A Long Reach Passive Optical Network Module for the NS-3 Simulator

Xiuchao Wu (wuxiuchao@gmail.com)
Department of Computer Science, University College Cork

Abstract

Long Reach Passive Optical Network (LR-PON) has been regarded as one of the key technologies for the future Internet access networks, and it becomes very important to study the performance issues that might arise with the deployment of LR-PON. In this report, we present the design of a LR-PON module for NS-3 that follows XG-PON standard from the FSAN working group of ITU-T. The aim is to provide a standard-compilant, configurable, and extensible module that can simulate LR-PON with reasonable speed and can support a wide range of research topics, such as improving the performance of LR-PON itself, studying the interactions between LR-PON and the upper-layer protocols, and investigating its integration with various wireless networks, etc.

I. Introduction

During the last few decades, we have witnessed the huge success of the Internet, which has changed our daily life significantly and has become one of the main economy engines. The infrastructure of the Internet has also evolved for providing better performance. In recent years, the speed of evolution becomes very fast due to the advances in optical technologies. Core networks of the Internet have evolved into all-optical networks and one optical fiber could provide huge amount of bandwidth through the matured DWDM (dense wavelength division multiplexing) technology. Instead of Mbps or Gbps, the bandwidth of a core network is now measured in Tbps (Tera-bps: 1000Gbps)[1] and it can provide almost un-limited bandwidth. Not only core networks, optical fibers have also been deployed in access networks to solve the *last/first mile* problem and provide high speed Internet access to end users. For instance, FTTx (Fiber To The Home/Building/Curb, etc.) networks, that are based on Passive Optical Network (PON) technology, have been widely deployed in Korea and Japan.

However, compared to ADSL and Cable that exploit the existing wires for telephone and TV, the much higher per-capital cost has hindered the deployment of FTTx in many areas. Fortunately, Long Reach PON (LR-PON) has emerged and it has been regarded as a feasible solution (both in technology and in economy) to bring optical fiber into access networks and provide high bandwidth to most of users in the world [1][2]. With the maturity and deployment of LR-PON, we can expect that most of landlines in access networks will be replaced by optical fibers for providing high speed Internet access. Since high speed Internet access provided by LR-PON could pave the way

[1] http://en.wikipedia.org/wiki/TPE_%28cable_system%29

for many bandwidth-intensive applications (IPTV, VoID, Video Conference, etc.) that might reform the Internet and the global economy, it is very important to study the issues arisen with the deployment of LR-PON.

In this report, the design of a LR-PON module (following XG-PON standard from the FSAN working group of ITU-T [3]) for NS-3 (the state of art network simulator [4]) is presented. In section II, we first introduce PON and LR-PON. We also present our vision of the future Internet access networks with the deployment of LR-PON. We then discuss the potential research opportunities and the feasible research platforms in section III. The motivations and the design principles of the LR-PON module for NS-3 are also presented in this course. In section IV, we introduce the details of XG-PON and discuss how it should be simulated. The design details of the LR-PON module for NS-3 are then presented in section V. Section VI concludes this report with a project plan for implementing the LR-PON module for NS-3.

II. PON, LR-PON, and The Future Internet Access Networks

Compared to copper, optical fiber can provide higher bandwidth in a longer distance. However, the deployment of optical fiber in access networks is severely hindered by the cost issue. Unlike core networks that are used by thousands/millions of users, the optical fibers and devices in a point-to-point optical access network cannot be shared among users and its per-capital cost is too high. In fact, *fiber in the last/first mile* is considered seriously only after the emergence of PON. In this section, we will introduce PON and LR-PON (the next generation PON). A vision of the future Internet access networks, in which LR-PON dominates the wired networks, will also be presented at the end of this section.

A. Passive Optical Network (PON)

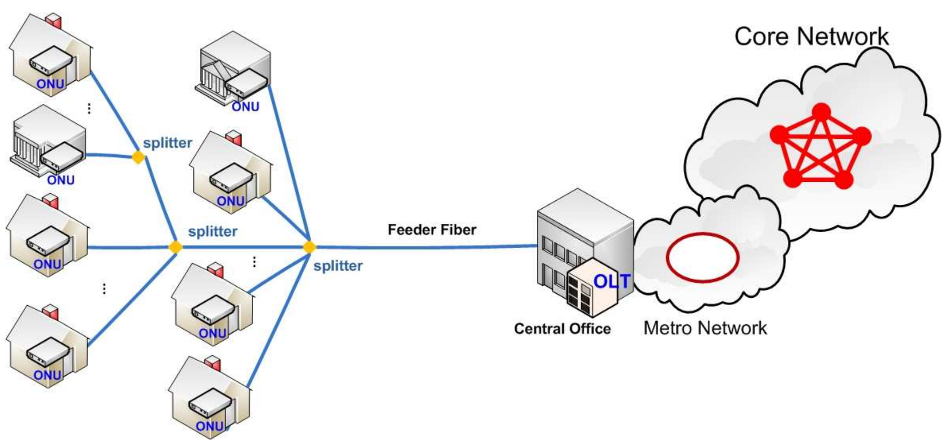


Fig. 1: An illustration of Passive Optical Network (PON)

Currently, PON is the dominant solution for *fiber in the last/first mile*. The existing FTTx networks are all based on PON technology. As shown in Figure 1, PON is a point-to-multipoint fiber network and there are three kinds of equipments in such a network: the OLT in central office, ONUs in/near customer premise, and passive optical splitters in the middle. Through passive optical splitters, the OLT (its optical interface included) and the feeder fiber are shared by multiple ONUs/users. Compared with the point to point architecture, PON can significantly reduce the amount of required optical fibers and the central office equipments. Since these splitters don't need power supply, the cost of deployment, maintenance and operation can also be reduced. In a word, PON reduces both capital expenditure (CapEx) and operational expenditure (OpEx) significantly. It could be a feasible way (in economy) for some countries to provide high speed Internet access to end users.

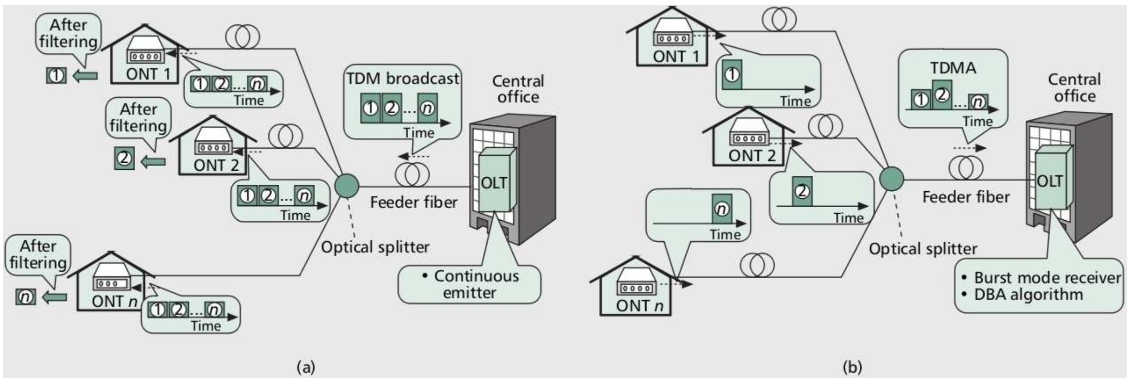


Fig. 2: PON principle in the a) downstream; b) upstream direction (Fig. 1 from [5])

As illustrated in Figure 2, downstream traffics are broadcasted by OLT to all ONUs that share a single optical fiber and encryption is used to prevent eavesdropping. Upstream traffics from ONUs are interleaved by the OLT for using the optical fiber in a TDMA-like manner. Please note that these ONUs normally have different distances, i.e., propagation delay, to the OLT. The OLT is responsible to schedule the data bursts from these ONUs for providing a collision-free and efficient upstream data communication. To accommodate the dynamics in bandwidth demands from users and exploit the gain of statistical multiplexing, DBA (dynamic bandwidth assignment) algorithm is normally used by the OLT.

PON has been standardized by both EFM of IEEE and FSAN of ITU-T. EPON from IEEE [6] is designed for carrying the naive Ethernet frames, and A/B-PON [7], the original PON standard from ITU-T, is designed for carrying ATM cells and ensuring various quality of service. Since the header overhead of the small ATM cell is very large and the ATM technology is obsoleted, ITU-T released GPON standard [8] which can carry various layer 2 traffics through encapsulation, such as Ethernet frames and ATM cells. The physical reach of the current PON-based networks is about 20km. EPON products can provide 1.25Gbps bandwidth in both directions and support 32 users per fiber. As for GPON products, the provided bandwidth is 2.488Gbps(downstream)/1.244(upstream) and the split ratio can be 64. Compared to EPON, GPON is more efficient and can provide higher bandwidth to more users [9]. However, GPON imposes more strict requirements on its hardware and its products are much more expensive than the EPON-based products. Due to the low cost and the simplicity of EPON, EPON has dominated the current market of optical access networks. For instance, FTTx networks, that are widely deployed in Korea and Japan, adopt EPON standard.

B. Long Reach PON (LR-PON)

Although PON technology can reduce the cost of optical access networks significantly, due to the small split ratio, the per-capital cost (even for EPON) is still much higher than ADSL and Cable that could exploit the existing wires for telephone and TV. Hence, the deployment of FTTx has been hindered in many areas.

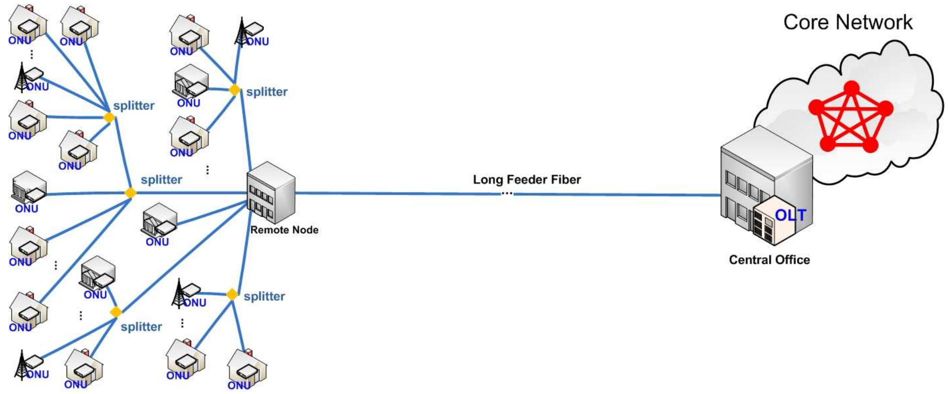


Fig. 3: An illustration of Long Reach PON (LR-PON)

Recently, Long-Reach PON (LR-PON) has been proposed to reduce the cost further through extending the physical reach and increasing the split ratio. One observation of LR-PON is that, due to the recent advances of optical technologies, a single wavelength can now provide 10Gbps bandwidth and a fiber can provide even higher bandwidth through DWDM. Consequently, the feeder fiber (and the OLT) can be shared by many more users (up to 1024) and per-capital cost can be reduced further. As shown in Figure 3, another key idea of LR-PON is to significantly increase the physical reach of the optical access network (up to 100km). Due to the much longer physical reach of LR-PON, the metro-network between core network and access network shown in Figure 1 can be removed. Through this kind of network consolidation, the access network can be simplified and its cost of operation and maintenance can be reduced.

