs, individually.

Chapter 8

Performance evaluation of TCP-based upstream traffic in LTE across the XG-PON backhaul

This chapter evaluates the performance of TCP-based realistic upstream applications in LTE when individually serviced by each of the four QoS-aware DBAs (XGIANT-D, XGIANT-P, XGIANT and EBU) in the context of a dedicated XG-PON backhaul in LTE.

Chapter 6 established the standard-compliant nature of the XGIANT and EBU DBAs in the context of a stand-alone XG-PON. However, the evaluations presented in Chapter 7 indicated the inability of XGIANT and EBU to provide standard-compliant QoS assurances in the context of a dedicated XG-PON backhaul in LTE. Chapter 7 also presented the design and evaluations of the Deficit XGIANT (XGIANT-D) and Proportional XGIANT (XGIANT-P) DBAs both of which improved the QoS and fairness policies of XGIANT and EBU to provide better QoS assurances for realistically-generated and UDP-based upstream applications in LTE, across the XG-PON backhaul. Since UDP, unlike TCP, takes no reactive measures on the Transport Layer data transmission rate of an application generated at a Application Layer at the sender (and transmitted towards a receiver), Chapter 8 evaluated DBAs, using UDP-based applications rather than TCP-based applications, to validate the behaviour of all the DBA under non-reactive application behaviour.

However, TCP is the most popular Transport Layer protocol in the Internet. Major concerns for possible performance degradation of TCP-based applications in the dedicated XG-PON backhaul in LTE, when served by the QoS-aware DBAs, are two-fold:

- 3 distinct ranges of operational frequencies in 1) the DBA, which schedules upstream frames in the XG-PON every $125\mu s$, 2) the LTE air-interface scheduler which operates at a granularity of 1ms[3] (TTI of a subframe in LTE) and 3) TCP

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

which operates in the range of 10s of milliseconds in a typical E2E Internet access from a UE in LTE[64].

- variations in the E2E RTT at the Transport Layer, due to the influence of the prioritised QoS policies used by and multiple queues required in the ONUs for the QoS-aware DBAs in the XG-PON upstream.

In order to address these concerns, this chapter evaluates the performance of XGIANT-D, XGIANT-P, XGIANT and EBU DBAs in the context of serving TCP-based applications in an XG-PON-based LTE backhaul. First, the chapter presents the evaluations for the performance of three realistically-generated and TCP-based applications (conversational voice, peer-to-peer video and best-effort Internet), in the over-provisioned point-to-point LTE backhaul, to establish a benchmarked performance of the TCP-based application in the LTE upstream. Then similar network architecture, simulation environment and evaluation metrics are used for the XG-PON backhaul in LTE to present comparative evaluations of instantaneous throughput (datarate) and E2E RTT behaviour at the Transport Layer. Two upstream loading scenarios in the backhaul are used, each with either 36 or 52 in LTE to create a slightly under-loaded and slightly over-loaded loading condition in the XG-PON upstream, respectively. In the under-loaded scenario, XGIANT-D and XGIANT-P assure total instantaneous throughput, average instantaneous throughput and E2E RTT performance equivalent to that in the corresponding over-provisioned point-to-point backhaul for all three application types. In the over-loaded scenario, due to the instantaneous congestions caused by the aggregated upstream traffic in LTE backhaul, only the voice and video applications are assured throughput and RTT performance equivalent to that in the point-to-point backhaul; best-effort traffic shows heavily degraded throughput when compared with the point-to-point equivalent while E2E RTT indicated the additional congestion caused by the limited XG-PON upstream capacity in the LTE backhaul.

In terms of fairness of the queuing-delay for the aggregated application types at the ONU, the intra-T-CONT-type fairness policy in XGIANT-D and XGIANT-P assured reasonably fair queueing-delay performance for all three application types, while the inter-ONU fairness policy in EBU provided reasonable fairness only for the voice and video traffic types; XGIANT, without a fairness policy, failed to provide reasonable fairness for any traffic type.

The chapter first presents a brief summary, in section 8.1, of four DBAs evaluated in the chapter; detailed rationalisation for and common network architectures of point-to-point and XG-PON-based backhauls in LTE are provided in section 8.2. Section 8.3 outlines the simulation environment. Sections 8.4 and 8.5 provide the results and discussion regarding the performance the TCP-based applications in the point-to-point and XG-PON-based backhauls respectively. Section 8.6 concludes the chapter.

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

• *8.1 Summary of the DBAs for the XG-PON*
backhaul in LTE

8.1   Summary of the DBAs for the XG-PON backhaul in LTE

This section presents an overview of four DBAs, namely XGIANT, EBU, XGIANT-D and XGIANT-P, which are so far implemented and evaluated in this thesis for the context of a stand-alone and an integrated (with LTE) XG-PON.

EBU, which was proposed recently by Han et al.[35] and XGIANT, which is presented in Chapter 6 of this thesis, are two standard-compliant DBAs designed to provide QoS treatment for three different classes of traffic in the context of stand-alone XG-PON.

EBU and XGIANT were evaluated in Chapter 7 of this thesis, to validate the suitability of the DBAs in serving upstream LTE traffic (of types voice, video and best-effort) across the XG-PON backhaul. The evaluations indicated that both DBAs were unable to provide reasonable QoS and fairness assurances for the realistically generated and per-eNB-aggregated upstream traffic in the XG-PON-based LTE backhaul. Improving upon the QoS framework of XGIANT and EBU DBAs, Chapter 8 presented the design, implementation and evaluations of XGIANT-D and XGIANT-PON DBAs, both of which, proved their superiority over XGIANT and EBU, by providing low, prioritised and fair mean queuing-delay in the XG-PON backhaul of LTE, for all three types of upstream applications in LTE.

Specifically, Chapter 7 evaluated all four DBAs, using UDP-based application traffic to provide unbiased performance analysis and validation in the context of non-responsive Transport layer protocols, as UDP does not alter the traffic pattern of the applications. However, since TCP-based mobile applications dominate the Internet today and the integrated network architecture of XG-PON and LTE is a large-BDP network (see Chapter 6 for the co-existence of TCP and stand-alone XG-PON, which is also a large BDP network), it is important to evaluate the performance of the TCP-based applications when served by the QoS-aware DBAs at the dedicated XG-PON backhaul, in order to identify the potential challenges regarding the design of suitable DBAs for the LTE backhaul, in the context of realistic application environments.

8.2   Network Architecture for the evaluation of TCP-based applications in LTE backhaul

This section presents the network architectures used in the evaluations of this chapter. Two network architectures, each with either a point-to-point backhaul or XG-PON backhaul are used to provide comparative analysis for the performance of the TCP-based upstream applications in LTE in the backhaul. The DBAs are used only in the XG-PON backhaul and the point-to-point backhaul-based LTE architecture is used only as a comparative environment for the XG-PON backhaul-based LTE architecture for the

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

8.2 Network Architecture for the evaluation of
TCP-based applications in LTE backhaul

following two reasons:

1. When several users share the upstream capacity in LTE air interface, the LTE air interface itself becomes a bottleneck for TCP due to the limited capacity offered to each user by the scheduler operating at a frequency of 1ms. Since the simulation scenarios are designed to establish the performance of instantaneously congested XG-PON, XG-PON will become an additional bottleneck in the LTE network using XG-PON-based backhaul. Hence, in a large network of several thousands of UEs and more than 100 UEs per eNB, it is important to establish the throughput behaviour of aggregated application traffic and the RTT behaviour of every UE, when the air interface is the only bottleneck link for upstream LTE applications; this will ensure that when the XG-PON is used as the backhaul for LTE, the evaluation of the DBAs and the corresponding discussions can accurately identify the additional implications of introducing XG-PON as the backhaul for LTE, causing any further reactions in TCP for the E2E application behaviour.

2. Due to the presence of the $125\mu s$-granular polling-based and QoS-aware DBA in the XG-PON, per-eNB-aggregated application traffic in LTE backhaul is subjected to numerous instances of instantaneous congestion (see Figure 7.2 in Chapter 7). Since TCP is reactive to congestion introduced in the in-between network, such instantaneous congestions will prompt TCP to reduce its sending rate in the short term, for the XG-PON backhaul in LTE. Therefore it is necessary to have a reference TCP behaviour to accurately compare the impact of the DBAs in XG-PON with regard to a (instantaneously) non-congested backhaul scenario.

To achieve both these targets, it is essential to establish the throughput and RTT behaviour of TCP-based applications across an over-provisioned (for capacity) and point-to-point backhaul with air interface being the only bottleneck between the TCP endpoints. It is also necessary, therefore, to have a network topology which is both standard-compliant, realistic and capable of producing directly comparable results between the point-to-point and XG-PON-based backhaul environments.

Figure 8.1 shows the two network architectures, each with either the point-to-point or XG-PON backhaul, used in the evaluations of this chapter. The LTE network architecture with the point-to-point backhaul (Figure 8.1a) is a direct result of replacing the XG-PON from the standards-compliant integrated network architecture (Figure 8.1b) implemented in Chapter 7. That is, in both architectures, the corresponding dedicated backhaul (point-to-point or XG-PON) is placed between the eNB and the *Gateway*, which, in both the network architectures, represents the combined data-plane terminal in the core of LTE network consisting of MME, PDN-GW and SGW. In the point-to-point backhaul architecture, each eNB is connected to the *Gateway* using a point-to-point Ethernet link with sufficiently large capacity (200Mbps) compared to the maximum effective upstream capacity (= 60Mbps by means of evaluations) of the LTE eNB in
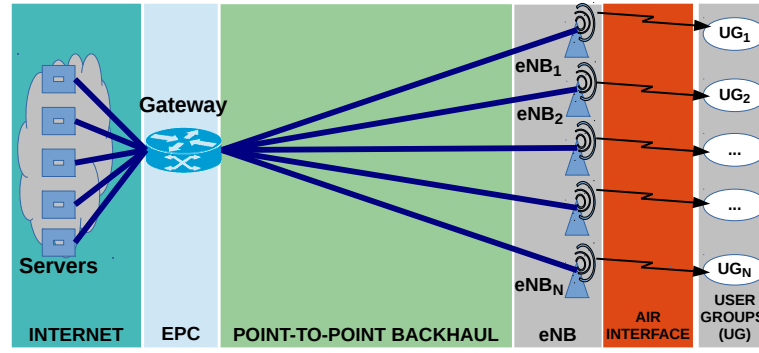
(a) LTE network with the point-to-point backhaul
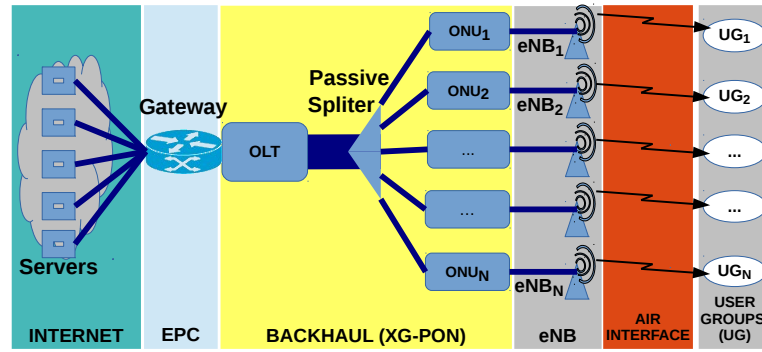
(b) LTE network with the XG-PON backhaul

Figure 8.1: LTE Network architectures with different dedicated backhauls

ns-3, so that the LTE air interface is the only bottleneck link (in terms of capacity) for the TCP-based upstream traffic in LTE; in the XG-PON backhaul architecture, both the *Gateway*-OLT and ONU-eNB links are connected via Ethernet links with zero latency and sufficiently large capacity so that XG-PON is the only additional bottleneck in this network architecture compared to the point-to-point backhaul architecture. To avoid E2E throughput throttling[38] in the upstream by the Application and Transport Layer buffers at the receiver, each UE (application generator/TCP sender) is assigned with a dedicated server (application/TCP receiver) having sufficient Application and Transport Layer buffer sizes. UEs attached to each eNB are represented by the *User Groups* in the network architectures.

Table 8.1 presents the QoS metric conversion scheme (same as in Chapter 7) that is used in these network architectures. The LTE architecture with point-to-point backhaul uses the QCI to DSCP conversion as the aggregated traffic goes from an eNB is passed on to the (carrier-grade) Ethernet link in the point-to-point backhaul; the XG-PON-based backhaul architecture uses the conversion from QCI to DSCP to the T-CONT type, as the traffic goes from an eNB, through the Ethernet link towards an ONU. Since T1 traffic will not be used in the evaluations, as in the earlier chapters of this thesis, only three radio bearers (hence 3 QCIs) are used in the air interface to differentiate the three types of applications generated in the upstream.

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

*8.3 Simulation environment for the evaluation of
TCP-based application performance in LTE*

Table 8.1: QoS metric conversion in the integrated network architecture of XG-PON and LTE

Application Type	EPS Bearer type	LTE QCI	DSCP	T-CONT Type
LTE Signalling	non-GBR	1	CS7 - 56	T1
Conversational voice	GBR	2	EF - 46	T2
Peer-to-peer video	GBR	4	CS4 - 32	T3
Best-effort ftp	non-GBR	9	BE - 0	T4

In the downstream, the network architectures assume and implement a single logical connection between each *Gateway*-eNB link for transmitting the ACK packets of all three application types. This is a reasonable assumption in the downstream since all the possible bottlenecks in the downstream have negligible influence on the delivery of ACKs because 1) the ACK packets are small in size, 2) LTE eNB in ns-3 is capable of having more than 100Mbps of downstream datarate at the air interface, 3) both the point-to-point and XG-PON backhauls have sufficiently large datarates, each, in the downstream and 4) XG-PON does not use a DBA in the downstream, thereby presenting no additional bottlenecks, in terms of latency and capacity, across the XG-PON backhaul in the downstream.

8.3 Simulation environment for the evaluation of TCP-based application performance in LTE

This section presents the common simulation environment used for the performance evaluations of the TCP-based upstream applications in LTE, when using the point-to-point and XG-PON backhauls, individually. The simulation environment includes the details of the application generation using the realistic traffic models in ns-3, the number of UEs attached per eNB and the simulation scenarios used in the evaluation.

8.3.1 Application Traffic Modelling

The following realistic traffic models are used to generate voice, video and best-effort applications in LTE upstream:

- **Voice:** The conversation voice (over IP) traffic is an ON-OFF model; the ON state generates 160 Bytes-long frames at a constant 64 kbps rate, representing common HD voice codecs (eg: G.722[48]). ON and OFF durations are exponential with a mean of 0.35s and 0.65s respectively[100].

- **Video:** To model a peer-to-peer video application having variable bit rate, Log-normal Distribution [84] with a mean frame size of 1447 Bytes and a variance of 12833 [27]) is used; frame rate was set at 29.865fps to represent an average video datarate of 345.62kbps [27].

- **Best-Effort:** Best-effort Internet traffic is modelled as FTP-based application using the $BulkSendApplication$ in ns-3 with a packet size of 512 Bytes (default value for packets size in FTP in ns-3).

All applications use TCP as their Transport Layer protocol for their data transmission.

8.3.2 Number of UEs in LTE

The number of UEs attached per eNB in the LTE network is influenced mainly by the ratio of UEs using the three applications[100, 14], with each UE generating traffic only for a single application. A ratio of 2:2:1 (as used in [100, 14] and in section 7.2.4 of Chapter 7) is used for the number of UEs generating voice:video:best-effort traffic. The total number of UEs attached to an eNB is then selected uniformly at random in the range of 105 - 145 (mean = 125), with a step of 5 in between. This ensures that the best-effort users are exact integers, uniformly distributed in the [21,29] range and have an average value of 25; this also results in an average of 50 users (in the uniformly distributed range of [42,58], with a step of 2) for the voice and video applications, individually, in each eNB.

These numbers also ensure that the aggregated (across all the UEs) average throughput per eNB for voice and video traffic is well below the effective upstream capacity of an LTE eNB in ns-3 (~60Mbps by means of exhaustive experiments for an eNB serving single best-effort UE in the upstream) such that there is substantial amount of aggregated best-effort traffic left to be utilised among all the best-effort UEs in any eNB.

All the UEs are placed randomly around the attached eNB, within a radius of 4km, while the eNBs (each with a single cell) are placed in a straight line, with an inter-eNB distance of 20km, to avoid interference at the air-interface between UEs of adjacent eNBs when using TCP-based applications. All UEs remain fixed to their position throughout the simulation, as the objective of the experiments is to evaluate the transmission of per-eNB-aggregated traffic in LTE upstream. PFS in the ns-3 is used to ensure a well-balanced throughput and fairness performance is ensured for capacity utilisation in the LTE air interface for all the UEs (see section 2.3 of Chapter 2 for more details on PFS)

8.3.3 Simulation Scenarios

To realise different upstream traffic loading in the LTE backhaul, the LTE network is configured with the following two scenarios in the simulations, such that the XG-PON upstream is:

- under-loaded with 36 eNBs in LTE and

- fully-loaded with 52 eNBs in LTE

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

*8.4 Evaluation results of TCP-based application
performance across the point-to-point backhaul in
LTE*

by means of the sum (across all the eNBs) of maximum datarate achievable per eNB (in ns-3). With ~60Mbps of maximum upstream capacity per eNB in ns-3, the 36-eNB scenario ensures that the DBAs are evaluated for influence when the XG-PON upstream is under-loaded (total upstream load in LTE backhaul < upstream capacity of XG-PON) in terms of average datarate, yet, slightly over-loaded in terms of instantaneous/bursty datarate. The 52-eNB scenario is a representation of an overloaded XG-PON-based LTE backhaul (total upstream load in LTE backhaul > upstream capacity of XG-PON) in terms of potential (based on the point-to-point backhaul behaviour only, as TCP does not transmit data more than the network capacity for long durations) for both the average and instantaneous datarates throughout the entire duration of the simulation.

Since the resulting network environment contains several thousands of UEs in LTE, it is not practically possible to begin traffic generation from all the UEs at the exact same time due to the restriction in ns-3 in simultaneously initiating data transfer from a large number of application agents. However, initiating each application at a different time (even at an interval of 1 ms) may take several 10s of simulation seconds and several days of actual time to reach network-wide convergence (steady-state) for the thousands of TCP flows in the network. Hence, a reasonable balance between the restriction in ns-3 and the duration for network-wide steady-state is achieved by initiating data transfer from all (in the entire LTE network) the voice UEs first, followed by the video UEs and finally the best-effort UEs; hence network-wide steady-state is achieved within a few simulation seconds (when using both point-to-point and XG-PON backhauls, individually) while ensuring all UEs are able to initiate data transfer without violating the restrictions in the simulator with regard to the number of application agents.

8.4 Evaluation results of TCP-based application performance across the point-to-point backhaul in LTE

This section presents the evaluations results for the performance of the TCP-based upstream applications across the point-to-point backhaul in LTE, with regard to the following two metrics:

1. **Total Instantaneous Throughput:** This is the network-wide throughput for each and all of the application type(s) across the LTE backhaul. To calculate the total (network-wide) instantaneous throughput for each application type, first, the amount of data transferred by each eNB for each application type across each of the Ethernet links in the point-to-point backhaul is measured at every 100ms interval. The total amount of data transferred across all the 36 or 52 Ethernet links is then accumulated per 100ms to represent the total instantaneous (per-100ms) throughput. 100ms measurement window remains both a large enough window to smooth the spikes in the throughput values seen for the over-provisioned Eth-

ernet links and a small enough window towards the short-term and long-term burstiness in all three applications aggregated at each eNB. Total instantaneous throughput will be used to provide discussions on the throughput performance of each application type, especially the best-effort, when served by the capacity-limited air interface in each eNB.

2. **Average RTT:** Average E2E RTT is measured at the Transport Layer using the *time stamp* option in TCP for every TCP flow in the LTE network. The resulting average RTT values for each UE during the steady-state simulation will be used to provide a reference to the two-way latency achieved by each UE of every application type, again when LTE air interface is the only bottleneck in LTE upstream.

The results presented here used the LTE network architecture in Figure 8.1a, with the following fixed one-way latencies: *Gateway* to eNB = 0.4ms to represent 60km of optical propagation delay; *Servers* to *Gateway* = 5ms to represent one-way delay between the core router at the service provider edge and the Internet servers situated close to the service provider edge. To ensure that LTE air interface is the only bottleneck link in the entire network architecture, per-UE buffer of 200KB was configured for each eNB, after performing several iterations of evaluations with different values of the buffer such as 10KB, 50KB, 100KB, 200KB and 300KB. Each experiment was simulated for a total duration of 62 seconds to ensure that each UE (and eNB) produces several TCP epochs within the simulation duration to provide statistically trustworthy values for the throughput and RTT values.Each simulation was repeated for 5 times, each with a different seed in ns-3, to vary the randomisation in the number of UEs per eNB and the application generate at each UE.

8.4.1   Total Instantaneous Throughput

Figure 8.2 presents the total instantaneous throughput for both 36-eNB and 52-eNB scenarios for a single seed of the experiment (a similar behaviour was seen across all 5 runs of each experiment). As seen, the network-wide steady-state for the LTE network with the point-to-point backhaul is achieved within less than 5 seconds of starting the simulation for both scenarios. Both voice and video traffic are (individually) able to achieve the expected total throughput values of having 36 or 52 eNBs, each with 50 UEs on average; best-effort utilises the un-utilised capacity (datarate) per eNB to reach a total instantaneous throughput close to the value of number of eNBs * 60Mbps for both scenarios. Between all three traffic types, the voice traffic show negligible congestion in the high-capacity point-to-point backhaul; the dips in the aggregated video traffic indicate that the TCP flows are subjected to scheduling in LTE air-interface, where the proportional fair scheduler periodically restricts the progressively increasing NewReno TCP flows' throughput of the aggregated video traffic request to match that of the air-interface throughput instantaneously available per UE; a correlated spike is seen for the aggregated best-effort traffic (when the aggregated video traffic throughput
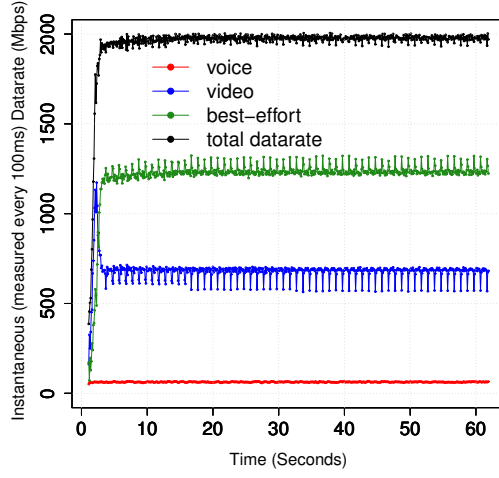
(a) 36-eNB Scenario
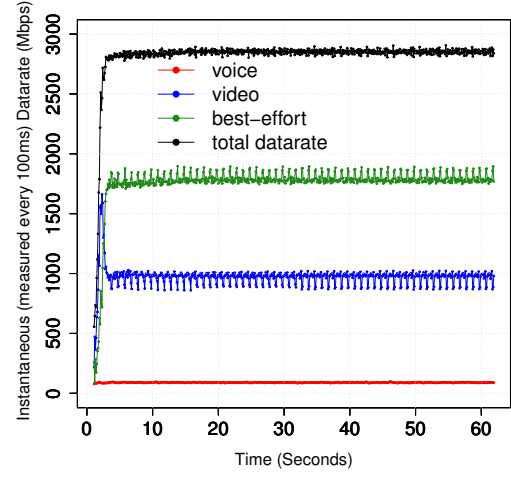
(b) 52-eNB Scenario

Figure 8.2: Total instantaneous (measured at 100ms intervals) throughput behaviour across the point-to-point backhaul in LTE
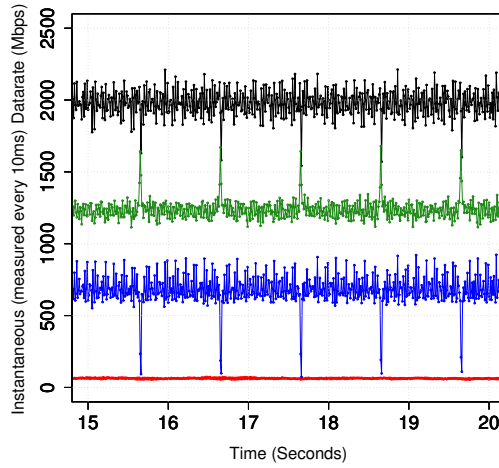
(a) 36-eNB Scenario
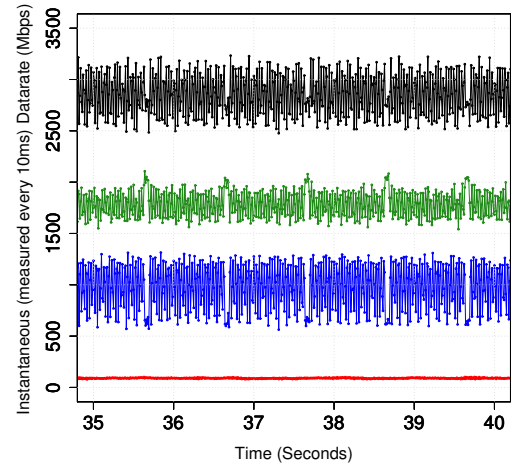
(b) 52-eNB Scenario

Figure 8.3: Snapshot of 10ms-measured total throughput behaviour across the point-to-point backhaul in LTE

dips) indicating that the proportional fair scheduling is fairly distributing the additional available air-interface throughput, if available, to the rest of the UEs; the best-effort traffic is eventually throttled by the eNB's maximum upstream capacity of 60Mbps. The complex relationship between the between the fair TCP protocol and the PF scheduler in the LTE air interface in such a large network setup, resulting in the (simultaneous) dips and spikes in the video and best-effort traffic, respectively, is clearer in Figure 8.3 where the instantaneous throughput is measured at a higher frequency of every 10ms.

Figure 8.3 also indicates that the smaller the throughput calculation interval is, the
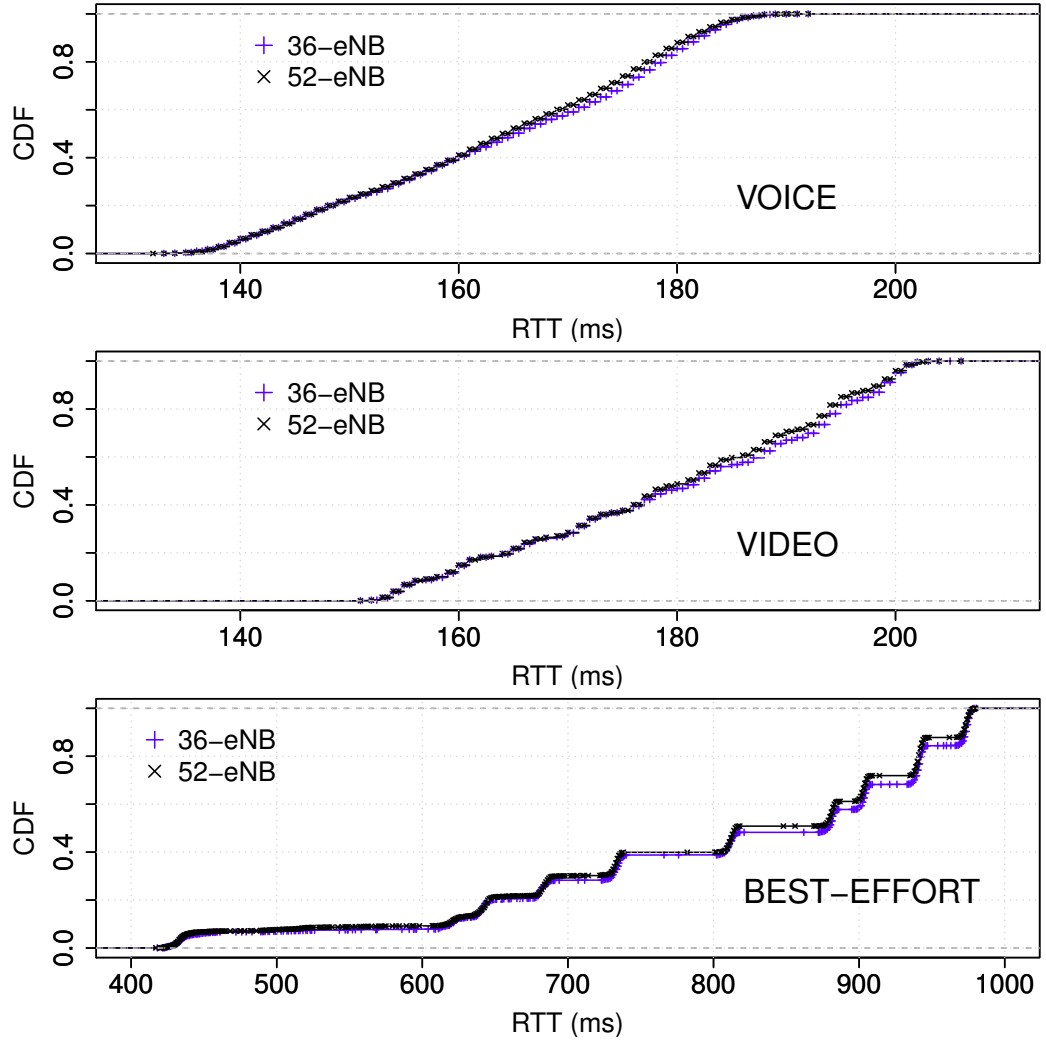
Figure 8.4: CDF of RTT values for all the UEs of voice, video and best-effort applications

more the (visibility of the) burstiness of the total instantaneous throughput for the aggregated video and best-effort traffic will be. The total instantaneously achievable throughput in the above figures also validate the rationalisation in choosing the values of 36 and 52 eNBs to evaluate the DBA performance for instantaneously congested, but on average slightly under-loaded and slightly over-loaded XG-PON (upstream) at the LTE backhaul. This ensures that the performance of the DBAs from the evaluations can be used not only to explain the challenges in the marginally filled XG-PON upstream scenarios but also to predict the (throughput and RTT) behaviour in highly under-loaded and over-loaded XG-PON upstream scenarios.

8.4.2 RTT

Figure 8.4 presents the CDF of the average RTT values experienced during the steady-state (10s - 60s) by the UEs belonging to each of the three application types. The CDF

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

*8.5 Evaluation results of TCP-based application
performance across XG-PON-based backhaul in LTE*

figures show every single RTT value across all the 5 seeds of the experiment to illustrate the range of RTT values obtained by the UEs.

UEs generating voice traffic experience the lowest average RTT values (at the Transport Layer) between 130ms and 190ms, followed by the video applications between 150ms and 210ms; those sending best-effort traffic experience very large RTT values between 420ms and 1000ms. Voice and video applications experience low average RTT values due to having defined datarates which suppress the bottleneck impact by the air interface, while best-effort applications experience relatively very high RTTs due to being throttled by the air interface capacity of each eNB.

8.5 Evaluation results of TCP-based application performance across XG-PON-based backhaul in LTE

This section presents the evaluation metrics, results and discussion with regard to the performance of TCP-based applications in an LTE network with the XG-PON backhaul.