To achieve the long reach and large split ratio, some active elements (optical amplifier, etc.) may be deployed between central office and remote node for fulfilling the necessary optical link budget[2]. Hence, LR-PON may not be a purely passive optical network again. Anyway, it does not matter since *passive* is just one technology, not the goal of optical access networks[3]. In [2], a 10-Gb/s 1024-Way-Split 100-km LR-PON has been demonstrated.

In summary, with LR-PON, it becomes feasible, both in technology and in economy, to bring optical fiber into access network and provide high bandwidth to most of users in the world [1]. Considering that high speed Internet access could pave the way for many potential exciting applications that could reform the Internet, hence the global economy, it is very important to study the issues that might arise when LR-PON is deployed.

[2]Just a few simple devices will be deployed in remote node in LR-PON and it is much cheaper compared with the central office in PON.

[3]In fact, *PON* has been redefined and the concept of *Reach Extender* is introduced in XG-PON [3], ITU-T standard for the next generation optical access network.

With the advances in optical communication, IEEE and ITU-T have released their standards for the next generation optical wireless network that can support 10Gbps, at least in downstream. Both 10G-EPON from IEEE [10] and XG-PON from ITU-T [3] define two architectures: symmetric and asymmetric. Considering that OLT must receive data burst from multiple ONUs with different distances, its receiver must operate in burst mode. Compared to the downstream, it's much harder to support 10Gbps in upstream and the asymmetric architecture is introduced for the short/middle term. In asymmetric architecture, the upstream bandwidth of 10G-EPON is 1Gbps and that of XG-PON1 is 2.5Gbps. Except the increased bandwidth, the split ratio is also increased by both of them (up to 256).

Among the two standards, XG-PON is closer to the concepts of LR-PON. *Reach Extender* has been introduced explicitly and its protocols need support the maximal fiber distance of 60km. XG-PON is also well standardized to support triple-play (voice, video, and data) and various quality of service. These features are very important to the future Internet access networks since the operators hope to use the same network to provide full service to end users. Furthermore, the higher upstream bandwidth provided by XG-PON1 could support many new scenarios, in which the bandwidth demand is more symmetric. For instance, XG-PON has been proposed to replace the current copper-based back-haul for base stations of cellular networks. Video conference is also a very promising applications for the future Internet. Although XG-PON might need more advanced hardware, the cost of XG-PON equipments could be reduced through the potential large-scale deployment in the future. Based on the above observations, XG-PON will be followed when we study LR-PON. Note that LR-PON and XG-PON are used interchangeably in the remaining parts of this report.

C. *The Future Internet Access Networks*

At the same time of the advances in optical communication, there are also a lot of improvements in wireless communication. Various wireless networks have been developed and deployed to enable users to access the Internet with satisfying experience anywhere and anytime. For instance, GPRS (177Kbps), 3G (2Mbps), LTE (100Mbps), and Wi-Max (40Mbps) have been developed for cellular networks to provide data services. Through OFDM (orthogonal frequency-division multiplexing) and MIMO (multiple-input and multiple-output), data rate of the under-developing 4G could be as high as multiple Gbps. Wireless LAN has also witnessed the evolution of Wi-Fi technology, from IEEE 802.11 (2Mbps) to 802.11b (11Mbps), 802.11a/g (54Mbps), and 802.11n (600Mbps). Cognitive radio, that could utilize radio spectrums more efficiently, has also emerged to provide better performance to mobile users. Just like the transition from fixed land-line telephone to mobile phone, it could also be a trillion-dollar business to fulfil mobile Internet access with these wireless technologies. Hence, it is also necessary to study the performance of mobile Internet access that may depend on various wireless technologies.

With the deployment of the above optical and wireless networks, we can expect that the future Internet access networks will be composed by LR-PON networks for providing high bandwidth to end users. Various wireless networks, that complement with each other in terms of bandwidth/coverage/energy efficiency, will also be attached to LR-PON for supporting mobile Internet access. As illustrated in Figure 4, LR-PON will be used to provide Internet access to PCs/Servers in the home, university, company, etc. Note that Wi-Fi access points and/or Femto-

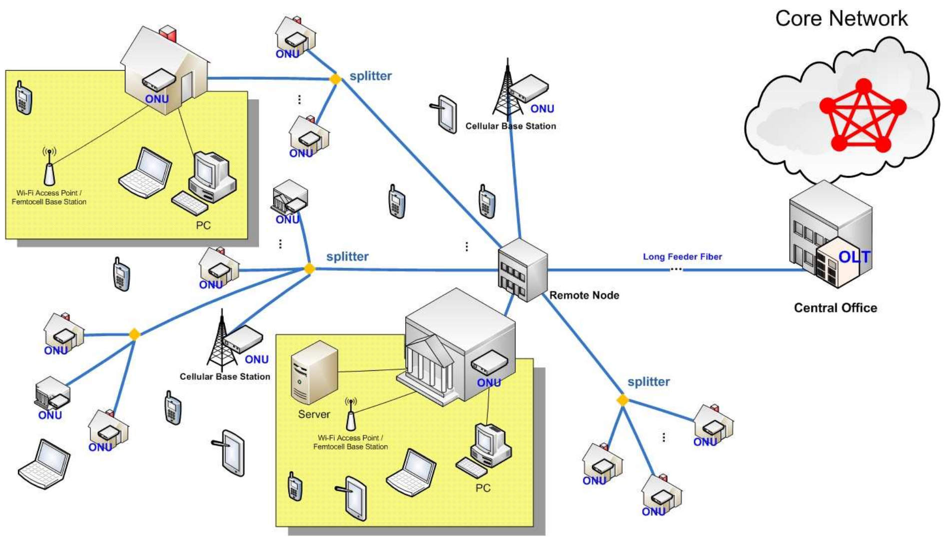


Fig. 4: An illustration of the future Internet access networks

cell base stations are normally installed in these places for mobile devices. LR-PON will also be used as the back-haul for base stations of cellular networks. Hence, mobile devices (Laptop, Tablet PC, smart phone, etc.) will access the Internet through various wireless networks (Wi-Fi, Femtocell, cellular network, etc.) plus LR-PON. It is very important to investigate the performance of different kinds of end users from the whole system point of view.

III. Research Opportunities and Research Platform

Based on the above vision of the future Internet access networks, we will first discuss the potential performance issues and research opportunities that might arise with the deployment of LR-PON. We then explain why NS-3 is selected as the research platform. At the end of this section, we present the principles that will be followed when designing the LR-PON module for NS-3.

A. Research Opportunities

Since LR-PON is still in its early stage, there are many issues to be studied to fulfil the promises of LR-PON. Under the scenario that LR-PON is deployed in the future Internet access networks, we will discuss several potential research opportunities in the following paragraphs. To efficiently and economically support high data rate, large split ratio, and long reach, it is very hot and important to carry out research on hardware devices and physical layer issues [2]. However, due to our background, we will focus on performance issues related with MAC and upper-layer protocols. More specifically, we will study the performance of LR-PON itself, its interactions with upper-layer protocols, and its integration with various wireless networks, etc. We believe that these researches are necessary parts for providing satisfactory user experience with LR-PON.

1) Performance of LR-PON Itself: Although the supported services, the frame structure, and the point to multi-point MAC protocol have been well standardized by XG-PON, there are still a lot of open issues related with its performance. For instance, product providers must determine how to allocate bandwidth and schedule traffics for various users with the aim of ensuring their QoS requirements[4]. They also need select the size of various buffers and the corresponding queue management mechanisms with considering both the cost and the performance.

One of the hottest topics in PON is the DBA algorithm used by the OLT for allocating upstream bandwidth to various users. DBA has been widely studied for EPON and these works have been surveyed in [11]. The challenges brought by LR-PON, especially its longer delay of the control loop, have also been studied in the context of EPON [12][13]. However, there are very few discussions about DBA algorithms for GPON. To our best knowledge, GiantMAC [14] and one variant of itself [15] may be the only proposals for GPON. Hence, starting from GiantMAC, it might be worthwhile to study the DBA algorithms for XG-PON. Not only the challenges brought by the long control delay, the large split ratio might also cause some problems. Since data bursts from ONUs are transmitted in upstream direction, some physical layer overhead (guard time, preamble, and delimiter) is needed between the consecutive data bursts from different ONUs. Hence, a large split ratio will introduce a larger overhead for upstream bandwidth allocation. Consequently, many issues must be considered when determining the frequency for allocating upstream bandwidth to one ONU. For instance, QoS requirements, upstream bandwidth utilization ratio, and the delay of control loop must be well balanced.

With the increase of bandwidth and split ratio, we can expect that one ONU in LR-PON will send a larger data burst with a lower frequency. If the closed-loop control is used by the upper layer protocol (TCP, etc.), the data traffic in downstream will also become very bursty. It is worthwhile to study the buffers used by OLT for answering some important questions. For example, what's the size of the buffer memory? How should the buffer memory be organized into the queues for a larger number of ONUs? Should we use a small queue for each ONU or let multiple ONUs share a larger queue[5]? Which queue management mechanism should be used? DropTail or RED (Random Early Detection)[16]?

2) Interactions with Upper Layer Protocols: As shown in Figure 4, LR-PON will be used to connect PCs and servers to the Internet and it is worthwhile to study the performance of TCP [17], the most widely used transport protocol of the Internet, in LR-PON[6]. Below, we will analyze the potential challenges faced by TCP in LR-PON and list several research problems to be answered.