8.5.1 Evaluation Metrics

The following 4 metrics are evaluated for each of the QoS-aware DBA in XG-PON-based LTE backhaul:

1. Total Instantaneous Throughput: This is the same metric used to represent the total instantaneous throughput across the point-to-point backhaul in section 8.4.1. Here the transmitted bytes are measured at the OLT (at the same frequency of 100ms) to present the total instantaneous throughput experienced by each and all of the application type(s) in the XG-PON upstream.

2. Total Average Throughput: This refers to the average throughput for each and all of the applications in the XG-PON upstream, during the entire steady state duration of the simulation.

3. Mean Queuing-delay: All four DBAs use three effective T-CONT queues in each ONU for collecting upstream data from the 3 application types in LTE. Finite values for the (FIFO) T-CONT queues in the ONUs inevitably introduce additional (queuing) delays and possible packet losses to the upstream LTE traffic at the ONUs, thereby influencing the E2E RTT values and throughput behaviour of each application. The resulting queuing-delay experienced only by the packets transmitted across the XG-PON, after entering the corresponding $AllocID$ queues in the ONUs is used to calculate the average (statistical mean) queueing-delay metric during the network-wide steady-state simulation, for each eNB-aggregated application type.

3. Average RTT: This is again the exact metric as in section 8.4.2, measured at the Transport Layer using the *time stamp* option in TCP for every UE, and averaged per-UE during the steady-state simulation.

8.5.2 Results and Discussion

The evaluation results and the corresponding discussions for the performance of upstream TCP-based applications in an LTE network with the XG-PON backhaul are presented here. For both (36-eNB and 52-eNB) scenarios, each evaluation was subjected to the same 5 seeds in the point-to-point backhaul environment, to provide equivalent randomisation into the number of UEs per eNB and the traffic generation at each UE.

8.5.2.1 Total Instantaneous Throughput

Figures 8.5 and 8.6 present the total instantaneous datarate experienced by voice, video and best-effort applications, in 36-eNB and 52-eNB scenarios, respectively. Both figures present only the results from the single-seeded experiment, which is sufficient to compare the relative performances among all four DBAs with regard to that seen in the point-to-point backhaul; the results from the point-to-point backhaul (for the same seed) are provided (in black color) to provide insight into relative performance of the DBAs in the XG-PON backhaul. The results from the other 4 seeds of the experiments are not given due to repetition in the discussions.

In the 36-eNB scenario (Figure 8.5), all four DBAs ensure average throughput around the similar values seen in the point-to-point backhaul environment, for all 3 application types: XGIANT-D and XGIANT-P perform exceptionally well, by provisioning the exact throughput as in the point-to-point backhaul, throughout the whole duration of (steady-state) simulation; XGIANT and EBU provide a similar throughput assurance, with an added short-range (spanning across few hundreds of milliseconds) burstiness in voice and short and long-range (spanning across few seconds) burstiness in video, due to their relative $MDR_{T2}:MDR_{T3}$ ratio; their inability to efficiently utilise the remaining transmission opportunity in the upstream after provisioning for voice and video traffic, also results in under-utilised best-effort traffic when compared to that in XGIANT-D and XGIANT-P. Specifically, EBU shows the most amount of short and long bursts in voice and video and the highest amount of instantaneous loss in best-effort. This is because of EBU holding the packets arriving in the T4 queues for several (5 or 10) cycles of US-FRAME transmission; as a result, queuing-delay (and therefore packet loss probability) increases at the T4 queues when using EBU in XG-PON, forcing TCP to reduce its transmission rate at the sender.

Figure 8.6 shows that, in the 52-eNB scenario, all the DBAs have introduced new be-
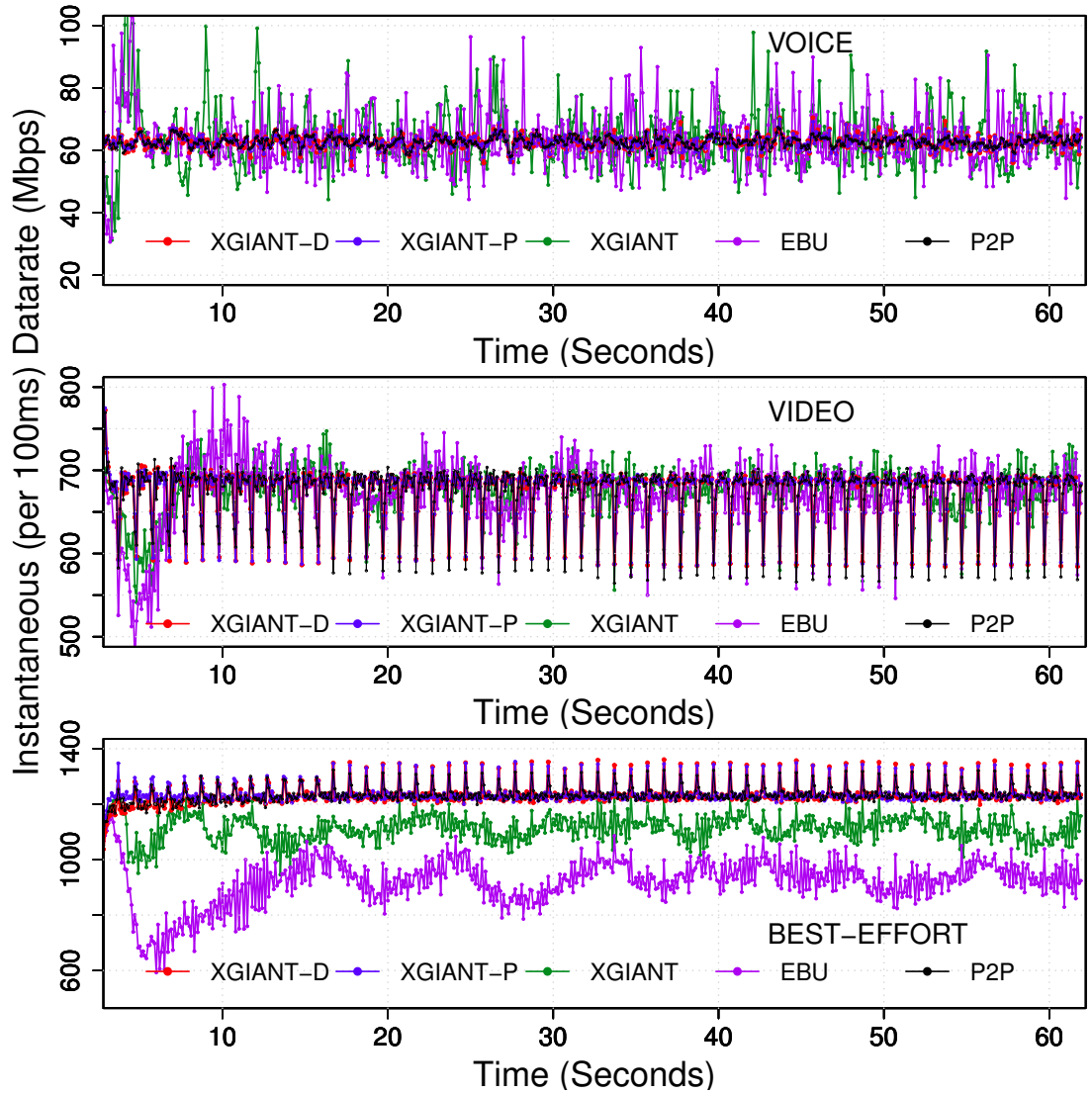
Figure 8.5: Instantaneous Throughput in the XG-PON (when using each of the four DBAs) and point-to-point (P2P) backhauls in LTE for the 36-eNB scenario

haviours for each application type with regard to the total instantaneous throughput. This is due to the instantaneous ($125\mu s$ range) congestion introduced by the QoS policies of T2, T3 and T4 in the DBAs towards the bursty application traffic in the backhaul, forcing TCP to use the Fast Retransmit and Fast Recovery mechanisms. In general, when TCP is notified of re-ordering issues, excess packet delays or packet losses in the in-between network by means of 3 duplicate ACKs, TCP assumes packet losses, trigger Fast Retransmit and Fast Recovery to maintain the congestion avoidance phase with reduced sending rate of the TCP segments. However, in the case of not getting the ACKs for the next set of packets within the RTO, TCP falls back to the slow start phase with a complete collapse in the sending rate of the segments. In the figure, all the DBAs show near-zero instantaneous throughput for voice traffic occasionally, indicating RTO in UEs sending voice traffic. The reason for this is that, in the short term DBAs
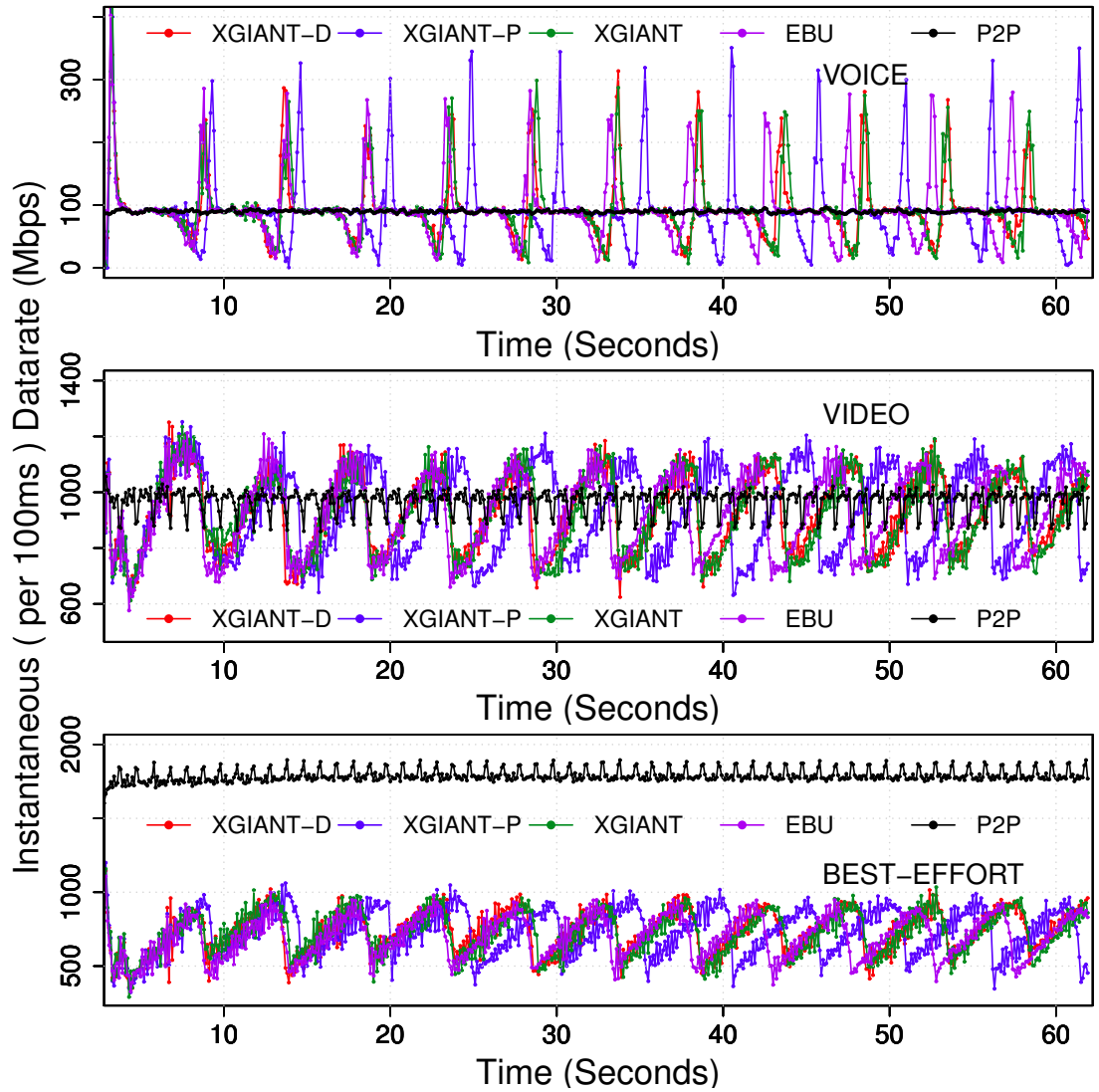
Figure 8.6: Instantaneous Throughput in XG-PON (when using each of the four DBAs) and point-to-point (P2P) backhauls in LTE for the 52-eNB scenario

require more than one $grant\_size$ allocation to clear the very highly bursty upstream voice requests from the eNBs; however, a waiting time of few milliseconds (due to two cycles of $grant\_size$ allocation) is still too small for the TCP to wait without triggering its RTO as the RTO measurement in popular congestions algorithms does not account for the duplicate ACKs[38]; it is also notable that the TCP recovers fastest in voice (with constant spikes), indicating that the T2 buffer is more than enough for the TCP to recover in time to reach expected (equivalent to the point-to-point backhaul) total instantaneous throughput while serving an average of 50 voice UEs per eNB; extensive evaluations carried out however indicated that a value less than 10KB in a T-CONT queue is too small for the voice traffic to have any effective throughput across XG-PON when XG-PON upstream is under-loaded.

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

*8.5 Evaluation results of TCP-based application
performance across XG-PON-based backhaul in LTE*

For the video and best-effort traffic, though the figures do not indicated the occurring of RTO, the effect of fast-retransmit and fast-recovery, caused by the instantaneous congestion of the aggregated throughput from all 52 eNBs can be seen from the figures. Video is still able to have an average (per-epoch) throughput close to that in the point-to-point backhaul due to the large and dual stage $grant\_size$ allocation for T3 $AllocIDs$ in all four DBAs. However, best-effort traffic shows the highest amount of throughput degradation among the three traffic types, with similar behaviour across all four DBAs, due to the unavoidable congestion in the XG-PON-based backhaul in the 52-eNB scenario caused by the over-loaded (instantaneous and average) XG-PON in the upstream.

8.5.2.2 Total Average Throughput

Figure 8.7 and Figure 8.8 represent the total average throughput for each and all of the application type(s), respectively, as measured during the steady state duration of [10s,60s]. Total average throughput is presented for each of the four DBAs in the XG-PON. All the figures present the values obtained in all 5 seeded experiments (indicated by 5 different colors in the plots), with the values corresponding to the 36-eNB and 52-eNB scenarios presented side-by-side. Values from the point-to-point backhaul environment are also presented in the *P2P x* axis for comparison, though not shown for the best-effort (an average of 1778Mbps) and all (an average of 2851Mbps) application behaviour in the 52-eNB scenario due to having elaborate y-axis ranges for the XG-PON backhaul environment.

In the 36-eNB scenario (left side figures in Figure 8.7), both XGIANT-D and XGIANT-P ensure total average throughput equivalent to that in the point-to-point backhaul. EBU and XGIANT ensure only the voice and video application throughput equivalent to that in the point-to-point backhaul, as their T4 allocation policies inefficiently utilise the unused bandwidth in the XG-PON upstream.

In the 52-eNB scenario (right side figures in Figure 8.7), voice applications experience slightly higher than the average throughput in the point-to-point backhaul as the increased (from 36-eNB) instantaneous congestions of the bursty voice traffic cause numerous retransmissions and RTO at the TCP, as seen for voice traffic in Figure 8.6. However, average video and best-effort throughput are slightly and heavily deteriorated respectively when compared with the point-to-point backhaul, for all the DBAs. Again, this is for the same reason of having more congestion instances from the video and best-effort applications in the upstream, with the added limitation of over-loaded XG-PON upstream capacity in the 52-eNB scenario, as opposed to an under-loaded upstream in the 36-eNB scenario. Among the DBAs, XGIANT-D and XGIANT-P allow highest total average throughput, followed by the XGIANT and then EBU, for video and best-effort applications in the 52-eNB scenario; this indicates that the improved QoS

(a) 36-eNB, throughput for voice

(b) 52-eNB, throughput for voice

(c) 36-eNB, throughput for video

(d) 52-eNB, throughput for video

(e) 36-eNB, throughput for best-effort

(f) 52-eNB, throughput for best-effort
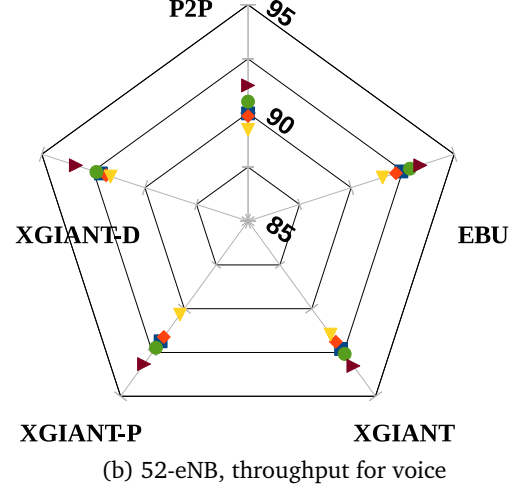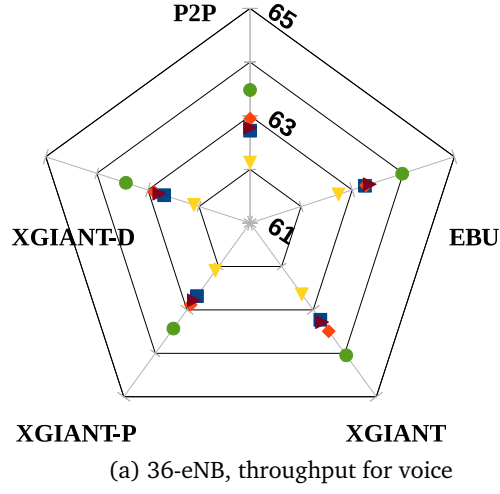
Figure 8.7: Total Average Throughput (in Mbps) for each of the applications in the 36-eNB & 52-eNB scenarios (colors represent results from different seeds of the same experiment; P2P best-effort average = 1778Mbps, not shown)

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

8.5 Evaluation results of TCP-based application
performance across XG-PON-based backhaul in LTE

(a) 36-eNB, throughput for all the applications   (b) 52-eNB, throughput for all the applications
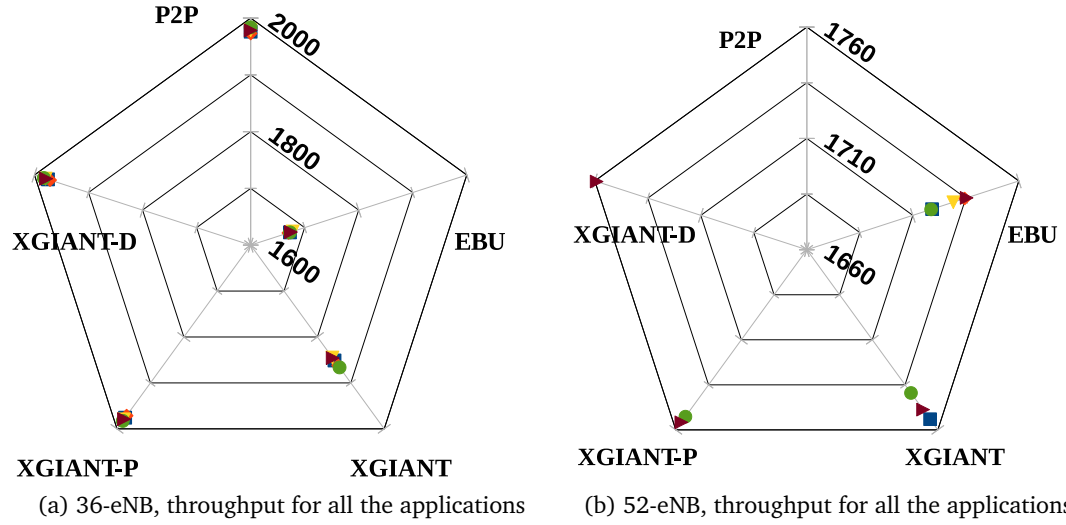
Figure 8.8: Total Average Throughput (Mbps) for all the applications in the 36-eNB and 52-eNB scenarios (colors represent results from different seeds of the same experiment; P2P average = 2851Mbps, not shown)

policies in XGIANT-D and XGIANT-P provide better (than XGIANT and EBU) throughput assurances when serving TCP-based applications the XG-PON-based LTE backhaul.

Since both XGIANT-D and XGIANT-P provide higher total average throughput for best-effort applications in the 36-eNB scenario and for video and best-effort applications in the 52-eNB scenario, both these DBAs show highest utilisation of the upstream capacity when XG-PON is slightly under-loaded and slightly over-loaded. Specifically, in the 36-eNB scenario, both XGIANT-D and XGIANT-P ensure total average throughput performance for all (and each) of the applications equivalent to that in the point-to-point backhaul environment, in all the 5 seeded experiments.

8.5.2.3   Mean Queuing-Delay

The mean queuing-delays experienced by each traffic type at each of its T-CONT queue for the 36-eNB scenario and 52-eNB scenario are presented here. Figures 8.9 and 8.10 illustrate the mean queuing-delays seen at every $AllocID$ in the entire XG-PON upstream for the 36-eNB and 52-eNB scenarios respectively. Sorted CDF values, with the range (represented by the vertical bars in the plot) and mean (represented by the dots in the plots) of the mean queuing-delays across all the 5 seeded experiments are plotted to provide the steady-state (between 10s and 60s of simulation) network-wide view of the influence of the QoS policies in the DBAs on the mean queuing-delays.

In both scenarios, all the DBAs, except XGIANT, provide XG-PON-standard-compliant prioritisation between voice, video and best-effort traffic by ensuring ascending mean queuing-delays in the descending order of the traffic type priority; that is, XGIANT-D, XGIANT-P and EBU ensure all the T2 $AllocIDs$ are assured with lowest queuing delays

(a) XGIANT-D

(b) XGIANT-P

(c) XGIANT

(d) EBU
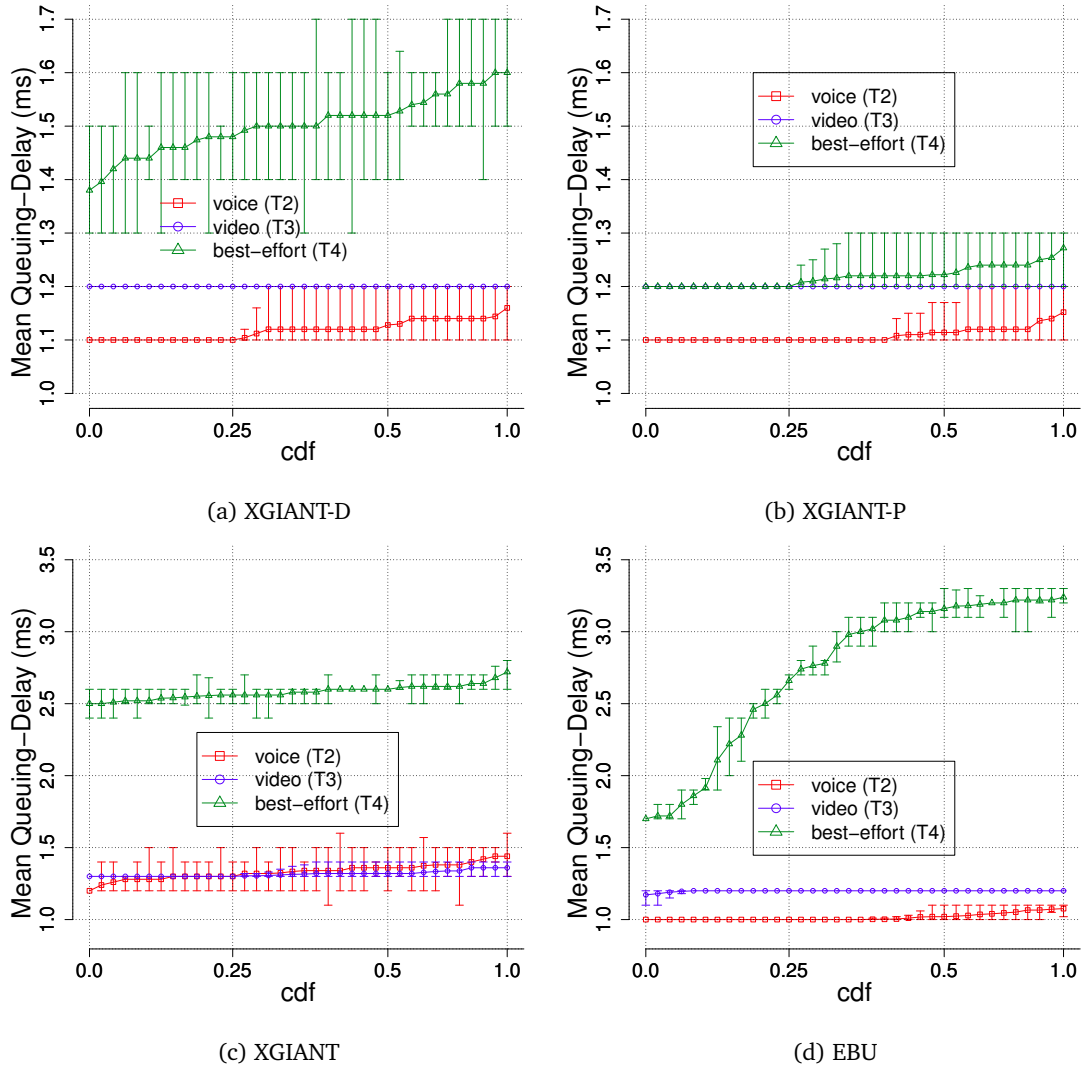
Figure 8.9: Mean Queuing-Delay (average and range) across 5 seeded experiments for the 36-eNB scenario in XG-PON-based LTE backhaul

(in all the 5 seeds) while the T4 $AllocIDs$ are provided with highest mean queuing-delays in all 5 seeds; T3 $AllocIDs$ show the mean queuing-delay values in between the values of any of the T2 or T4 $AllocIDs$.

When compared among XGIANT-D, XGIANT-P and EBU DBAs, EBU provides the lowest mean queuing-delay for T2 $AllocIDs$ due to its ability to smooth the burstiness of aggregated voice traffic by delaying the $grant\_size$ allocation every 5 US-FRAMEs; XGIANT-D and XGIANT-P provide the lowest mean queuing-delay values (compared to that in EBU) for video traffic due to their ability to dynamically utilise up to 70% + λ * 40% [or 70% + $(1-\lambda)$ * 40%; λ = 0.4] of the upstream capacity of XG-PON compared to 0.5 * 67% in EBU (and XGIANT). For the best-effort traffic, again XGIANT-D and XGIANT-P provide lowest mean queuing-delay values due to their ability to efficiently

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

8.5 Evaluation results of TCP-based application
performance across XG-PON-based backhaul in LTE

(a) XGIANT-D

(b) XGIANT-P

(c) XGIANT

(d) EBU
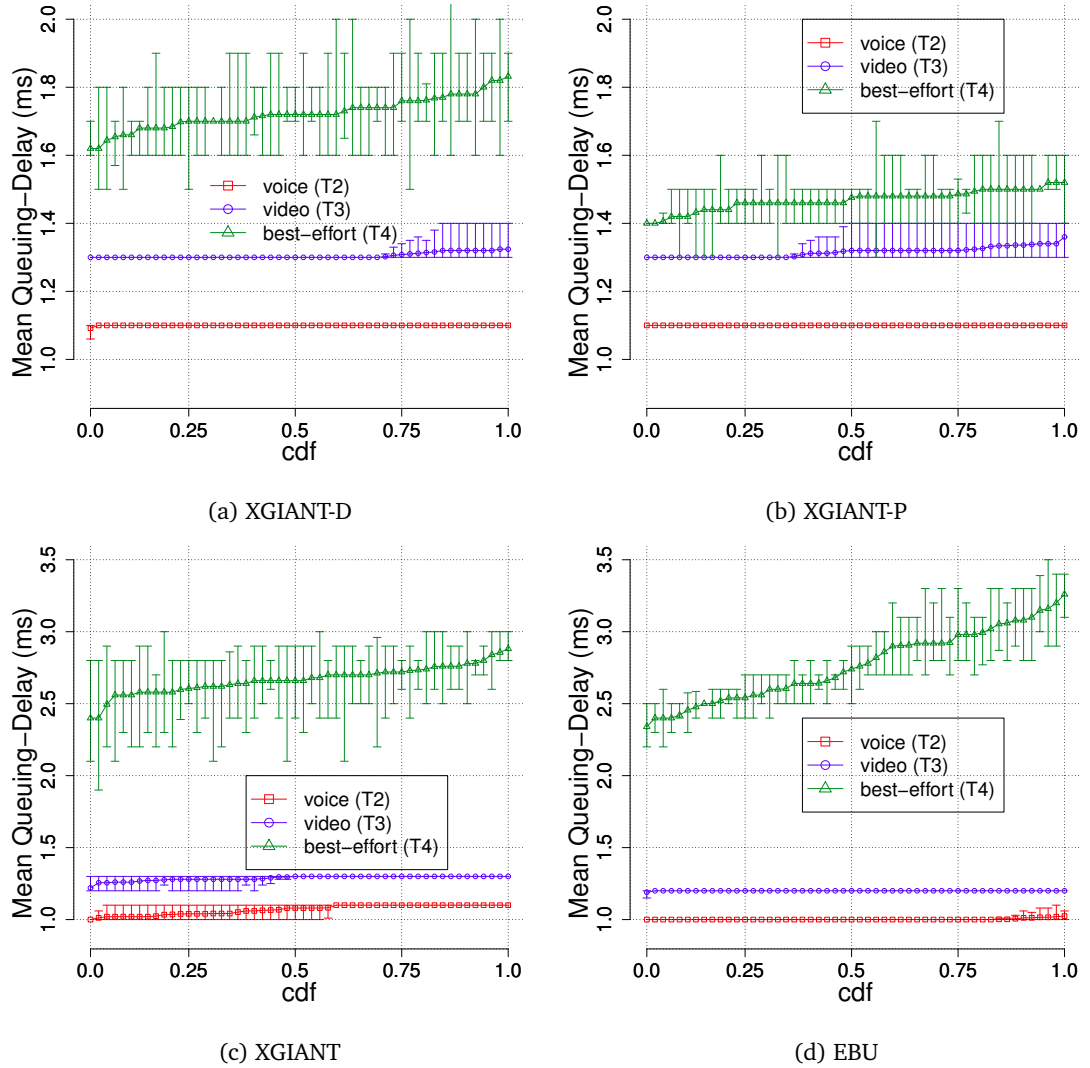
Figure 8.10: Mean Queuing-Delay (average and range) across 5 seeded experiments
for the 52-eNB scenario in the XG-PON backhaul in LTE

exploit the unutilised (after provisioning for voice and video $AllocIDs$) transmission
opportunity in every US-FRAME.

Fairness in queuing-delay is a direct interpretation of the number of steps a DBA intro-
duces for a given application type, when the mean queuing-delays are plotted individ-
ually for each seed of the experiment. Among the four DBAs, XGIANT-P indicated the
most fair mean queuing-delay behaviour for all three application types, due to having
intra-TCONT-type fairness and controlled $GrantSize$ over-provision for T4 $AllocIDs$;
specifically, for video traffic, EBU also indicated high degree of fairness, due to the
combined effect of having the inter-ONU fairness policy and dual-stage $GrantSize$ al-
location mechanism for T4 $AllocIDs$.