Except the high bandwidth of LR-PON, its way of allocating the upstream bandwidth among users will bring many challenges to TCP. For simplicity, we assume that the upstream wavelength is allocated to ONUs in turn, and there is some physical layer overhead (T_{plo}) between the consecutive data bursts from different ONUs[7]. When deciding the size of a burst, one threshold (T_{max}) is normally used to avoid that one ONU monopolizes the upstream wavelength. Here, T_{max} is the longest time-slot that could be allocated to one ONU when it is scheduled, and it

[4]QoS parameters are normally deduced from customers' SLA (Service Level Agreement).

[5]In LR-PON, per-flow/per-traffic-class queue may be necessary for differentiating services provided to end users. However, if the expensive and power-hungry memory can be shared among multiple queues, it may be able to reduce the packets dropped at the OLT due to large data burst. The issue is how to achieve this sharing while avoiding packet re-ordering (which does hurt TCP performance).

[6]The research output should also be useful for other transport layer protocols, such as SCTP [18], DCCP [19], etc.

[7]In XG-PON1, T_{plo} is 256bits, i.e., 100ns for a 2.5Gbps upstream wavelength.

should be much larger than T_{plo} since the highest link utilization ratio is $\frac{T_{max}}{T_{plo}+T_{max}}$.

Due to the large split ratio (N), the interval between the consecutive upstream accesses of the same ONU can be very large when the load of upstream wavelength is high. The upstream delay in LR-PON could be as high as $N*(T_{plo}+T_{max})$ even when queue delay is not considered[8]. Since RTT experienced by TCP includes both the downstream and upstream delays in LR-PON, it may be increased significantly due to LR-PON. The jitter can also be very large. For instance, ACKs for the downstream traffics may be sent back immediately after the data segments are received in case that the upstream wavelength is currently granted to this ONU. Otherwise, these ACKs may wait for a long time until this ONU is allowed to access the upstream wavelength. Please note that these ACKs may be hindered further if some high-priority traffics arrive at that time.

We agree that GPON can assure low delay and small jitter for some high-priority traffics with the cost of low upstream bandwidth efficiency. Considering that TCP traffics are normally transmitted with the lowest priority (best-effort service), RTT and jitter experienced by TCP segments could be quite large[9]. Previous studies show that the delay in PON/LR-PON can be as high as 100 milliseconds and the jitter can be multiple tens milliseconds [5][20][21]. Furthermore, since ACKs for the downstream traffics tend to be sent back in a burst through the upstream wavelength of LR-PON, data segments will also arrive at the OLT in a burst and multiple segments may be dropped due to the large data burst (high data rate * long delay). In addition, due to the dynamics of both high-priority traffics and best-effort traffics, the amount of downstream/upstream bandwidth that could be allocated to a TCP connection may change abruptly. Due to these characteristics, the following issues should be studied.

- Could TCP and/or its high speed variants (Cubic TCP used by Linux [22], Compound TCP used by Microsoft Windows [23], etc.) can efficiently utilize the bandwidth provided by LR-PON under most of the potential scenarios? Due to the large BDP (bandwidth delay product), TCP may not work well. Due to the large jitter in LR-PON, we suspect that Compound TCP, which tries to exploit queue delay, cannot work well too. Cubic TCP may also face severe challenges when burst segment loss becomes common in LR-PON. In addition, in XG-PON1, the downstream bandwidth is 10Gbps and the upstream bandwidth is 2.5Gbps. Considering that bandwidth demand becomes more symmetric (Video conference, P2P applications, etc.) and the upstream bandwidth allocation is less insufficient, bandwidth asymmetry may also become one issue for TCP in LR-PON and should be investigated [24].
- How to set T_{max} and queue size at OLT to optimize the performance (the whole network and/or per user)? The larger T_{max} is, the higher the upstream bandwidth efficiency is. However, the delay for end users will be increased, segments may be dropped at OLT due to large data burst, and TCP senders may not send data fast enough to utilize the downstream wavelength of LR-PON. The relationship could be quite complex and should be well investigated. The tradeoffs among different QoS metrics should also be studied further. Latency and jitter could be modeled easily based on LR-PON parameters, such as N and T_{max}. Through extending the

[8]When traffic load is high, a packet must wait in the queue until multiple wavelength grants have been issued to the same ONU.

[9]In GPON, the QoS parameters' objectives for best effort service is (delay: 500ms, jitter: 40ms, plr: 10^{-4}).

standard queueing theory, we may also model packet drop rate with the queue size and the burstiness of coming packets at OLT (which also depends on N and T_{max}). Hence, we can jointly model TCP performance in LR-PON, identify the relationships among various parameters of TCP and LR-PON, carry out joint optimization, and find some good combinations of these parameters. One much simpler problem has been studied in [25]. Their assumption is that TCP performance solely depends on the small and fixed socket buffer size and RTT. The authors then optimize the network parameters of PON so that the allocated bandwidth and the delay in PON can match with each other. Note that their assumption isn't true in most cases.

- Due to the dynamics of network traffics, the available surplus bandwidth for best-effort service may change abruptly. We should study how to allocate surplus bandwidth among TCP connections and how to let TCP senders to utilize the available bandwidth as soon as possible (through some cross-layer techniques, maybe).

3) Integration with Wireless Networks: As discussed in section II, various wireless networks will be attached to LR-PON for supporting high speed and portable/mobile Internet access. For instance, in home and business where LR-PON is used for broadband Internet access, Wi-Fi access points are also installed widely to remove the wires and support portable computing. Femto-cell base-stations may also be deployed in these customer premises. The traffics of these users will pass through both LR-PON and Wi-Fi/Femto-cell. In the case that LR-PON is used by mobile telecom as the back-haul of their base stations, cellular data packets will pass through LR-PON and Wi-Max/LTE, etc. In a word, The integration between optical and wireless technologies will reform the future Internet access networks. There will be many research opportunities and several of them are listed below.

- Could TCP efficiently utilize the bandwidth provided by LR-PON plus various high speed wireless networks? This problem will arise when we use Laptops to access the Internet through Wi-Fi/Wi-Max/LTE plus LR-PON. Will the characteristics of LR-PON (long delay, large jitter, and bursty data traffic) nullify the existing TCP performance enhancements for wireless networks? Will the characteristics of wireless networks bring new challenges to the TCP variants proposed for high speed networks (Cubic TCP[22], Compound TCP[23], etc.)? Note that not only high packet corruption rate, the bandwidth in wireless networks will change with the time due to user mobility, flow dynamic, optimization in cognitive networks, etc.
- Considering that TCP throughput decreases with the increase of RTT and the delay of LR-PON could be significant, caching system should also be studied for *LR-PON plus wireless* access networks. Instead of OLT, should we deploy cache proxy at ONU which is much closer to mobile users? What's the tradeoff between benefits and costs? Through deploying cache proxy at ONUs, we can save the bandwidth of downstream wavelength. More importantly, we can speed up the downloading of mobile devices. Hence, their radios can be turned off sooner for saving energy. This is very important to battery-powered mobile devices, such as smart phones and tablets. If deployed at ONU, which kind of caching policy should be used for reducing cache size and increasing hit ratio?
- With the integration between optical and wireless networks, user traffics need consume both the resources of these networks. Should we coordinate the resource allocation in both LR-PON and wireless networks for

efficiently utilizing these resources, especially those of wireless networks?[10] Since the downstream of LR-PON and wireless networks are both broadcast medium, it should also be worthwhile to study how to support multi-cast and broadcast applications in a better way.

- Fundamentally, downstream traffics in LR-PON are broadcasted to all ONUs. In case that LR-PON is used as the back-haul of many base stations of a cellular network, it is worthwhile to study how to support user mobility in a better way. Packet Pre-caching may be a feasible and promising solution. Since base stations belong to the same operator, they can share the key and receive all traffics for mobile users. Hence, a base station could store the packets, whose destinations are mobile users in neighboring base stations, for a short while. In case that a mobile user does move in, seamless handoff can be achieved in a very short time. Although some control packets must be exchanged, the old base station need not forward data packets to the new base station any more. To exploit these benefits, we need to decide the size of buffer for pre-caching, the mobile users to pre-cache, the time to start pre-cache, etc. Of course, we also need study the architecture issues for Mobile IP [26] in the *LR-PON plus wireless* access networks. Furthermore, the way of assigning ONUs to the wavelength will become a challenging task. Based on user mobility, we should group the ONUs that are visited sequentially together (the same wavelength) for improving mobile Internet access through Packet Pre-caching. Based on the load distribution, we should put the ONUs, whose bandwidth demand can complement with each other, on the same wavelength for exploiting the gain of statistical multiplexing . Of course, the ONUs that could be put together should not be too far away from each other. Hence, there could be many variants of the ONU assignment problem, and a lot of researches need be carried out.

B. Research Platform

As discussed above, there are a lot of research opportunities to improve the performance of the future Internet access networks, i.e., LR-PON attached with various wireless networks. Although these research problems could and will be modeled and analyzed numerically, many simplifications must be made for tractability. Simulation, testbed, and/or in situ experiments must also be used to identify the problems and cross-validate with the modeling/analysis results. Below we discuss the platform to be used for studying the above research topics in *LR-PON plus wireless* access networks.

Considering that LR-PON is still in its early stage, it's impossible to study in the real networks. Although some devices in the lab-phase can be used to set up a LR-PON testbed in the near future, it is too expensive to get the necessary equipments for supporting our research. For example, to study DBA algorithms for LR-PON, hundreds

[10] Pedro also mentioned to let optical and wireless networks allocate resources jointly. Some collaboration may be possible. Pedro also plans to study whether the tree topology of XG-PON can support FTTCell well. Considering that the 125μs frame and propagation delay, the smallest delay may not be much less than 1ms. If it's acceptable, it may be worthwhile to study how many base stations can be supported by a wavelength? What's the tradeoff between delay and bandwidth (for upstream, high scheduling frequency can ensure short delay, but cause large bandwidth overhead.)? The last issue that Pedro wants to study is how to simplify the signaling protocols (inter-BaseStation) when LR-PON is used as the back-haul for base stations of cellular networks.

of ONUs are needed. Many high-performance PCs and several high-end routers are also needed to set up a 10Gbps network. If we study *LR-PON plus cellular networks*, Wi-Max and/or LTE products are also very expensive. Hence, simulation should be the only choice in the current phase.