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

*8.5 Evaluation results of TCP-based application
performance across XG-PON-based backhaul in LTE*
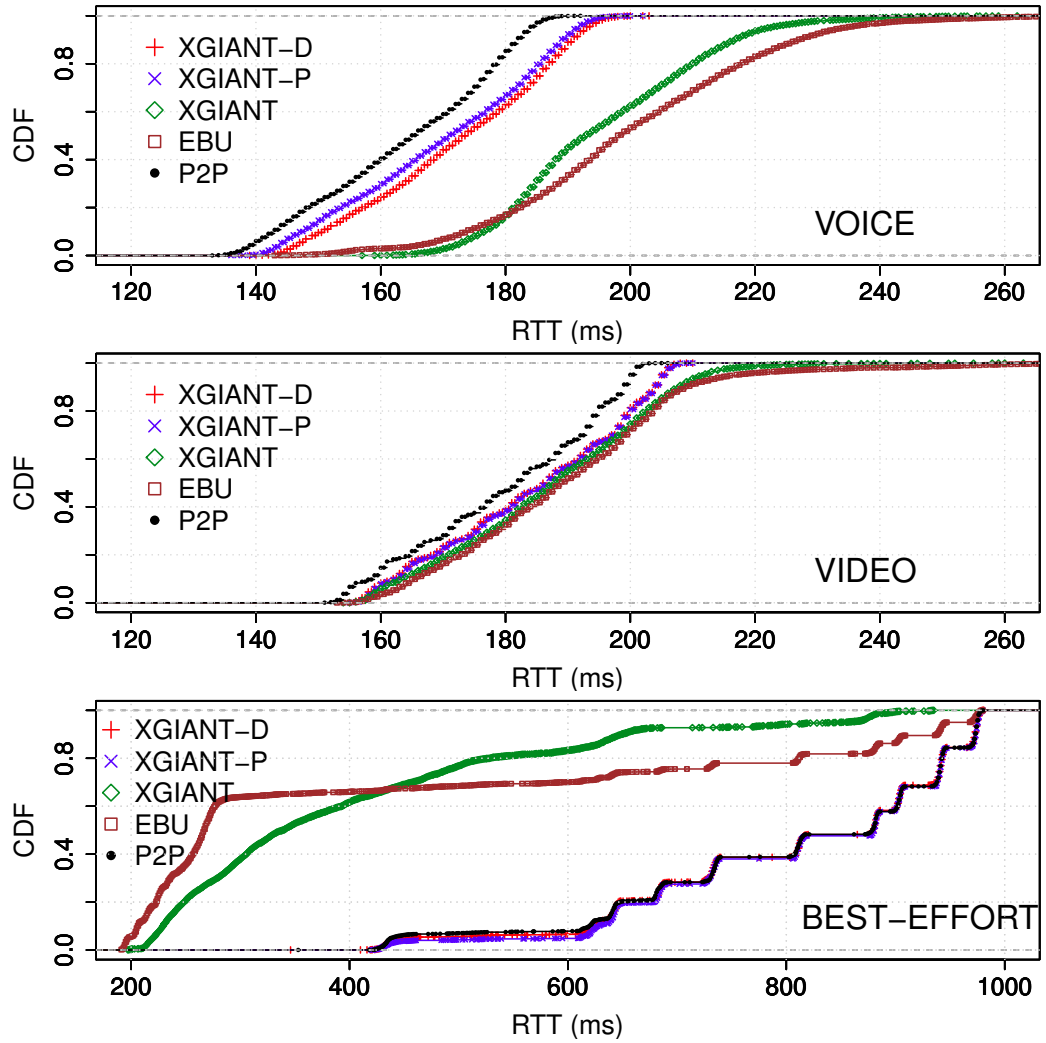
Figure 8.11: Network-wide RTT in the XG-PON (four DBAs) and point-to-point backhauls in LTE for the 36-eNB scenario

8.5.2.4 Average RTT

Figures 8.11 and 8.12 indicate the CDF of average the RTT values experienced by all the UEs in the 36-eNB (4540 UEs) and 52-eNB (6520 UEs) scenarios respectively. Each CDF plot contains the average RTT values experienced by the UEs from all 5 seeds of the experiment, during the steady-state simulation between 10s and 60s.

Values obtained for the point-to-point backhaul are also given for comparison purposes as well as differentiating the influence of 2 bottleneck links - air interface and XG-PON network - for upstream transmission in LTE. In the figures, if a DBA produces bigger RTT values compared to that in the point-to-point backhaul, it is an indication that XG-PON is an added bottleneck to the air interface for TCP-base upstream traffic in LTE; if the RTT values by the DBA are smaller, however, it is a clear indication that XG-PON is not an added bottleneck link, rather the only bottleneck link; an equivalent RTT value indicates negligible bottleneck effect by XG-PON in LTE backhaul.

8. Performance evaluation of TCP-based
upstream traffic in LTE across the
XG-PON backhaul

*8.5 Evaluation results of TCP-based application
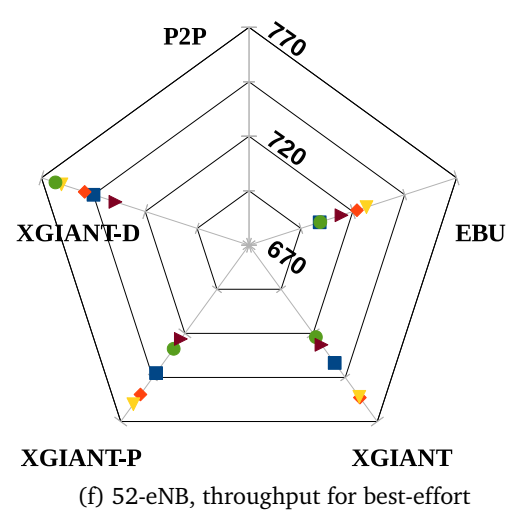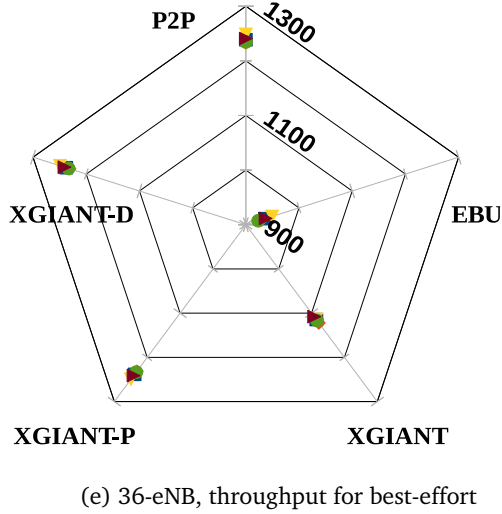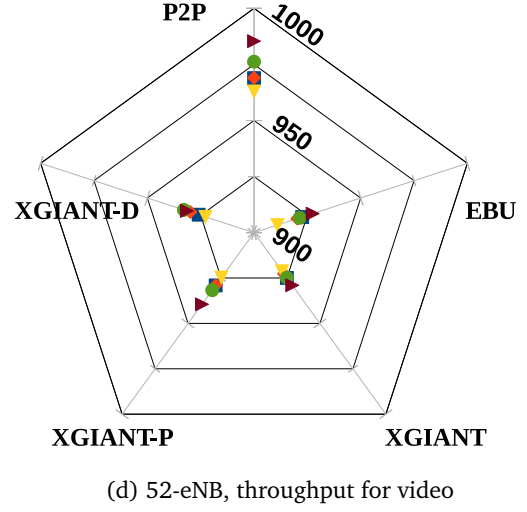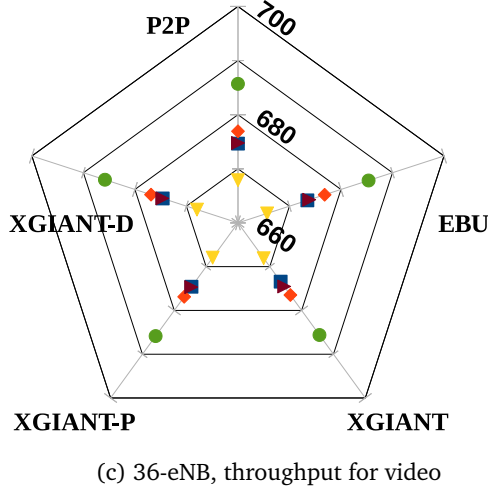performance across XG-PON-based backhaul in LTE*
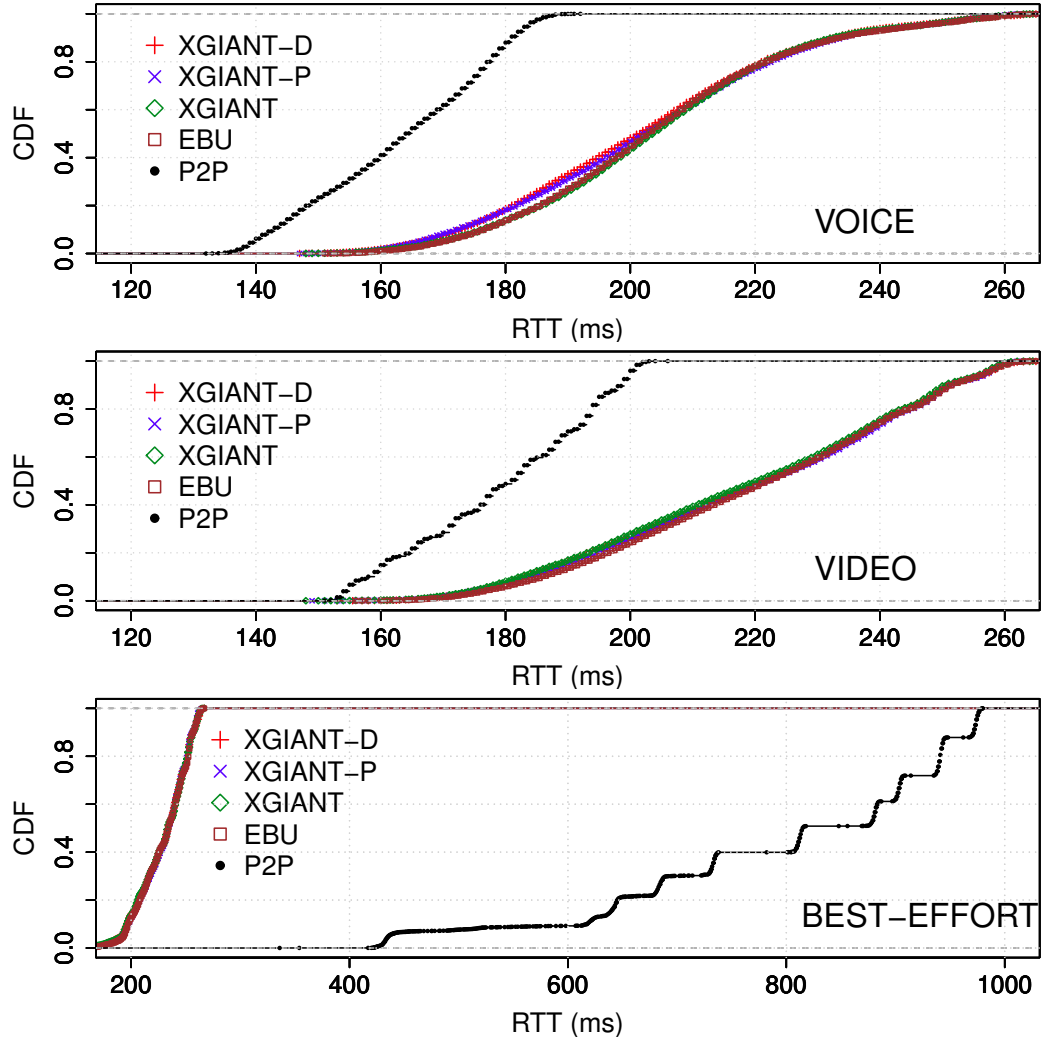
Figure 8.12: Network-wide RTT in the XG-PON (four DBAs) and point-to-point backhauls in LTE for the 52-eNB scenario

In both scenarios all the DBAs provide higher-than-point-to-point backhaul RTT values for voice and video UEs, indicating that XG-PON is an added bottleneck for the upstream transmission in LTE. However, the RTT values of XGIANT-D and XGIANT-P are very close to (but bigger than, due to queue-delays) that of point-to-point backhaul in the 36-eNB scenario for voice and video traffic; this indicates the ability of these two DBAs to ensure RTT performance of TCP-based applications in the XG-PON backhaul, similar to the performance across the over-provisioned point-to-point backhaul; as a result, a comparatively better instantaneous throughput performance can be seen for the voice and video applications when served by these two DBAs (section 8.5.2.1). When eNBs are increased from 36-eNBs to 52-eNBs, all the DBAs show higher-than-point-to-point backhaul RTT values for voice and video UEs indicating the substantial influence of the queuing-delays at the $AllocID$ queues with the increasing instances of instantaneous congestion (due to the over-loaded XG-PON upstream) experienced in the backhaul. Again section 8.5.2.1 further validates this behaviour with the mildly-

degraded (compared to point-to-point backhaul) throughput performance by all the DBAs for voice and video traffic.

A different RTT behaviour is seen for the best-effort traffic, when compared with the voice and video traffic, in both scenarios. That is, for the voice and video applications, the TCP sending rate is limited by the application datarates; however, for the best-effort traffic, TCP sending rate is limited not by the FTP application but by the limited LTE air interface capacity in the point-to-point backhaul environment. When the XG-PON replaces the point-to-point backhaul in LTE for the 36-eNB scenario, the throughput for best-effort still remained throttled by the air interface rather than the XG-PON, when using XGIANT-D and XGIANT-P DBAs (RTT values equivalent to point-to-point backhaul for best-effort traffic in Figure 8.11), validating the efficient T4 $GrantSize$ allocation in the DBAs; XGIANT and EBU, due to employing inefficient $GrantSize$ allocation policies for the T4 $AllocIDs$, cause XG-PON to be the bottleneck (in terms of capacity) in the upstream (lower RTT values than in the point-to-point backhaul for best-effort traffic in Figure 8.11) even though XG-PON is not fully loaded in the upstream.

In the 52-eNB scenario, due to the explicit congestion caused by all the application types in general and the best-effort FTP traffic in particular, all the DBAs indicate that XG-PON has become the only bottleneck link for the best-effort traffic (in terms of capacity) in the backhaul, by having smaller RTT values than in the point-to-point backhaul for best-effort traffic in Figure 8.11). These findings are further validated by the total instantaneous throughput performance in section 8.5.2.1 of this chapter.

8.6   Conclusion

This chapter presented the performance evaluation of TCP-based upstream applications in LTE, when served by four individual DBAs, in the XG-PON backhaul of LTE. The four DBAs, namely XGIANT-D, XGIANT-P, XGIANT and EBU, which were designed to provide differentiated QoS for XG-PON, are compared for their influence in providing reasonable throughput, RTT and queuing-delay performances towards TCP-based applications in the XG-PON-based LTE backhaul.