There are quite a lot of network simulators, such as OPNET [27], QualNet [28], NS-2 [29], and NS-3 [4]. Among them, the commercialized OPNET and QualNet are well documented and supported. OPNET has also been widely used to study EPON and GPON [30][31]. However, the latest release of OPNET doesn't include any EPON or GPON model. These authors also don't want to release their models to the public. In addition, OPNET and QualNet are not open-source softwares. Their APIs may be too limited for our research. For example, many optimization, perhaps in the core of simulator (the scheduler, etc.), must be carried out for simulating a 10Gbps network with a reasonable speed. Even if we can develop a LR-PON module for them, it may be impossible to integrate with their models for Wi-Max, LTE, etc. Hence, open-source simulator (NS-2, NS-3, etc.) are preferred.

NS-2 is one of the most successful network simulators and it is widely used by the academic and industrial societies. NS-3 is a new simulator written from scratch. It is not evolved from NS-2 and there is no backward compatibility with NS-2. In fact, NS-3 is designed based on many lessons from NS-2 with the aim of replacing NS-2 (in long-term). Many new software engineering techniques are adopted in the development of NS-3 for ensuring the quality. For instance, *smart pointer* is used for avoiding memory leakage, *object aggregation* is adopted for flexibly defining nodes and protocol stacks, *attribute* is used for configuration, and *a call-back based trace system* is designed for debug and results collection. In NS-3, parallel simulation is supported so that a large network can be simulated on a cluster. Network simulation cradle (NSC) [32] is included so that the performance of the real OS's protocol stacks can be studied in simulation. Pcap trace format is also supported so that the existing tools (TcpTrace [33], Wireshark [34], etc.) can be used to analyze packet traces. Through *EmuNetDevice*, the NS-3 simulator can be integrated with testbed too. Although we are more familiar with NS-2, considering these good features, NS-3 is adopted as our platform for the above research topics[11].

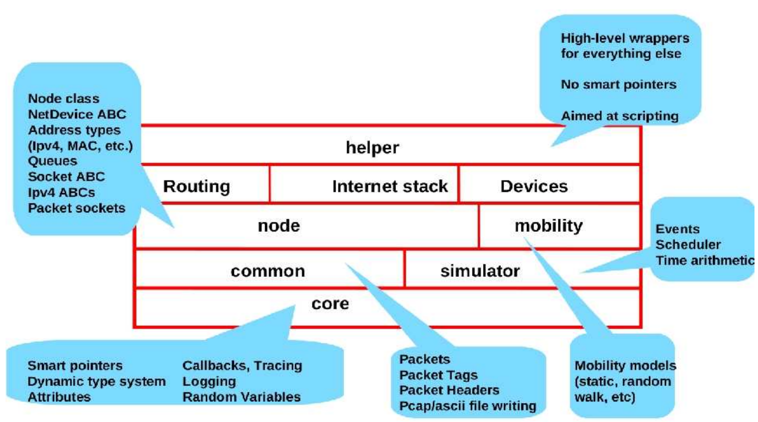


Fig. 5: The Organization of NS-3 Modules

Figure 5 shows the organization of NS-3 modules. The Internet protocol stack has been developed so that we can

[11]We note that it takes time to learn NS-3. The efforts should be worthwhile since NS-3 will be used widely in CTVR-II, a five-years project.

study the performance of TCP. Mobility models also exist for studying mobile Internet access. A lot of network devices have been developed for simulating different kinds of networks, such as point-to-point link, CSMA-based ethernet, Wi-Fi, Wi-Max, and LTE. Hence, NS-3 has included almost all of the modules required by us. The only missed one is a point to multi-point module for simulating LR-PON, and we plan to fill this gap by developing a LR-PON module for NS-3.

C. Design Principles for the LR-PON Module for NS-3

When designing the LR-PON module for NS-3, there are many issues to be considered. Below are the design principles followed by us in the course.

- Standard Compilance:

  The ultimate goal of our research is to improve the performance issues arisen with the deployment of LR-PON. It is highly desirable that the simulated LR-PON is close to the real LR-PON networks that will appear in the future. We can then identify the real problems and provide solutions that can be directly used in the real networks. Since there are still no mature LR-PON products, the best way is to simulate LR-PON based on some related standards. As discussed before, XG-PON is closer to the concepts of LR-PON and it is more likely to be deployed in th future. Hence, we will follow XG-PON when designing the LR-PON module.
- Simplicity:

  Considering that XG-PON is a quite complex standard, it will take a very long time to simulate the whole network, from physical layer to network management. Hence, we must decide the functions to be simulated in the first phase. We will only simulate the functions needed by our research. Other functions will be left alone or simulated as some stub functions. For instance, since we focus on MAC and upper layer issues, we may simulate the physical layer in a very simple way. We can assume that power budget for the optical fiber has been satisfied through various techniques. Hence, the reach extenders and passive optical splitters need not be simulated. The channel, that simulates downstream wavelength of LR-PON, can simply pass the same frames to all ONUs[12]. As for Forward Error Correction (FEC), instead of the exact algorithm, we can simulate only its effect, i.e., the bandwidth overhead and the much lower packet corruption rate.
- Extensibility:

  When designing the LR-PON module for NS-3, we should also consider its extensibility since many other research topics might also be studied with this module[13]. When designing the class architecture of the LR-PON module, abstract class should be used widely and the interface should be well designed for the future implementation that simulates more details. Of course, we will only provide a much simpler implementation for the functions that we don't care in current phase. For instance, when designing the class interface for the

[12]Based on my understanding, with considerations of both Pedros' research topics and ours, we don't need to simulate the details of the optical distribution network.

[13]Not only for ourself, we plan to release this module to the public and try to merge it into NS-3. Hence, the extensibility is very important. The style and quality of source codes will also follow the NS-3 project.

the channel that simulates the downstream wavelength of LR-PON, we should enable researchers to specify the tree structure of fibers, reach extenders, and splitters. When adding one ONU, they can also specify the splitter that it will be attached and the distance between them. With this interface, it is possible to simulate the optical signal propagation and the possible packet corruption. However, for the initial implementation, we can let the channel store a list of ONUs and send the frame to all of them (without any error)[14].

- Configurability:

  In a LR-PON, there could be thousands of computing devices, such as the OLT, hundreds of ONUs, and hundreds of data traffic generators/sinks. Many nodes will also be attached to ONUs through various networks and act as traffic generators/sinks. We should export many configurable parameters, but provide default parameters for most of them. Other methods should also be considered to easy the researcher's task for configuring the LR-PON to be simulated.
- Simulation Speed:

  Considering the LR-PON to be simulated is a 10Gbps network, simulation speed must be considered in all times. One module, that can simulate LR-PON accurately (but very slowly), is useless for our research in which extensive simulations are needed. We should select the data structures and algorithms carefully for saving CPU and memory. For example, instead of sending one individual copy of a downstream frame to each ONU, we may send the pointer of the same frame to all ONUs. For uni-cast traffic, each ONU will take away the packets for itself. Only for multi-cast packets, the ONU will make its own copy since multiple ONUs may operate on the same data in different ways. When studying TCP performance, instead of simulating TCP on hundreds of ONUs, we may simulate TCP on a few ONUs and let other ONUs (Dumb ONUs) generate traffics according to some distributions. Hence, most of packets won't pass through Internet protocol stack and the simulation can be speeded up[15]. Considering the complexity, Dumb ONUs will be implemented a little later, i.e., after checking the simulation speed without this simplification.

IV. XG-PON and Its Simulation

A series of standards for XG-PON have been released by FSAN of ITU-T. ITU-T G.987 lists the definitions, abbreviations, and acronyms used in this series for XG-PON. Some important concepts of XG-PON are also explained and clarified. In ITU-T G.987.1, the general requirements of XG-PON have been presented. The network architecture, migration and coexistence with GPON, services to be supported, hardware specifications, and the requirements for physical and MAC layers are all introduced briefly. ITU-T G.987.2 and ITU-T G.987.3 focus on

[14]Depend on the situation, it may be worthwhile to simulate a likely packet corruption rate, which should be very low with considering the use of FEC.

[15]We agree that the cross traffics generated based on some open-looped distributions are different with the traffics generated by TCP connections. However, even when all packets pass through the protocol stack, we still don't know whether we have simulated the realistic cross traffics. In fact, except the closed-loop control of TCP, the traffics are also affected by the behaviors of applications and users. Through changing the distributions followed by cross traffics, we can study TCP performance in various situations. Considering the speedup of simulation time, this simplification should be worthwhile.

the physical media dependent (PMD) layer and the transmission convergence (TC) layer, respectively[16]. Another related standard is ITU-T G.988, which specifies ONU management and control interface (OMCI) for both GPON and XG-PON. Since we are interested in the performance issues with LR-PON, the complex MIB (Management Information Base) and OMCI will not be implemented for both ONU and OLT. We will provide a simple interface for specifying the necessary configurations, such as Alloc-IDs and XGEM ports for each ONU and their QoS parameters. Figure 6 summarizes the common functions of XG-PON and the corresponding standards. In this section, we will introduce XG-PON with focus on the parts related with our research. During this course, we will also discuss how to simulate these features in NS-3.