To provide a reasonable comparison environment, the chapter first presents a reference performance analysis for the TCP-based applications in an over-provisioned point-to-point LTE backhaul. Then two scenarios, each with a slightly under-loaded (36 eNBs) or slightly over-loaded (52 eNBs) XG-PON upstream, are used, to evaluate the performance of three types (voice, video and best-effort) of eNB-aggregated application traffic in LTE. In the under-loaded upstream scenario of the XG-PON backhaul in LTE, XGIANT-D and XGIANT-P (the two tailor-made DBAs for the XG-PON-based LTE backhaul) provided throughput and RTT assurances equivalent to that in the point-to-point backhaul, for all three types of TCP-based applications in the LTE backhaul; XGIANT

8. Performance evaluation of TCP-based
   upstream traffic in LTE across the
   XG-PON backhaul   *8.6 Conclusion*

and EBU DBAs (designed for the stand-alone XG-PON) however provided equivalent (to point-to-point backhaul) throughput and RTT assurances only for voice and video applications, while degrading throughput performance and proving increased E2E RTT values for the best-effort application traffic. In the over-loaded XG-PON upstream scenario, though all the DBAs indicated a highly degraded throughput performance for best-effort application traffic (compared to the point-to-point backhaul scenario), both voice and video applications were protected for reasonable throughput performance due to having differentiated QoS policies in each of the four DBAs.

Chapter 9

Conclusions and Future Work

This thesis proved that XG-PON-standard-compliant DBA mechanisms can ensure prioritised and fair QoS treatment for multiple classes of application traffic in the upstream of LTE networks using a dedicated XG-PON backhaul.

Chapter 1 provided the introduction to the thesis, with a brief overview, thesis statement, the contributions (and publications) of the author and the structure of the entire thesis. Chapter 2 introduced the technologies, concepts and terminologies required in a general reader to understand the contributions of the author presented in this thesis. Chapter 3 presented the related work from the literature to establish the rationalisation for the contributions in the thesis.

In order to develop standard-compliant DBA mechanisms for the FMC of XG-PON and LTE, where the LTE traffic is served by a dedicated XG-PON backhaul, Chapter 4 presented the design, implementation and validation details of a standard-compliant XG-PON simulation module for the ns-3 network simulator platform; the XG-PON simulation module was extensively used to analyse the influence of a simple Round-Robin DBA when using UDP-based (Chapter 4) and TCP-based (Chapter 5) application traffic in a stand-alone XG-PON, which serves fixed-broadband users; the TCP flows and not the UDP flows indicated significant unpredictability for the performance (throughput efficiency, fairness, convergence and responsiveness) across the XG-PON network, which in itself a large BDP network, thereby presenting UDP-based application traffic as the suitable candidate for DBA validations in the subsequent contributions presented in the thesis.

Chapter 6 presented the standard-compliant DBA, namely XGIANT, which was designed to provide differentiated QoS treatment for three different classes of upstream application traffic in a stand-alone XG-PON. XGIANT improved the benchmarked GIANT DBA (designed for GPON), by introducing effective improvements for the QoS policies and key parameters of GIANT. The performance evaluations of XGIANT, carried out using the standard-compliant XG-PON simulation module, indicated the QoS capabilities of

XGIANT to provide prioritised mean queuing-delay and throughput assurances for the three classes of deterministic application traffic, indicating the QoS assurances by the DBA for the fixed-broadband users in the XG-PON upstream. XGIANT also indicated its superiority in terms of simplicity and lower mean queuing-delay values, when compared with the validated implementation (in the same XG-PON simulation module) of the EBU DBA, which is also a recent GIANT-improved DBA for XG-PON.

Chapter 7 provided evaluations of XGIANT and EBU, for providing QoS treatment for three different classes (conversational voice, peer-to-peer video and best-effort Internet) of realistically-generated upstream traffic in the dedicated XG-PON backhaul in LTE; the evaluations, performed after implementing an integrated network architecture and the QoS metrics conversion scheme for the FMC of XG-PON and LTE in ns-3, indicated that both XGIANT and EBU were not capable of ensuring prioritised mean queuing-delay performances for the application traffic in the XG-PON backhaul in LTE. Based on the evaluations and improving upon the QOS framework of both XGIANT and EBU, two DBAs, namely XGIANT-D and XGIANT-P were designed for the XG-PON backhaul in LTE. XGIANT-D and XGIANT-P incorporated improved policies such as strict prioritisations among different traffic classes, burstiness smoothing of eNB-aggregated voice and video traffic, efficient bandwidth allocation policies (hence two flavours of suitable DBAs) for the best-effort traffic, simplified bandwidth allocation policies and efficient intra-TCONT-type fairness policy. The improved DBAs were evaluated and compared with XGIANT and EBU DBAs for mean queuing-delay and throughput performances for the . Evaluations indicated that, for under-loaded and marginally over-loaded upstream conditions in the XG-PON backhaul, XGIANT-D and XGIANT-P provided fair and low mean queuing-delay performances and efficient throughput assurances as opposed to XGIANT and EBU. Chapter 7 also provided performance evaluations of all the four DBAs, in terms of datarate burstiness and queuing-jitter to provide insight into the relative performance of the DBAs to further validate the suitability and standard-compliance of XGIANT-D and XGIANT-P DBAs for providing QoS treatment for different classes of upstream traffic in LTE across a dedicated XG-PON backhaul.

Since TCP is the most dominant Transport protocol in the Internet, Chapter 8 finally presented the evaluations regarding the performance of TCP-based upstream application traffic in LTE across the dedicated XG-PON backhaul, when individually employing the XGIANT-D and XGIANT-P (as well as XGIANT and EBU) DBAs. The evaluations of the DBAs, when serving three classes of realistically-generated and TCP-based application traffic, indicated that XGIANT-D and XGIANT-P ensured reasonable and better (than XGIANT and EBU) throughput efficiency and E2E RTT values specifically for the voice and video traffic, be it in the under-loaded or over-loaded upstream condition in the XG-PON backhaul; for the best-effort traffic, while a non-congested performance (throughput and RTT) was provided in the under-loaded scenario, reasonably-degraded throughput performances was observed when the XG-PON upstream was

over-loaded for capacity; these results, hence, indicated the robustness of the QoS framework of XGIANT-D and XGIANT-P in providing standard-compliant QoS treatment for realistic (TCP-based) upstream traffic in LTE across the dedicated XG-PON backhaul.

Having proved, in this thesis, that the XG-PON-standard-compliant DBA mechanisms can ensure prioritised and fair QoS treatment for three different classes of upstream application traffic in LTE across a dedicated XG-PON backhaul, the author is interested in exploring the following research avenues in the future:

- Symmetrical XG-PON (XGSPON[49]) is a very recent standard by ITU-T and 5G is an emerging standard in the wireless last-mile domain. While XGSPON extends XGPON1 by adding a symmetrical 10Gb/s datarate in both downstream and upstream directions, 5G promises several improvements in terms of throughput and delay guarantees at the wireless interface. Since the scale of last-mile nodes (number of 5G base stations supported by XGSPON, UEs attached to per base station, aggregated traffic pattern at a 5G base station, etc.) may not be the exact proportion of the capacity upgrade from XG-PON to XGSPON, it will be an interesting feature to validate the performance of the XGIANT-D and XGIANT-P DBA in the context of a dedicated XGSPON-based 5G backhaul when providing QoS guarantees for multiple classes of 5G user traffic. As the Time-Wavelength Division Multiplexed PON (TWDM-PON[65]) is an emerging standard for the access network[73], the DBA mechanism developed only for the XG-PON-based LTE backhaul in this thesis can also be extended towards supporting multiple wavelengths, in a TWDM-PON-based 4G/5G backhaul.

- The *independent* network architecture used in this thesis integrates the LTE and XG-PON networks as two individual network entities thereby allowing little sharing between the resources of MAC layers of both technologies. A shared MAC layer architecture such as in the *generic hybrid* architecture 3.2.1 where the resource sharing (e.g.: MAC queues) can be quantified in addition to the performance assurances by the XGIANT-D/-P DBAs so that an infrastructure provider is further motivated to adapt the research outcomes of the QoS framework in realistic conditions requiring strong integration between LTE and XG-PON.

- When using an XG-PON-based backhaul in LTE, the impact of backhaul traffic provisioning by a QoS-aware DBA when subjected to different mobility patterns and hand-over mechanisms in the last-mile may significantly alter the end-to-end QoS guarantees required for different classes of user traffic, generated by both a static LTE user and mobile LTE user who may or may not be subjected to a hand-over during the experiments. Such a validation may provide validation into the robustness of the DBAs (of XG-PON) for a more realistic (mobile and static users) last-mile scenario in LTE wireless interface.

- Providing QoS in an XG-PON backhaul which is shared among a mix of wired and wireless technologies is challenging is several aspects. Firstly a suitable integration architecture as well as a QoS metric conversion scheme should be in place for multiple last-mile technologies to efficiently share a single XG-PON backhaul. Secondly, since different last-mile technologies (e.g.: LTE and Fixed Broadband) is inherently different in terms of the last-mile throughput and delay guarantees for traffic flows, it may be a challenging task to employ techniques such as dynamic T-CONT queue sizes and adaptive QoS metric conversion schemes will be required to provide different scales of QoS treatment in the backhaul for traffic flows from different last-mile technologies. This avenue of research can also be extended towards the popular research topic of WiFi offloading of mobile date, where a single XG-PON-based backhaul can be used for the same mobile user traffic regardless of the wireless last-mile technology and/or TCP protocol (multipath or generic) chosen by the user.

The author of the thesis also remains optimistic, based on the market analysis [19], that the XG-PON-based LTE backhaul is a practical solution for deployment in the near future. The main driving factor for the deployment is the increased downstream and upstream datarates up to 10Gb/s in the recently upgraded XGSPON standard, and up to 40Gb/s in the standardisation of TWDM PON. A fixed broadband operator who provides the XG-PON-based LTE backhaul solution will, however, largely face socioeconomic challenges such 1) promising better return-over-investment against the existing point-to-point copper, fibre and microwave links [32] and 2) justification of providing a joint and profitable solution alongside the FTTH deployment, specifically in the European region, where a massive-scale deployment of GPON is already present and the saturation level of fixed broadband users is low [19] compared to countries (e.g.: the United States of America, Canada and Japan), thus requiring a huge additional investment to support a large number of cellular base stations. The author of the thesis is also hopeful that when the socio-economic factors are met, the proposals in this thesis for the standard-compliant, QoS-aware DBAs will become an inspiration for prototyping, validating and implementing DBAs in the real deployments of the XG-PON-based LTE backhaul solutions.

Appendix A

Class Diagram

Figure A.1: Class diagram of the XG-PON simulation module presented in Chapter 4
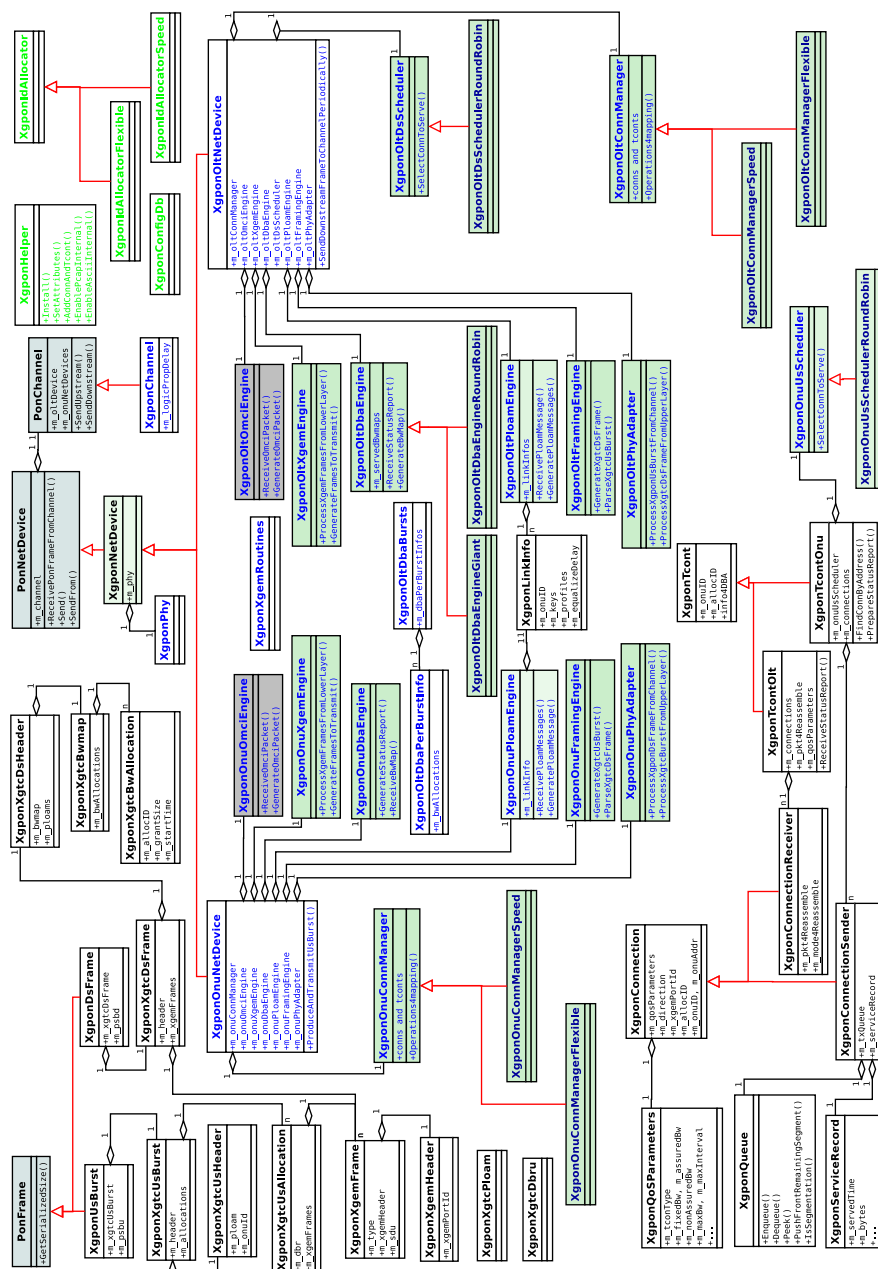
Appendix B

Tables

Table B.1: Number of UEs per eNB for the 5 seeds of experiments in Chapter 8

eNB No.	36-eNB Scenario					52-eNB Scenario				
	Seed 1	Seed 2	Seed 3	Seed 4	Seed 5	Seed 1	Seed 2	Seed 3	Seed 4	Seed 5
1	28	27	27	25	21	28	27	27	25	21
2	26	28	29	22	29	26	28	29	22	29
3	25	26	29	27	22	25	26	29	27	22
4	24	22	28	27	25	24	22	28	27	25
5	24	25	23	29	26	24	25	23	29	26
6	28	29	27	28	25	28	29	27	28	25
7	21	28	21	25	28	21	28	21	25	28
8	26	22	29	26	24	26	22	29	26	24
9	29	26	29	25	24	29	26	29	25	24
10	26	27	23	21	21	26	27	23	21	21
11	28	28	24	24	25	28	28	24	24	25
12	26	25	21	28	29	26	25	21	28	29
13	23	26	26	27	22	23	26	26	27	22
14	28	22	24	22	22	28	22	24	22	22
15	21	27	26	23	29	21	27	26	23	29
16	28	22	27	29	29	28	22	27	29	29
17	29	23	26	22	21	29	23	26	22	21
18	28	26	23	21	26	28	26	23	21	26
19	21	28	22	28	29	21	28	22	28	29
20	25	21	27	29	29	25	21	27	29	29
21	22	26	23	28	22	22	26	23	28	22
22	24	27	21	27	26	24	27	21	27	26
23	24	25	25	28	24	24	25	25	28	24
24	24	28	21	24	24	24	28	21	24	24
25	29	23	28	29	29	29	23	28	29	29
26	25	22	22	21	22	25	22	22	21	22
27	24	25	28	23	26	24	25	28	23	26
28	23	27	22	29	22	23	27	22	29	22
29	29	28	24	24	26	29	28	24	24	26
30	27	28	23	23	21	27	28	23	23	21
31	25	21	24	24	29	25	21	24	24	29
32	23	22	24	28	25	23	22	24	28	25
33	25	28	28	21	27	25	28	28	21	27
34	22	23	23	28	27	22	23	23	28	27
35	27	21	27	29	28	27	21	27	29	28
36	21	29	25	29	24	21	29	25	29	24
37						26	25	27	25	21
38						27	28	28	26	27
39						26	23	25	25	27
40						25	21	21	21	22
41						22	23	27	25	28
42						24	26	23	26	25
43						28	25	22	25	29
44						22	26	22	22	28
45						24	21	24	23	27
46						24	24	21	21	28
47						27	21	25	25	24
48						23	29	29	26	23
49						26	23	26	26	27
50						23	21	22	22	28
51						25	28	28	28	28
52						24	29	27	25	26