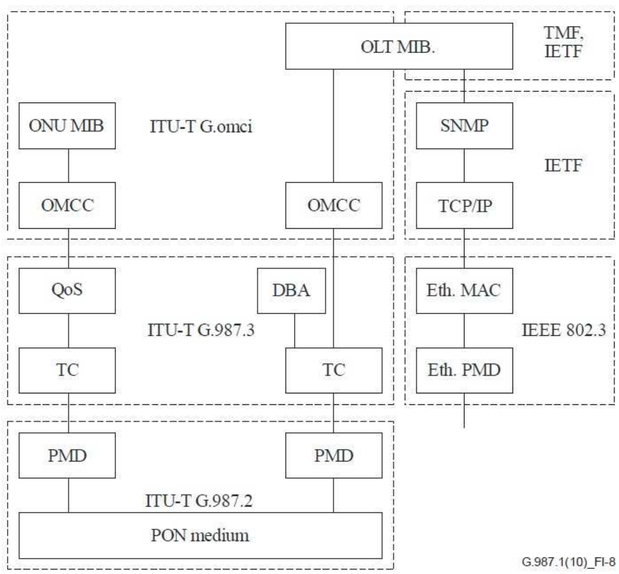


Fig. 6: XG-PON common functions

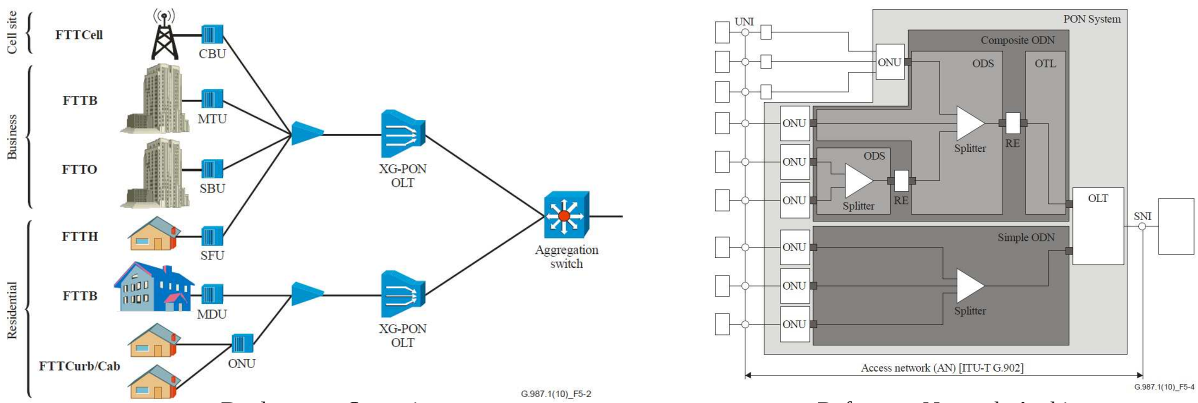
A. *Overview of XG-PON*

As illustrated in Figure 7(a), XG-PON has been proposed for various deployment scenarios. Not only the traditional Fiber to the Home, Fiber to the Curb, Fiber to the Building, etc., Fiber to the Cell has also been proposed so that XG-PON can be used as the back-haul for base stations of cellular networks. To support these deployment scenarios, XG-PON also lists the services to be provided, such as Telephony (VoIP and emulated/simulated POTS), TV (IPTV and Digital TV through RF-video overlay), Leased Line (T1, E1, etc.), High speed Internet access, Mobile backhaul, etc. Hence, XG-PON has been well standardized for providing full services to various users with one network. To provide these services, XG-PON must support various class of services with different QoS parameters.

As shown in Figure 7(a), corresponding to these deployment scenarios, XG-PON also introduces various ONU variants that may provide different functions and interfaces to users. For simplicity, we will simulate one kind of

[16]XG-PON TC layer is the focus of our LR-PON module for NS-3. Considering the simulation speed and research topics to be studied, we won't simulate the characteristics of optical fiber, splitter, laser, etc.

ONU. It is a node that is installed with a LR-PON network device. Other network devices will also be installed on this node for connecting with PCs/Laptops/Tablets/smartphones through Ethernet, Wi-Fi, Wi-Max, LTE, etc. As for different sites to be connected through LR-PON, we may simulate them through providing different quality of services to these ONUs.



(a) Deployment Scenarios (b) Reference Network Architecture

Fig. 7: Scenarios and Reference Architecture of XG-PON

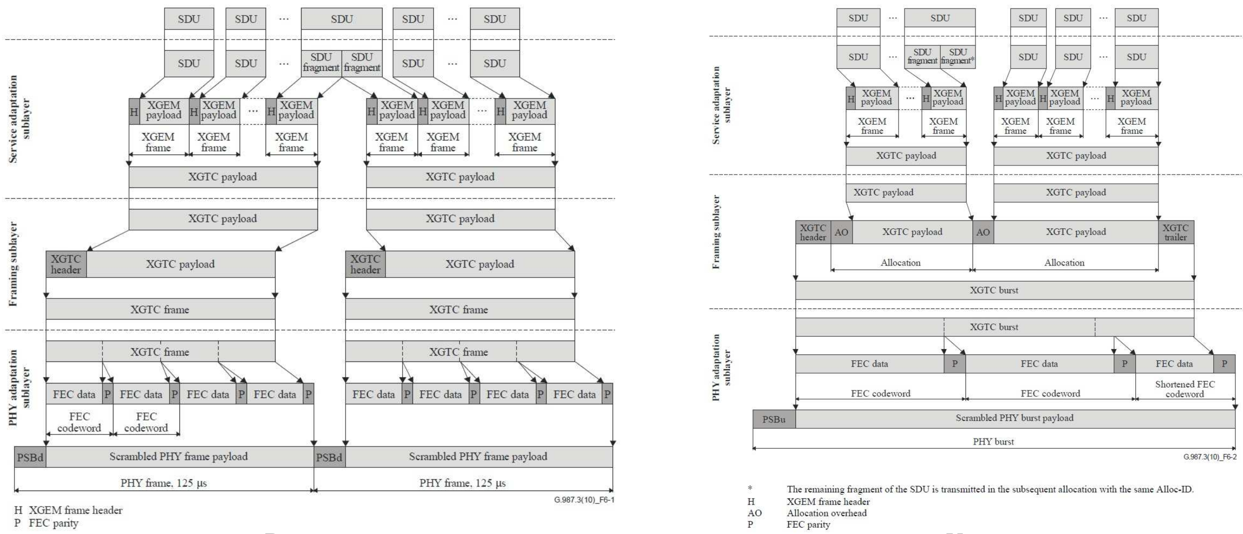
Figure 7(b) shows the reference access network architecture of XG-PON. In a composite ODN (optical distribution network), Reach Extender (RE) has been explicitly introduced to support longer distance and larger split ratio. Since we plan to study the issues related with MAC and upper layer protocols, RE and splitter won't be modeled explicitly. We won't simulate how the signal strength is changed when passing through fiber/RE/splitter and how the receiver detects/decodes the data. All components between OLT and ONU will be modeled as one channel with two wavelengths (downstream and upstream). With the assumptions that power budget is ensured through various techniques, the laser receiver can work well, and FEC (Forward Error Correction) is used, these wavelengths simply pass the frames/bursts to the intended receivers[17].

ITU-T G.987.2 presents many details about the PMD layer of XG-PON (XG-PON1, more specifically). It gives the wavelength to be used, the data rates to be supported, the requirements to fiber/laser/clock, etc. However, these issues are out of the scope of our simulation. The data rates (Downstream: 9.95328Gbps; Upstream: 2.48832Gbps) and the physical layer overhead for upstream bursts (256 bits) in ITU-T G.987.2 are the only information to be used. From now on, we will present the details of XGTC layer, the Transmission Convergence (TC) layer of XG-PON.

B. XGTC Layer Structure

As a part of XG-PON protocol stack, XGTC is composed by three sub-layers: service adaptation sublayer, framing sublayer, and PHY adaptation sublayer. Figure 8 shows how the SDUs of XG-PON are encapsulated into 125μs downstream frames or upstream bursts with various length.

[17] When designing the simulation for physical media, we will reserve the interfaces for simulating PON medium more accurately.



(a) Downstream (b) Upstream

Fig. 8: SDU Mapping in XG-PON

The main functions of PHY adaptation sublayer are FEC (forward error correction) and scrambling. Based on the planned research topics, we won't carry out any coding/decoding operations which are CPU-intensive. But the bandwidth overhead of FEC will be considered for simulating the correct throughput and studying performance issues. As for service adaptation sublayer, it is responsible for upper layer SDU encapsulation, multiplexing and delineation in the course of transmission over XG-PON.

On the transmitter side, the XGTC service adaptation sublayer accepts the upper layer SDUs (user data frames[18] and OMCI traffic), performs SDU fragmentation as necessary, assigns an XGEM Port-ID to an SDU or SDU fragment, and applies the XG-PON encapsulation method to it for obtaining an XGEM frame. The XGEM frame payload can be optionally encrypted. A series of XGEM frames form a payload of an XGTC frame in the downstream direction or an XGTC burst in the upstream direction. On the receiver side, the XGTC service adaptation sublayer accepts the payload of the XGTC frames/bursts, performs XGEM frame delineation, filters XGEM frames based on the XGEM Port-IDs, decrypts the XGEM payload if it has been encrypted by the transmitter, reassembles the fragmented SDUs, and delivers these SDUs to the respective clients.

In our simulation, XGEM header will be represented by a header added into the packet from upper layers. A XGTC frame or a burst will be represented by a structure with an array of such packets. Since a XGTC frame or burst can be transmitted in a short time, we will send the whole frame/burst to the intended receivers. In another word, the time sequence of the packets in the same frame/burst won't be simulated. Of course, the order of these packets must be specified to avoid packet re-ordering at the receiver side. Hence, all packets of a frame/burst can be scheduled by one event and the simulation can be speeded up. Encryption and header error correction functions

[18]Instead of the layer-2 frames expected by XG-PON standard, our LR-PON module will accept IP packets from the Internet protocol stack as the SDUs. We won't encapsulate IP packets in Ethernet frames and regard Ethernet frames as SDUs of XG-PON.

are not simulated too for the speed of simulation. Although fragmentation/reassemble are quite complex, we decide to implement it since without this function, the channel may be wasted a lot when packet size is large, especially for the short upstream burst transmitted in upstream direction. In addition, the design of packet in NS-3 can support fragmentation and reassemble well. Since the service adaptation sublayer deals with two types of SDUs, as illustrated in Figure 9, it can be logically decomposed into an XGEM engine, responsible for XGEM Port-ID multiplexing and filtering, and two service adapters: the user data adapter and the OMCI adapter. The user data adapter can be configured to accommodate a variety of upper layer transport interfaces. In our simulation, OMCI adapter is just a stub class and we will focus on the user data adapter that interacts with Internet protocol stack of NS-3 directly.

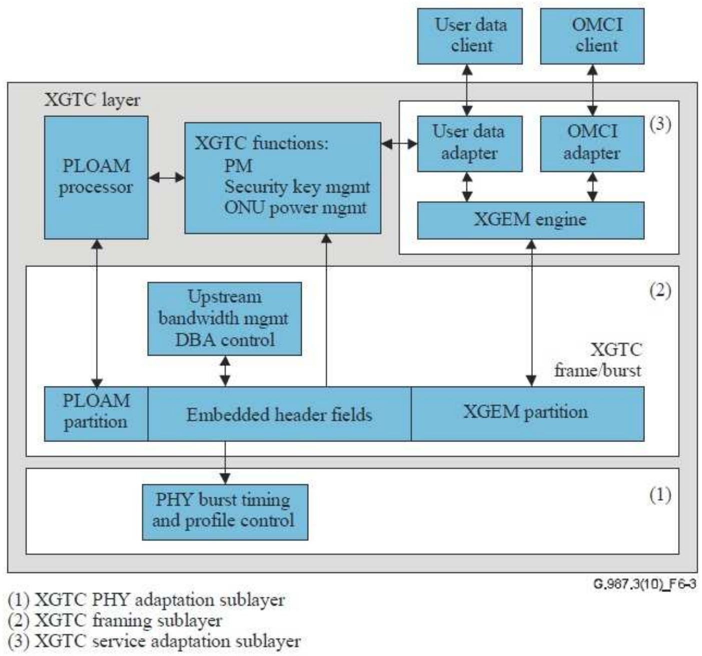


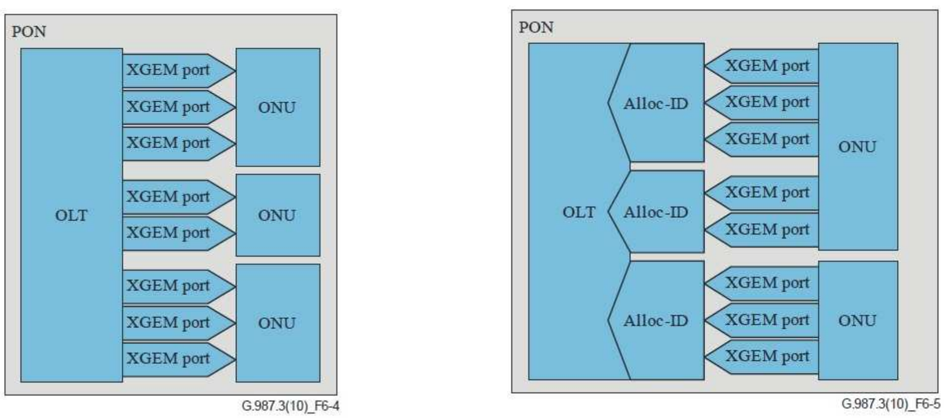
Fig. 9: XG-PON Information Flow

The XGTC framing sublayer is responsible for the construction and parsing of the overhead fields that support the necessary PON management functions. These frame formats are designed so that the frames and their elements are aligned to 4-byte word boundaries, whenever possible.

On the transmitter side, from the XGTC service adaptation sublayer, the XGTC framing sublayer accepts multiple XGEM frames as the XGTC payload, and constructs the downstream XGTC frame or upstream XGTC burst by providing embedded OAM and PLOAM messaging channel overhead fields. Hence, for a downstream XGTC frame, the size of its payload must be obtained by subtracting the variable size of the upstream bandwidth management overhead and the PLOAM channel load from its fixed size (125μs).

On the receiver side, the XGTC framing sublayer accepts the XGTC frames or XGTC bursts, parses the XGTC overhead fields, extracts the incoming embedded management and PLOAM messages, and delivers the XGTC payloads to the service adaptation sublayer. The incoming PLOAM messages are delivered to the PLOAM processing engine. The embedded OAM messages related with upstream bandwidth management (BW$_{map}$ parsing and dynamic bandwidth assignment) are processed within the framing sublayer itself. The OAM messages related with upstream PHY burst timing and profile control are sent to the PHY adaptation sublayer. As for the service adaptation sublayer, it will accept the OAM messages related with encryption key management. The rest of the embedded OAM messages are then delivered to the control entities outside of the framing sublayer. Figure 9 outlines the information flow related with XGTC layer. In our simulation, we will implement all of these logic. But we only implement the control messages needed by our study. For instance, the messages related with data encryption will be left alone. The OLT won't broadcast *Profile* message (physical layer information used by ONUs to generate their bursts correctly) too. The messages related with power savings will also be postponed[19].

Please note that XGEM port is used to identify a one-way logic connection (downstream or upstream), which is also configured with some QoS parameters. However, DBA algorithm allocates upstream bandwidth to Alloc-ID that belongs to a ONU. In XG-PON, one ONU may have multiple Alloc-IDs, and several connections (XGEM ports) may belong to the same Alloc-ID. In the upstream direction, an XGTC burst might multiplex XGTC payloads associated with multiple Alloc-IDs[20]. The size of each Alloc-ID's payload is determined based on the incoming bandwidth management information (BW_{map} in the header of downstream frames)[21].



(a) Downstream (b) Upstream

Fig. 10: Multiplexing in XG-PON

Figure 10 shows the multiplexing in XG-PON. The corresponding structures should be designed well for fast simulation since thousands of XGEM ports may be used in a LR-PON network. On the transmitter side, we must

[19]They should be added later for studying how to save the energy consumed by ONUs.

[20]Through multiplexing XGTC payloads associated with multiple Alloc-IDs of the same ONU, the physical layer overhead caused by each burst can be reduced.

[21]In the real network, Alloc-ID, XGEM port, and the QoS parameters should be configured and managed through OMCI interface. However, due to the complexity of implementing OMCI and MIB (Management Information Base) at OLT/ONUs, we will develop some helper class for configuring these parameters. Considering that a LR-PON could be composed by hundreds of ONUs and thousands of PCs, it is too tedious and inconvenient to configure each node by hand in the simulation codes. Hence, it should be better if we can find a way to specify LR-PON configuration (ONU-ID, Alloc-ID, XGEM Port, QoS parameters, IP address, etc.) more concisely (through some text files, maybe) and use a helper class to interpret and create the LR-PON network automatically.

lookup these tables to find the XGEM port for each IP packet and put the packet inside the corresponding queue. On the receiver side, table lookup must also be carried out to do the potential reassembling. With the large number of XGEM ports, these operations could consume a lot of CPU resources and many efforts are needed to speed up the simulation[22].

C. DBA, Scheduling, and Queue Management

Except data manipulation, there are still many important issues to let a large number of users share the bandwidth efficiently and fairly with respect of their QoS requirements. Hence, DBA, scheduling, and queue management are also very important parts of our LR-PON module for NS-3.

As mentioned early, XGEM port is used to identify a one-way connection with some QoS parameters. Hence, we should maintain one queue per XGEM port at the transmitter side[23]. DropTail mechanism will be used for queue management in the first phase, and we should support different queue management schemes and provide interfaces for configuration. For DBA, OLT will allocate upstream bandwidth to Alloc-IDs and for each Alloc-ID, OLT should maintain a virtual/logic queue (just the queue length reported by the ONU). In case that multiple XGEM ports are mapped to one Alloc-ID. OLT needs to calculate the aggregated QoS parameters for that Alloc-ID and carry out the resource allocation. In the first phase, we will implement GiantMAC [14] proposed for GPON as the baseline. As for scheduling schemes, we plan to first implement the priority-based scheduling scheme for both downstream scheduler at OLT and upstream scheduler at ONU, i.e., low priority packets are transmitted only when all high priority packets had been sent out. For flows with the same priority, round robin scheme will be used.

Since the existing LTE and Wi-Max modules were also designed for point to multi-point networks with considerations of Quality of Service, we will resort these modules when designing and implementing the corresponding functions. Many further studies will be carried out for simulating these functions in NS-3.

[22]IP addresses, ONU-IDs, Alloc-IDs, and XGEM ports should be allocated carefully so that we can use some simple functions to replace table lookup operation. At the first phase, we may apply the $1:1$ relationship between XGEM port and the computer connected to the ONU. For more flexibility, we may just apply the $1:1$ relationship between ONU-ID and the sub-net managed by this ONU. In that case, a fast table lookup becomes necessary for connections of the same ONU. This issue may should be re-investigated when considering user mobility and the changes of IP address. As for ONU, Alloc-ID, and XGEM port, we can apply a $1:M:N$ relationship. Here, M and N are set to some fixed integer values and N is a multiple of M. With these constraints, we can implement a simple classifier that can directly associate IP address of a computer with the XGEM port allocated to this computer. For the first phase, we may use $1:M:M$ (there is an one-to-one map between Alloc-ID and XGEM port) for simplifying the codes further. Hence, we just allow one XGEM port per Alloc-ID. The tasks of OLT and ONUs will be simplified. Especially for ONU, it need not decide which XGEM port to be served with the current upstream bandwidth allocation for an Alloc-ID.

[23]In principle, we should have another queue for reassembling at the receiver side. However, we plan to simplify it since the packet size won't be huge (without considering jumbo ethernet frame) and the remaining part of a fragmented packet will be transmitted immediately in the next frame/burst. Another reason is that we won't simulate the hardware/software boundary between XG-PON network interface card and internet protocol stack. In the real product, Internet protocol stack, especially IP, may be implemented in hardware and data traffics are handled by line card only (w/o CPU operations).

D. Timelines in XG-PON

In XG-PON, all ONUs are synchronized with OLT according to its continuous 125μs frames. Figure 11 illustrates how the start point of upstream frame is agreed among all devices. Since ONUs may have different propagation delay (with the OLT), different values of EqD must be used by them to have a common view and avoid collision in upstream direction. In the real products, each ONU will measure its propagation delay and calculate its EqD during the activation and ranging phase.

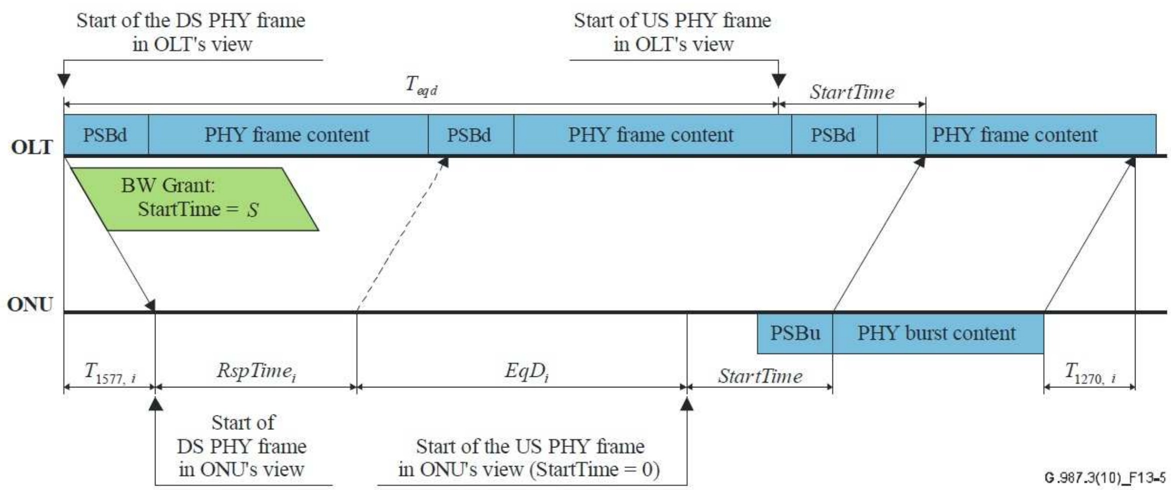


Fig. 11: Timelines in XG-PON

However, for the planned research, we need not simulate the distance and propagation delay for each ONU. We also assume that ONUs are always there. Hence, it may be fine to eliminate the activation and ranging procedures. For simplicity, we will let the channel for ODN send a downstream frame to all ONUs at the same time. All ONUs will add the same equalization delay (selected based on the maximal distance) for determining the start time of the upstream frame. Hence, they can transmit their bursts according to OLT's instructions without any collision in upstream direction. With this simplification, we can also leave the PLOAM messages related with activation/ranging alone.