Table B.2: Number of UEs per eNB for the 10 seeds of experiments in Chapter 7

eNB No	52-eNB Scenario										80-eNB Scenario									
	Seed1	Seed2	Seed3	Seed4	Seed5	Seed6	Seed7	Seed8	Seed9	Seed10	Seed1	Seed2	Seed3	Seed4	Seed5	Seed6	Seed7	Seed8	Seed9	Seed10
1	28	27	27	25	21	28	23	22	29	23	28	27	27	25	21	28	23	22	29	23
2	26	28	29	22	29	23	23	24	23	25	26	28	29	22	29	23	23	24	23	25
3	25	26	29	27	22	27	25	28	28	26	25	26	29	27	22	27	25	28	28	26
4	24	22	28	27	25	22	21	25	25	25	24	22	28	27	25	22	21	25	25	25
5	24	25	23	29	26	27	21	29	28	26	24	25	23	29	26	27	21	29	28	26
6	28	29	27	28	25	21	26	21	21	21	28	29	27	28	25	21	26	21	21	21
7	21	28	21	25	28	22	25	21	29	29	21	28	21	25	28	22	25	21	29	29
8	26	22	29	26	24	23	24	26	22	24	26	22	29	26	24	23	24	26	22	24
9	29	26	29	25	24	25	21	22	28	25	29	26	29	25	24	25	21	22	28	25
10	26	27	23	21	21	22	29	27	23	22	26	27	23	21	21	22	29	27	23	22
11	28	28	24	24	25	22	27	26	27	29	28	28	24	24	25	22	27	26	27	29
12	26	25	21	28	29	24	24	29	26	28	26	25	21	28	29	24	24	29	26	28
13	23	26	26	27	22	28	22	28	21	28	23	26	26	27	22	28	22	28	21	28
14	28	22	24	22	22	29	24	22	26	28	28	22	24	22	22	29	24	22	26	28
15	21	27	26	23	29	21	21	26	27	26	21	27	26	23	29	21	21	26	27	26
16	28	22	27	29	29	29	21	25	24	24	28	22	27	29	29	29	21	25	24	24
17	29	23	26	22	21	21	26	28	27	26	29	23	26	22	21	21	26	28	27	26
18	28	26	23	21	26	29	23	28	21	25	28	26	23	21	26	29	23	28	21	25
19	21	28	22	28	29	23	28	25	23	26	21	28	22	28	29	23	28	25	23	26
20	25	21	27	29	29	24	24	28	26	22	25	21	27	29	29	24	24	28	26	22
21	22	26	23	28	22	22	21	21	26	24	22	26	23	28	22	22	21	21	26	24
22	24	27	21	27	26	29	26	27	24	28	24	27	21	27	26	29	26	27	24	28
23	24	25	25	28	24	27	27	26	22	29	24	25	25	28	24	27	27	26	22	29
24	24	28	21	24	24	21	22	29	22	27	24	28	21	24	24	21	22	29	22	27
25	29	23	28	29	29	28	24	23	24	27	29	23	28	29	29	28	24	23	24	27
26	25	22	22	21	22	25	29	26	26	26	25	22	22	21	22	25	29	26	26	26
27	24	25	28	23	26	22	22	21	23	22	24	25	28	23	26	22	22	21	23	22
28	23	27	22	29	22	26	28	22	22	25	23	27	22	29	22	26	28	22	22	25
29	29	28	24	24	26	26	28	26	27	24	29	28	24	24	26	26	28	26	27	24
30	27	28	23	23	21	27	21	27	24	24	27	28	23	23	21	27	21	27	24	24
31	25	21	24	24	29	25	29	24	22	27	25	21	24	24	29	25	29	24	22	27
32	23	22	24	28	25	24	27	26	21	22	23	22	24	28	25	24	27	26	21	22
33	25	28	28	21	27	21	27	22	27	29	25	28	28	21	27	21	27	22	27	29
34	22	23	23	28	27	24	26	27	23	28	22	23	23	28	27	24	26	27	23	28
35	27	21	27	29	28	29	28	21	24	27	27	21	27	29	28	29	28	21	24	27
36	21	29	25	29	24	21	22	25	28	28	21	29	25	29	24	21	22	25	28	28
37	26	25	27	25	21	22	24	21	22	21	26	25	27	25	21	22	24	21	22	21
38	27	28	28	26	27	28	27	28	21	27	27	28	28	26	27	28	27	28	21	27
39	26	23	25	25	27	27	28	21	21	24	26	23	25	25	27	27	28	21	21	24
40	25	21	21	21	22	22	27	29	23	28	25	21	21	21	22	22	27	29	23	28
41	22	23	27	25	28	24	29	25	27	27	22	23	27	25	28	24	29	25	27	27
42	24	26	23	26	25	21	26	28	23	26	24	26	23	26	25	21	26	28	23	26
43	28	25	22	25	29	27	22	28	23	24	28	25	22	25	29	27	22	28	23	24
44	22	26	22	22	28	27	22	24	24	23	22	26	22	22	28	27	22	24	24	23
45	24	21	24	23	27	29	29	27	26	25	24	21	24	23	27	29	29	27	26	25
46	24	24	21	21	28	24	25	23	27	24	24	24	21	21	28	24	25	23	27	24
47	27	21	25	25	24	24	24	25	25	24	27	21	25	25	24	24	24	25	25	24
48	23	29	29	26	23	27	22	23	21	26	23	29	29	26	23	27	22	23	21	26
49	26	23	26	26	27	21	21	28	24	26	26	23	26	26	27	21	21	28	24	26
50	23	21	22	22	28	28	21	21	24	21	23	21	22	22	28	28	21	21	24	21
51	25	28	28	28	28	23	22	25	22	23	25	28	28	28	28	23	22	25	22	23
52	24	29	27	25	26	22	27	27	23	27	24	29	27	25	26	22	27	27	23	27
53											26	27	26	24	28	29	29	28	22	21
54											25	25	21	22	27	23	27	29	26	22
55											22	23	24	21	27	22	28	27	29	23
56											22	26	25	23	27	28	21	22	27	26
57											28	21	26	24	27	27	21	24	24	24
58											26	21	27	23	29	29	24	26	23	25
59											24	25	23	28	22	25	26	29	22	21
60											26	24	28	28	21	28	27	28	24	25
61											25	22	21	29	25	28	29	25	24	29
62											21	27	28	26	27	21	25	25	22	25
63											21	25	25	25	27	27	27	23	21	23
64											25	26	21	25	23	24	22	24	28	29
65											25	27	27	21	24	25	29	26	25	23
66											27	29	22	28	24	27	22	25	23	23
67											29	23	25	28	29	22	27	21	26	26
68											23	24	23	26	22	25	28	27	26	22
69											22	28	29	27	22	29	25	26	24	29
70											29	26	28	21	27	26	21	28	23	21
71											27	21	21	24	21	26	25	26	29	21
72											28	23	28	27	24	28	24	24	29	26
73											27	21	23	23	21	22	21	21	27	28
74											29	28	22	29	27	24	28	21	27	25
75											26	28	27	29	26	24	23	21	29	24
76											25	22	21	22	28	22	25	22	26	29
77											26	29	24	26	21	28	25	25	26	24
78											29	24	29	22	24	24	21	22	24	28
79											24	23	25	23	26	28	26	26	28	24
80											23	25	24	21	28	25	22	27	28	24

References

[1] 3GPP-TS-23.203. Policy and charging control architecture.

[2] The NS-3 network simulator. Available at: http://www.nsnam.org, 2015.

[3] Najah Abu-Ali, Abd-Elhamid M. Taha, Mohamed Salah, and Hossam Hassanein. Uplink Scheduling in LTE and LTE-Advanced: Tutorial, Survey and Evaluation Framework. *IEEE Communications Surveys & Tutorials*, PP(99):1–27, 2014.

[4] Mohamed A Ali, Georgios Ellinas, Hasan Erkan, Antonis Hadjiantonis, and Roger Dorsinville. On the Vision of Complete Fixed-Mobile Convergence. *J. Light. Technol.*, 28(16):2343–2357, Aug 2010.

[5] N.M. Alvarez, P. Kourtessis, R.M. Lorenzo, and J.M. Senior. Full-Service MAC Protocol for Metro-Reach GPONs. *Journal of Lightwave Technology*, 28(7):1016–1022, Apr 2010.

[6] Pedro Alvarez, Nicola Marchetti, David Payne, and Marco Ruffini. Backhauling mobile systems with XG-PON using grouped assured bandwidth. In *2014 19th European Conference on Networks and Optical Communications - (NOC)*, pages 91–96. IEEE, Jun 2014.

[7] Doreid Ammar, Thomas Begin, and Isabelle Guerin-Lassous. A new tool for generating realistic internet traffic in ns-3. In *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*, SIMUTools '11, pages 81–83, ICST, Brussels, Belgium, Belgium, 2011. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[8] N. Prasanth Anthapadmanabhan, Nga Dinh, Anwar Walid, and Adriaan J. van Wijngaarden. Analysis of a probing-based cyclic sleep mechanism for passive optical networks. In *2013 IEEE Global Communications Conference (GLOBECOM)*, pages 2543–2548. IEEE, Dec 2013.

[9] B. Mehrabi Armin, P. Benson Shin, P. Schuller Michael, and L. Noland James. Experimental Evaluation of Cubic-TCP. In *Journal of Structural Engineering*, volume 122, pages 228–237, 1996.

[10] Jerome A Arokkiam, Pedro Alvarez, Xiuchao Wu, Kenneth N Brown, Cormac J Sreenan, Marco Ruffini, Nicola Marchetti, Linda Doyle, and David Payne. Design, implementation, and evaluation of an XG-PON module for the ns-3 network simulator. *SIMULATION*, pages 1–18, jan 2017.

[11] Jerome A. Arokkiam, Kenneth N. Brown, and Cormac J. Sreenan. Refining the GIANT dynamic bandwidth allocation mechanism for XG-PON. In *2015 IEEE International Conference on Communications (ICC)*, pages 1006–1011. IEEE, Jun 2015.

[12] Jerome A. Arokkiam, Kenneth N. Brown, and Cormac J. Sreenan. Optimised QoS-Aware DBA Mechanisms in XG-PON for Upstream Traffic in LTE Backhaul. In *2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW)*, pages 361–368, Vienna, Austria, Aug 2016. IEEE.

[13] Jerome A. Arokkiam, Xiuchao Wu, Kenneth N. Brown, and Cormac J. Sreenan. Experimental evaluation of TCP performance over 10Gb/s passive optical networks (XG-PON). In *2014 IEEE Global Communications Conference*, pages 2223–2228. IEEE, Dec 2014.

[14] Carlos Astudillo and Nelson Da Fonseca. Standard-compliant QoS provisioning scheme for LTE/EPON integrated networks. *IEEE Wireless Communications*, 21(3):44–51, Jun 2014.

[15] Frank Aurzada, Martin Levesque, Martin Maier, and Martin Reisslein. FiWi Access Networks Based on Next-Generation PON and Gigabit-Class WLAN Technologies: A Capacity and Delay Analysis. *IEEE/ACM Transactions on Networking*, 22(4):1176–1189, Aug 2014.

[16] Nicola Baldo, Marco Miozzo, Manuel Requena-Esteso, and Jaume Nin-Guerrero. An open source product-oriented LTE network simulator based on ns-3. In *Proceedings of the 14th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems - MSWiM '11*, page 293, New York, New York, USA, 2011. ACM Press.

[17] Andreas Bodozoglou. EPON for OMNeT++. Available at: http://sourceforge.net/projects/omneteponmodule/, Sep 2010.

[18] P Bolletta, A Del Grosso, L Rea, A M Luisi, S Pompei, A Valenti, and D Del Buono. Monitoring of the user Quality of Service: Network architecture for measurements, role of the user operating system with consequences for optical accesses. In *Optical Network Design and Modeling (ONDM), 2011 15th International Conference on*, pages 1–5, 2011.

[19] Krzysztof Borzycki. Development of FTTH-PON technologies: Market reality check 2010–2014. In *2014 16th Int. Telecommun. Netw. Strateg. Plan. Symp.*, pages 1–6. IEEE, Sep 2014.

[20] Christos J. Bouras. *Trends in Telecommunications Technologies*. InTech, 2010.

[21] C.-H. Chang, P. Kourtessis, and J.M. Senior. GPON service level agreement based dynamic bandwidth assignment protocol. *Electronics Letters*, 42(20):1173, 2006.

[22] Ching-Hung Chang. *Dynamic Bandwidth Allocation MAC Protocols for Gigabit-capable Passive Optical Networks*. PhD thesis, University of Hertfordshire, Jul 2008.

[23] Kai-Chien Chang and Wanjiun Liao. On the throughput and fairness performance of TCP over ethernet passive optical networks. *IEEE Journal on Selected Areas in Communications*, 24(12):3–12, Dec 2006.

[24] Cisco. Cisco VNI: Forecast and Methodology, 2015 – 2020, 2016.

[25] CTTC. Design Documentation — LENA v8 documentation, 2016.

[26] R. D.Leith, Shorten, and Hamilton Ins. H-TCP protocol for high-speed long-distance networks. In *Proc. PFLDnet, Argonne*, pages 1–16, 2004.

[27] S. Domoxoudis, S. Kouremenos, A. Drigas, and V. Loumos. Frame-based modeling of H.264 constrained videoconference traffic over an IP commercial platform. In *2nd International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities, TRIDENTCOM 2006*, volume 2006, pages 216–221. IEEE, 2006.

[28] Jahanzeb Farooq and Thierry Turletti. An IEEE 802.16 WiMAX module for the NS-3 simulator. In *Proceedings of the Second International ICST Conference on Simulation Tools and Techniques*. ICST, 2009.

[29] D. Nauka V. Fernando, Milos Milosavljevic, Pandelis Kourtessis, and John M. Senior. Cooperative cyclic sleep and doze mode selection for NG-PONs. In *2014 16th International Conference on Transparent Optical Networks (ICTON)*, pages 1–4. IEEE, Jul 2014.

[30] S Floyd, T Henderson, and A Gurtov. The NewReno Modification to TCP's Fast Recovery Algorithm. RFC 3782, Apr 2004.

[31] Navid Ghazisaidi and Martin Maier. Fiber-wireless (FiWi) access networks: Challenges and opportunities. *IEEE Network*, 25(1):36–42, Jan 2011.

[32] S Gosselin, D B Joseph, F Moufida, T Mamouni, J A Torrijos, L Cucala, D Breuer, E Weis, F Geilhardt, D v. Hugo, E Bogenfeld, A Hamidian, N Fonseca, Y Liu, S Kuehrer, A Gravey, A Mitcsenkov, J V Galan, E Masgrau Rite, L S Gomez,

L Alonso, S Hoest, A Magee, J De Biasio, M Feknous, D Hugo, E Bogenfeld, A Hami, N Fonseca, Y Liu, S Kuehrer, A Grav, L Gómez, L Alonso, S Höst, and A Mage. Fixed and Mobile Convergence: Needs and a Solutions. In *European Wireless 2014; 20th European Wireless Conference; Proceedings of*, number 2, pages 425–430, may 2014.