Based on the above discussions, Table I summarizes the functions of XG-PON and how to implement them in NS-3 for simulating XG-PON.

Function	Layer	Choice	Comments
Signal Propagation & Line Coding & Frame Detection & Profile	PMD	X	Use one simple channel to simulate ODN that passes all frames/bursts to recipients. We only simulate propagation delay and bandwidth, and provide physical layer overhead information to DBA. *Profile* message won't be sent through ODN, but interface will be be designed.
FEC & Scrambling	PHY_ad	X	Just provide FEC overhead information to other components.
XGTC Framing	Framing	$\surd$	XGTC header, BW_map, and PLOAM will be simulated. Trailer and AO for upstream bursts are also simulated. *HEC* is left alone and we won't calculate CRC for saving CPU.
PLOAM Engine	Framing	$\surd$	Just implement the logic (passing messages among entities) w/o transmitting through ODN.
DBA & QoS	Framing	$\surd$	Might start from GiantMAC. It needs QoS parameters from control plane.
DS Scheduling & QoS	N/A	$\surd$	Simulate at OLT for generating continuous 125μs frames based on QoS parameters. In the first phase, priority-based and Round Robin schemes will be implemented.
US Scheduling & QoS	N/A	$\surd$	Simulate at ONUs for generating bursts with various length from XGEM ports of Alloc-IDs based on DBA. Priority-based and Round Robin schemes are also used here.
XGEM Framing	Service_ad	$\surd$	*HEC* is left alone. *Fragmentation and reassembling* will be implemented.
Encryption	Service_ad	X	Encryption will be removed for saving CPU. Interfaces for key management will be kept.
(De)Multiplexing	Service_ad	$\surd$	We will implement packet classification, flow management, etc. Alloc-ID, XGEM port, and IP address may be assigned carefully for simulation speed.
Queue Mechanism	Service_ad	$\surd$	Implement at both OLT and ONUs (one DropTail queue per XGEM port).
OMCI & MIB	N/A	X	Configuration will be carried out through some helper classes. Interface should be kept. I'm afraid that it will become necessary when studying the integration with wireless networks. XGEM ports and QoS parameters for one ONU may change with time due to user mobility.
Activation & Ranging	N/A	X	Leave it alone, but the interface will be designed.
Dumb ONUs	N/A	?	decide after checking the simulation speed.

TABLE I: Choices for Simulating XG-PON in NS-3

V. THE DESIGN OF THE XG-PON MODULE FOR NS-3

As stated above, the details of physical layer (signal propagation, frame detection, etc.) won't be simulated in the first phase. ODN (optical distribution network) will be implemented as a simple channel that just passes frames/bursts to their recipients. Optical devices (passive optical splitter, fiber, amplifier, etc.) won't be modeled explicitly. Of course, interfaces will be designed so that the ODN channel can be extended in the future and a split-tree/ring structure can be specified. Hence, to simulate a PON network in NS-3, we only need implement three main entities: OltNetDevice, OnuNetDevice, and OdnChannel. Figure 12 shows how these entities are used to simulate a PON network in NS-3. Note that this framework is independent to the variants of PON network. Through implementing different sub-classes, we can simulate EPON, GPON, XG-PON, etc.

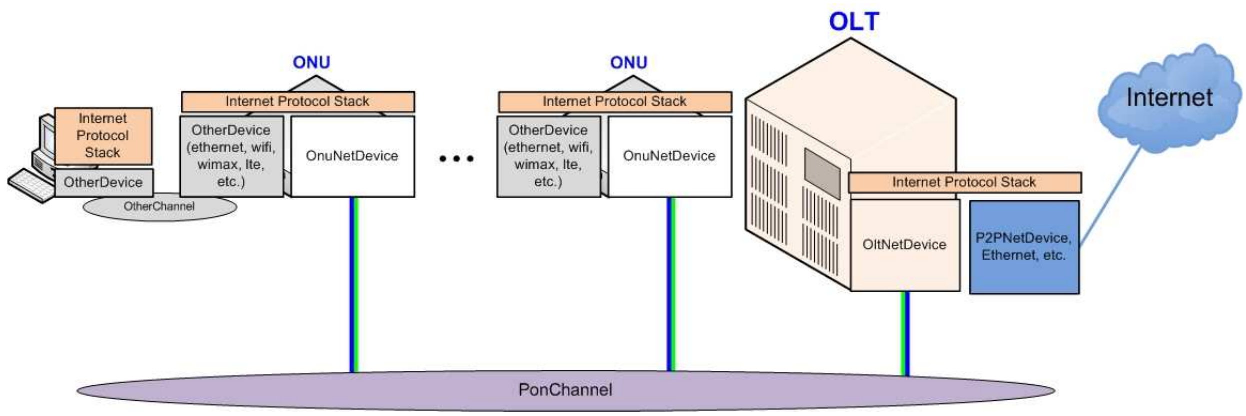


Fig. 12: The Entities of a PON Module for NS-3

In this section, we present the design of our XG-PON module for NS-3 which is composed by three main classes: XgponOltNetDevice, XgponOnuNetDevice, and XgponOdnChannel. We first describe its function block diagram. We then present the class diagram used to implement this XG-PON module.

A. Function Block Diagram

In this subsection, we will discuss the functions that are needed for simulating a XG-PON network in NS-3 through walking through how packets are exchanged between the OLT and one ONU.

1) Downstream Traffics at OLT Side: As shown in Figure 13, the green arrows show the downstream traffics. At the OLT side, when a packet is received from upper layers (Internet protocol stack or OMCI Manager), it will first be classified (based on IP address of the destination, etc.) and put into its corresponding queues. Hence, classification needs to be implemented for mapping between IP address and XGEM port that identifies a logical connection of XG-PON. In XG-PON, the OLT will send the 128μs downstream frames continuously. Hence, the main task of Downstream Scheduler at the OLT is to decide the connections to be served and the amount of data to be transmitted for each connection. The OLT Downstream Scheduler will resort to its *Downstream Connection Manager* which has the knowledge of queue length, QoS parameters, and the history of being served for each downstream connection.

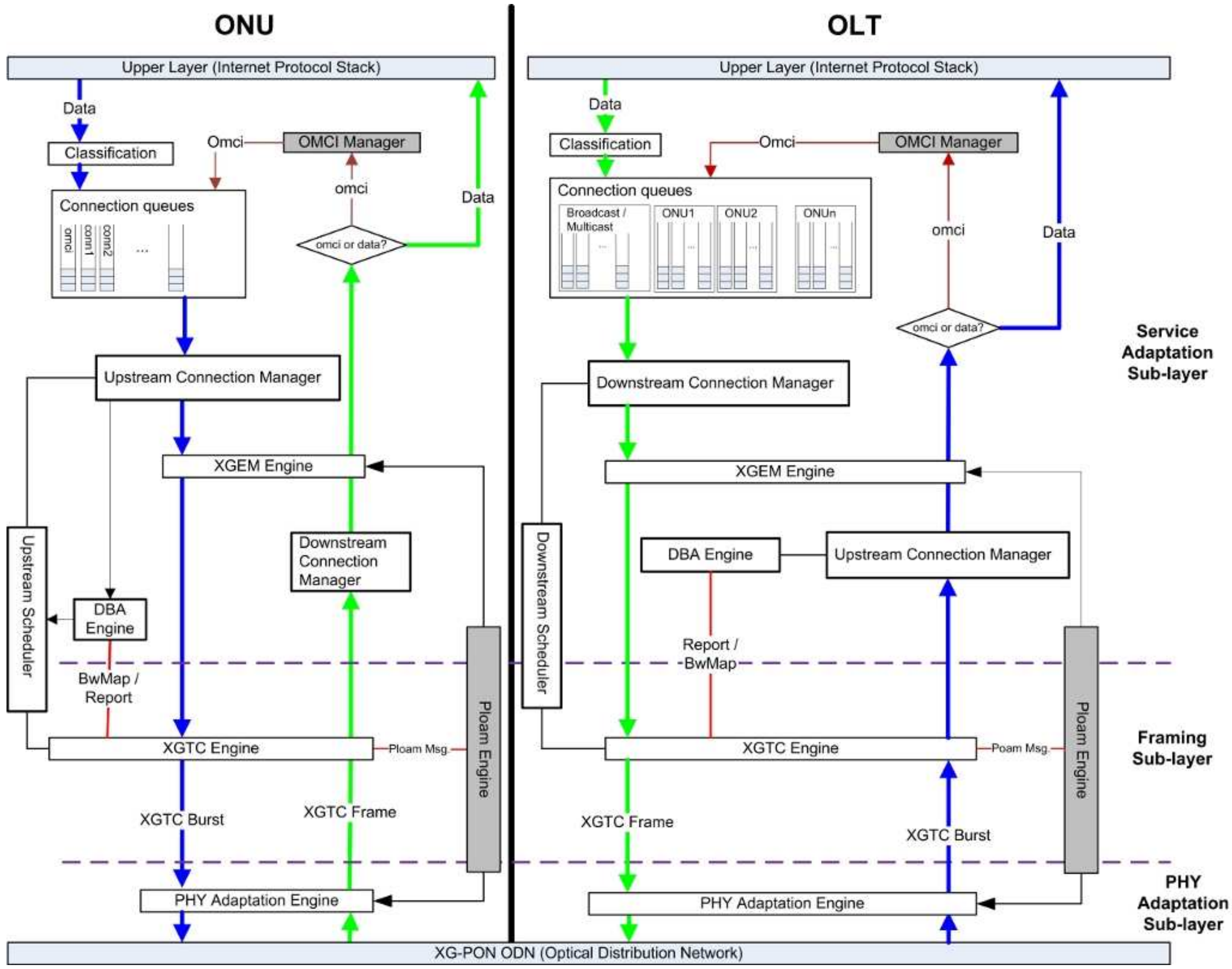


Fig. 13: Function Block Diagram of XG-PON

When generating the downstream frame through XGTC Engine, OLT will first calculate the length of payload with considerations of FEC and physical layer overhead, which can be gotten from PHY Adaption Engine. It then gets the BW$_{map}$ from DBA Engine. Note that DBA Engine generates BW$_{map}$ through resorting to Upstream Connection Manager, which has the knowledge of all upstream connections (QoS parameters, Serving history, and queue occupancy status). Note that for connections of the same Alloc-ID, their queue length will be aggregated and the sum is reported to the OLT[24]. The OLT also checks with its Ploam Engine for the potential PLOAM messages to be sent out for Activation, Ranging, Keying, and Burst Profile, etc. After all of these overheads are deduced, the OLT gets to know the amount of data that can be sent in this frame.