[33] Sangtae Ha, Injong Rhee, and Lisong Xu. CUBIC: a new TCP-friendly high-speed TCP variant. *Operating Systems Review*, 42(5):64–74, Jul 2008.

[34] Man-Soo Han. Simple and feasible dynamic bandwidth and polling allocation for XGPON. In *16th International Conference on Advanced Communication Technology*, pages 298–304. Global IT Research Institute (GIRI), Feb 2014.

[35] Man Soo Han, Hark Yoo, and Dong Soo Lee. Development of Efficient Dynamic Bandwidth Allocation Algorithm for XGPON. *ETRI Journal*, 35(1):18–26, Feb 2013.

[36] Man-Soo Han, Hark Yoo, Bin-Young Yoon, Bongtae Kim, and Jai-Sang Koh. Efficient dynamic bandwidth allocation for FSAN-compliant GPON. *Journal of Optical Networking*, 7(8):783, Jul 2008.

[37] Thomas R Henderson, Mathieu Lacage, George F Riley, Craig Dowell, and Joseph B Kopena. Network simulations with the ns-3 simulator. In *Demonstrations of the ACM SIGCOMM 2008 conference on Data communication*, 2008.

[38] Junxian Huang, Feng Qian, Yihua Guo, Yuanyuan Zhou, Qiang Xu, Z. Morley Mao, Subhabrata Sen, Oliver Spatscheck, Junxian Huang, Feng Qian, Yihua Guo, Yuanyuan Zhou, Qiang Xu, Z. Morley Mao, Subhabrata Sen, and Oliver Spatscheck. An in-depth study of LTE: effect of network protocol and application behavior on performance. In *Proceedings of the ACM SIGCOMM'13*, volume 43, page 363, New York, New York, USA, 2013. ACM Press.

[39] I-Shyan Hwang, Tzu-Jui Yeh, MohammadAmin Lotfolahi, Bor-Jiunn Hwang, and AliAkbar Nikoukar. Synchronous Interleaved DBA for Upstream Transmission over GPON - LTE Converged Network. In *Int. MultiConference Eng. Comput. Sci.*, volume 2216, pages 550–555. Newswood and International Association of Engineers, Mar 2015.

[40] IEEE. 802.3ah: Ethernet in the First Mile, 2004.

[41] IEEE. 802.3ah Task Force, Jun 2004.

[42] IEEE. 802.3av 10G-EPON Task Force, Sep 2009.

[43] ITU. 10-Gigabit-Capable Passive Optical Networks (GPON) Series of Recommendations. G.987.x, Mar 2010.

[44] ITU-R. M.2134: Requirements Related to Technical Performance for IMT-Advanced Radio interface(s), 2008.

[45] ITU-T. G.984.x: Gigabit-Capable Passive Optical Networks (G-PON). Rec. G.984.x, Oct 2008.

[46] ITU-T. G.987.1: XGPON General Characteristics, 2010.

[47] ITU-T. G.987.3: XGPON Transmission Convergence layer specification, 2010.

[48] ITU-T. G.722 : 7 kHz audio-coding within 64 kbit/s, 2012.

[49] ITU-T. G.9807.1 (G.XGS-PON): 10-Gigabit-capable symmetric passive optical network (XGS-PON), Jun 2016.

[50] Mauricio Iturralde, Steven Martin, and Tara Ali Yahiya. Resource allocation by pondering parameters for uplink system in LTE Networks. In *38th Annual IEEE Conference on Local Computer Networks*, pages 747–750. IEEE, Oct 2013.

[51] V Jacobson, R Braden, and D Borman. TCP Extensions for High Performance. RFC 1323, May 1992.

[52] Van Jacobson. Congestion avoidance and control. *ACM SIGCOMM Computer Communication Review*, 18(4):314–329, aug 1988.

[53] Rajendra Jain, Dah-Ming Chiu, and William R. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer system. In *DEC technical report TR301*, pages 1–38, 1984.

[54] Sam Jansen and Anthony McGregor. Simulation with Real World Network Stacks. In *Proceedings of the Winter Simulation Conference, 2005.*, pages 2454–2463. IEEE, 2005.

[55] Haiqing Jiang, Yaogong Wang, Kyunghan Lee, and Injong Rhee. Tackling bufferbloat in 3G/4G networks. In *Proceedings of the 2012 ACM conference on Internet measurement conference - IMC '12*, page 329, New York, New York, USA, 2012. ACM Press.

[56] K. Kanonakis and I. Tomkos. Offset-Based Scheduling With Flexible Intervals for Evolving GPON Networks. *Journal of Lightwave Technology*, 27(15):3259–3268, Aug 2009.

[57] Frank Kelly. Charging and rate control for elastic traffic. *European Transactions on Telecommunications*, 8(1):33–37, Jan 1997.

[58] Glen Kramer, B. Mukherjee, and G. Pesavento. IPACT a dynamic protocol for an Ethernet PON (EPON). *IEEE Communications Magazine*, 40(2):74–80, 2002.

[59] H C Leligoun, Ch. Linardakis, K. Kanonakis, J. D. Angelopoulos, Th. Orphanoudakis, H. C. Leligou, Ch. Linardakis, K. Kanonakis, J. D. Angelopou-

los, and Th. Orphanoudakis. Efficient medium arbitration of FSAN-compliant GPONs. *International Journal of Communication Systems*, 19(5):603–617, Jun 2006.

[60] Yee-Ting Li, Douglas Leith, Robert N. Shorten, Douglas Leigh, and Robert N. Shorten. Experimental Evaluation of TCP Protocols for High-Speed Networks. *IEEE/ACM Transactions on Networking*, 15(5):1109–1122, Oct 2007.

[61] Wansu Lim, Milos Milosavljevic, Pandelis Kourtessis, and M. John. QoS mapping for LTE backhauling over OFDMA-PONs. In *2012 14th International Conference on Transparent Optical Networks (ICTON)*, pages 1–4. IEEE, Jul 2012.

[62] Xin Liu, Ashwin Sridharan, Sridhar Machiraju, Mukund Seshadri, and Hui Zang. Experiences in a 3G Network : Interplay between the Wireless Channel and Applications. In *MobiCom*, pages 211–222, New York, New York, USA, 2008. ACM Press.

[63] Yong Liu. Profiling Skype video calls: Rate control and video quality. In *2012 Proceedings IEEE INFOCOM*, pages 621–629. IEEE, Mar 2012.

[64] Ralf Lubben and Markus Fidler. On characteristic features of the application level delay distribution of TCP congestion avoidance. In *2016 IEEE International Conference on Communications (ICC)*, pages 1–7. IEEE, May 2016.

[65] Yuanqiu Luo, Xiaoping Zhou, Frank Effenberger, Xuejin Yan, Guikai Peng, Yinbo Qian, and Yiran Ma. Time- and Wavelength-Division Multiplexed Passive Optical Network (TWDM-PON) for Next-Generation PON Stage 2 (NG-PON2). *Journal of Lightwave Technology*, 31(4):587–593, Feb 2013.

[66] Luis Marrone, Andres Barbieri, and Matias Robles. TCP Performance - CUBIC, Vegas & Reno. *Journal of Computer Science and Technology*, 13(1):1–8, Apr 2013.

[67] M Mathis, J Mahdavi, S Floyd, and A Romanow. TCP Selective Acknowledgment Options. RFC 2018, 1996.

[68] M.P. McGarry, M. Maier, and M. Reisslein. Ethernet PONs: a survey of dynamic bandwidth allocation (DBA) algorithms. *IEEE Communications Magazine*, 42(8):S8–15, aug 2004.

[69] Microsoft. How much bandwidth does Skype need? https://support.skype.com/en/faq/FA1417/, 2015.

[70] John E. Mitchell. Integrated Wireless Backhaul Over Optical Access Networks. *Journal of Lightwave Technology*, 32(20):3373–3382, Oct 2014.

[71] Binh Nguyen, Arijit Banerjee, Vijay Gopalakrishnan, Sneha Kasera, Seungjoon Lee, Aman Shaikh, and Jacobus Van der Merwe. Towards understanding TCP performance on LTE/EPC mobile networks. In *Proceedings of the 4th workshop*

on All things cellular: operations, applications, & challenges - AllThingsCellular '14, pages 41–46, New York, New York, USA, 2014. ACM Press.

[72] Hiroki Nishiyama, Zubair M. Fadlullah, and Nei Kato. Inter-Layer Fairness Problem in TCP Bandwidth Sharing in 10G-EPON. *IEEE Systems Journal*, 4(4):432–439, Dec 2010.

[73] Nokia. TWDM-PON: Taking Fiber to New Wavelengths. Available at: https://insight.nokia.com/twdm-pon-taking-fiber-new-wavelengths, 2016.

[74] Julio Orozco and David Ros. TCP Performance over Gigabit-Capable Passive Optical. In *Third International Conference on Access Networks*, volume 6, pages 264–279, 2009.

[75] T Orphanoudakis, E. Kosmatos, J. Angelopoulos, and A. Stavdas. Exploiting PONs for mobile backhaul. *IEEE Communications Magazine*, 51(2):S27–S34, Feb 2013.

[76] Theofanis G. Orphanoudakis, Evangelos A. Kosmatos, Chris Matrakidis, Alexandros Stavdas, and Helen-Catherine Leligou. Hybrid resource reservation scheme for transparent integration of access and core optical transport networks. In *2014 16th International Conference on Transparent Optical Networks (ICTON)*, pages 1–4. IEEE, Jul 2014.

[77] A. K. Parekh. A generalized processor sharing approach to flow control in integrated services networks. PhD Thesis, Dept. of Electrical Engineering and Computer Science, M.I.T., 1992.

[78] Zhiwen Peng and Pj Radcliffe. Modeling and simulation of Ethernet Passive Optical Network (EPON) experiment platform based on OPNET Modeler. In *2011 IEEE 3rd International Conference on Communication Software and Networks, ICCSN 2011*, pages 99–104, 2011.

[79] Giuseppe Piro, Nicola Baldo, and Marco Miozzo. An LTE module for the ns-3 network simulator. In *Proceedings of the 4th International ICST Conference on Simulation Tools and Techniques*, number March, pages 415–422. ACM, 2011.

[80] Jon Postel. Transmission Control Protocol - DARPA Internet Program Protocol Specification. RFC 793, Sep 1981.

[81] Chathurika Ranaweera, Elaine Wong, Christina Lim, and Ampalavanapillai Nirmalathas. Next generation optical-wireless converged network architectures. *IEEE Network*, 26(2):22–27, Mar 2012.

[82] V. Srinivasa Rao. Protocol Signaling Procedures in LTE. http://go.radisys.com/rs/radisys/images/paper-lte-protocol-signaling.pdf, Sep 2011.

[83] Julius Robson and NGMN Optimised Backhaul Project Group. Guidelines for LTE Backhaul Traffic Estimation. https://www.ngmn.org/uploads/media/NGMN_Whitepaper_Guideline_for_LTE_Backhaul_Traffic _Estimation.pdf, 2011.

[84] K. Salah, F. Al-Haidari, M.H. Omar, and A. Chaudhry. Statistical analysis of H.264 video frame size distribution. *IET Communications*, 5(14):1978–1986, Sep 2011.

[85] Antonios G Sarigiannidis, Maria Iloridou, Petros Nicopolitidis, Georgios Papadimitriou, Fotini-Niovi Pavlidou, Panagiotis G Sarigiannidis, Malamati D Louta, and Vasileios Vitsas. Architectures and Bandwidth Allocation Schemes for Hybrid Wireless-Optical Networks. *IEEE Commun. Surv. Tutorials*, 17(1):427–468, Jan 2015.

[86] Panagiotis Sarigiannidis, Dimitris Pliatsios, Theodoros Zygiridis, and Nikolaos Kantartzis. DAMA: A data mining forecasting DBA scheme for XG-PONs. In *2016 5th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, pages 1–4. IEEE, May 2016.

[87] M. Shreedhar, George Varghese, M. Shreedhar, and George Varghese. Efficient fair queueing using deficit round robin. In *Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication - SIGCOMM '95*, volume (4)25, pages 231–242, New York, New York, USA, 1995. ACM Press.

[88] Amanpreet Singh, Indika Abeywickrama, A. Konsgen, Xi Li, and Carmelita Goerg. Statistical analysis of traffic aggregation in LTE access networks. In *6th Joint IFIP Wireless and Mobile Networking Conference (WMNC)*, pages 1–4. IEEE, apr 2013.

[89] Huan Song, Byoung-Whi Kim, and Biswanath Mukherjee. Multi-thread polling: a dynamic bandwidth distribution scheme in long-reach PON. *IEEE Journal on Selected Areas in Communications*, 27(2):134–142, Feb 2009.

[90] Marko Sosic and Vladimir Stojanovic. Resolving poor TCP performance on high-speed long distance links — Overview and comparison of BIC, CUBIC and Hybla. In *2013 IEEE 11th International Symposium on Intelligent Systems and Informatics (SISY)*, pages 325–330. IEEE, Sep 2013.

[91] Kunwadee Sripanidkulchai, Bruce Maggs, and Hui Zhang. An Analysis of Live Streaming Workloads on the Internet. In *Proceedings of the 4th ACM SIGCOMM Conference on Internet Measurement*, IMC '04, pages 41–54, New York, NY, USA, 2004. ACM.

[92] W Stevens. TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms. RFC 2001, Apr 1997.

[93] David Stynes, Kenneth N. Brown, and Cormac J. Sreenan. Using opportunistic caching to improve the efficiency of handover in LTE with a PON access network backhaul. In *2014 IEEE 20th International Workshop on Local & Metropolitan Area Networks (LANMAN)*, pages 1–6. IEEE, May 2014.

[94] Riverbed Technology. Riverbed Modeler (earlier OPNET Modeler). Available at http://www.riverbed.com/gb/products/steelcentral/opnet.html, 2016.

[95] Tsung-Yu Tsai, Yao-liang Chung, and Zsehong Tsai. Introduction to Packet Scheduling Algorithms for Communication Networks. In *Communications and Networking*, pages 264–287. Sciyo, Sep 2010.

[96] Elias Weingartner, Hendrik vom Lehn, and Klaus Wehrle. A Performance Comparison of Recent Network Simulators. In *2009 IEEE International Conference on Communications*, pages 1–5. IEEE, jun 2009.

[97] Pedro Alvarez Xiuchao Wu, Jerome A. Arokkiam. XG-PON Simulation Module for NS-3. Available at http://sourceforge.net/projects/xgpon4ns3, 2014.

[98] Kun Yang, Shumao Ou, Ken Guild, and Hsiao-Hwa Chen. Convergence of ethernet PON and IEEE 802.16 broadband access networks and its QoS-aware dynamic bandwidth allocation scheme. *IEEE Journal on Selected Areas in Communications*, 27(2):101–116, Feb 2009.

[99] M. Yuksel, K.K. Ramakrishnan, S. Kalyanaraman, J.D. Houle, and R. Sadhvani. Required extra capacity: A comparative estimation of overprovisioning needed for a classless {IP} backbone. *Computer Networks*, 56(17):3723 – 3743, 2012.

[100] Yasir Zaki, Thushara Weerawardane, Carmelita Gorg, and Andreas Timm-Giel. Multi-QoS-Aware Fair Scheduling for LTE. In *2011 IEEE 73rd Veh. Technol. Conf. (VTC Spring)*, volume (2)51, pages 1–5. IEEE, May 2011.