The OLT can then use the Downstream Scheduler to decide the connection to be served and the amount of its data to be transmitted. With XGEM Engine, these packets will first be encapsulated into a XGEM frame during which fragmentation might be executed. Note that encryption won't be carried out in the simulation. XGTC engine will put these XGEM frames into a XGTC frame.

[24]At the OLT side, it just maintains the amount of data queued at the ONU and there is no corresponding queue for holding the data.

Since FEC and scrambling are not executed in our simulation, the function of PHY Adaptation Engine is very simple. In fact, it just passes XGTC frames to the XG-PON ODN. In addition, it also tells others about the upstream/downstream bandwidth, physical layer overhead, and the overhead of FEC. After getting a XG-PON frame, the XG-PON ODN will wait for a while to simulate propagation delay and pass this frame to all ONUs (at the same time).

2) Downstream Traffics at ONU Side: When the ONU receives a downstream frame from the XG-PON ODN, its PHY Adaptation Engine just passes this frame to XGTC engine for parsing.

The PLOAM messages will be given to Ploam Engine, which is responsible for Activation, Ranging, Keying, and Burst Profile, etc. The Ploam Engine might forward the corresponding messages to XGEM engine (Keys) and PHY Adaptive Engine (burst Profile). The XGTC engine will pass the BW$_{map}$ (the request for status report included) to its DBA Engine. Based on BW$_{map}$ from its DBA Engine, the Upstream Scheduler can know the length of the upstream burst and when to send its upstream burst. Through refering to the Upstream Connection Manager, ONU can get to know the connections to be served in this burst.

As for XGEM frames, the XGTC engine will resort to the ONU Downstream Connection Manager to filter the XGEM frames for itself. These XGEM frames will be handled by XGEM engine, which may also cooperate with the Downstream Connection Manager for reassembling a packet. After removing XGEM header, the packet will be sent to the upper layers (Internet protocol stack or OMCI Manager).

3) Upstream Traffics at ONU Side: As shown in Figure 13, the blue arrows show the upstream traffics. At the ONU side, when a packet is received from upper layers (Internet protocol stack or OMCI Manager), it will first be classified (based on IP address of the source, etc.) and put into its corresponding queues. Hence, we also need implement a similar classifier for the ONU.

In XG-PON, the ONU mainly sends burst according to the bandwidth assignment from the OLT[25]. When it is this ONU's turn for transmission in upstream direction, the ONU scheduler will resort to its *DBA Engine* which has the knowledge of the amount of bandwidth (burst size) granted to each Alloc-ID of this ONU. Through resorting to *Upstream Connection Manager*, which knows the queue length, QoS parameters, and the history of being served of each upstream connection, the Upstream Scheduler can decide the packets to be transmitted in this burst. In case that there is just one XGEM port per Alloc-ID, the function of Upstream Scheduler will become very simple. These packets will be first encapsulated into a XGEM frame during which fragmentation might be executed. XGTC engine is responsible to put these XGEM frames into a XGTC burst. The PHY Adaptation Engine just passes XGTC bursts to the XG-PON ODN that will send the burst to the OLT after a while (to simualate propagation delay of XG-PON ODN).

Note that when generating the upstream burst, the ONU will also first calculate the payload length based on the overheads of FEC, physical layer, and the potential queue occupancy status report (from its *DBA Engine*) and PLOAM messages (from its Ploam Engine).

[25]One ONU may also send some data in the quiet window for activation and ranging

4) Upstream Traffics at OLT Side: When the OLT receives an upstream burst from the XG-PON ODN, its PHY Adaptation Engine just passes this burst to XGTC engine for parsing. The potential PLOAM messages will be forwarded to Ploam Engine, and the potential queue occupancy status report will be given to DBA Engine.

As for XGEM frames, they will be given to XGEM engine, which might resort to its Upstream Connection Manager for reassembling since it maintains the upstream connections at the OLT. After removing XGEM header, the packet will be sent to the upper layers (Internet protocol stack or OMCI Manager).

B. Class Diagram

Figure 14 shows the preliminary class diagram designed for the XG-PON module. It illustrates the relationship among the main classes of this module. XgponOltNetDevice, XgponOnuNetDevice, XgponPhy, XgponChannel, and XgponHelper are the five classes that will be used by the researchers. XgponOltNetDevice, XgponOnuNetDevice, and XgponChannel are sub-classes of some NS-3 core classes for network device and communication channel. XgponPhy is designed to simulate different physical layer specifications (data rate, etc.) supported by XG-PON. The researchers need attach XgponOltNetDevice and XgponOnuNetDevice to XgponChannel through XgponPhy. XgponHelper is a helper class to alleviate their jobs to configure a XG-PON with hundreds of nodes. In the following paragraphs, we will describe the functions of main classes in Figure 14.

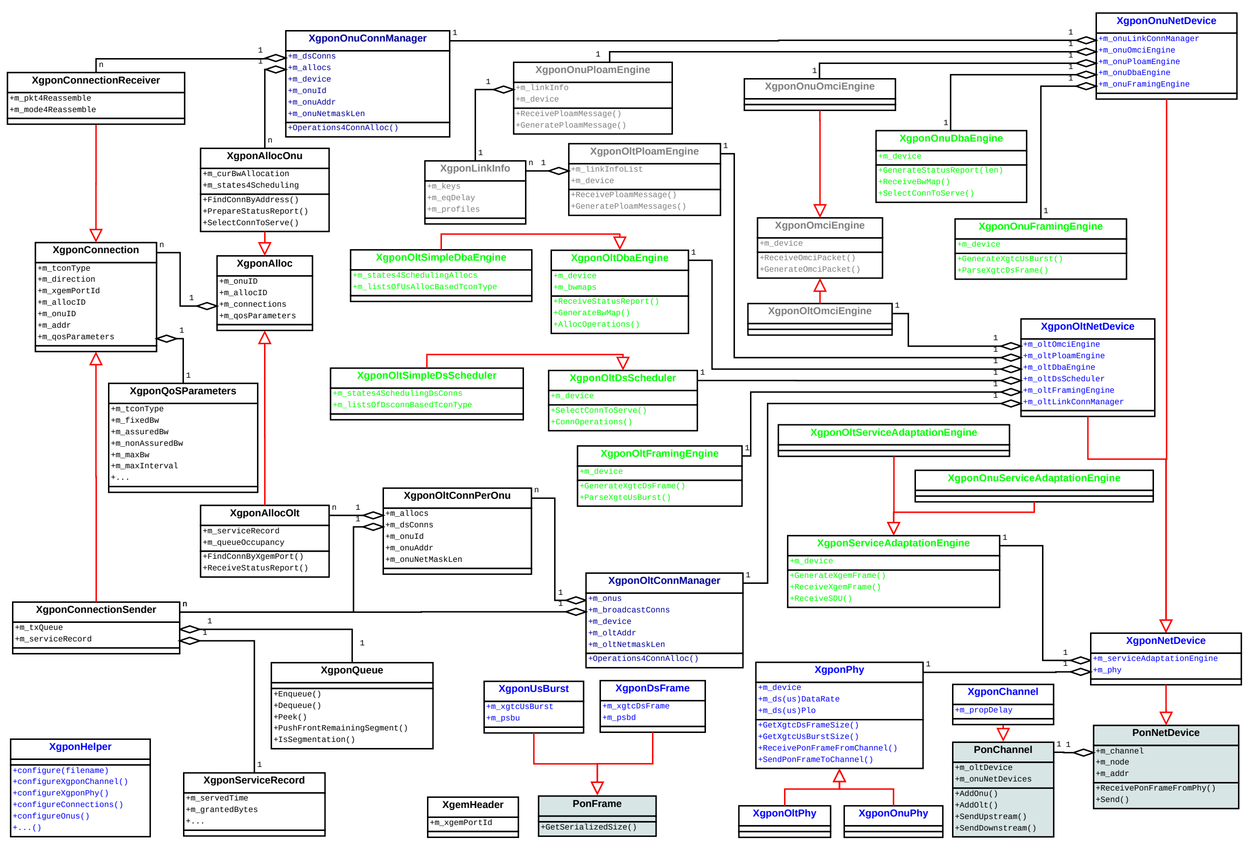


Fig. 14: Class Diagram of the XG-PON module for NS-3

1) PonFrame, PonChannel, PonPhy and PonNetdevice are designed as the framework for any PON that is based on Point-to-MultiPoint protocol.
2) XgemHeader: This class is designed to represent the XGEM header. It will be added to a packet from upper layers to form a XGEM frame.
3) XgtcDsFrame and XgtcUsBurst: XgtcDsFrame is designed to represent the XGTC frame in downstream direction. It normally contains an array of XGEM frames, XGTC header, BW$_{map}$, and the potential PLOAM messages. XgtcUsBurst is designed to represent the XGTC burst in upstream direction. It contains an array of XGEM frames, XGTC header/trailer, queue occupancy status report of Alloc-ID, and the potential PLOAM message. Many other classes are also designed to represent different parts of XgtcDsFrame and XgtcUsBurst.
4) XgponChannel: It is designed as a simple XG-PON ODN shown in Figure 13. It just simulate the propagation delay with the simplification that all ONUs have the same distance to the OLT. Its superclass (PonChannel) will simulate how the data is exchanged between OLT and ONUs.
5) XgponPhy: It is one subclass of PonPhy for XG-PON. It fulfils the functions of Physical Adaptation Engine in Figure 13. Each PonNetDevice has one PonPhy that is used to attach network device to communication channel. PonPhy simply passes data between the channel and network device. The main responsibility is to define the data rates, physical layer overhead and FEC overhead. Various subclasses can be used to simulate different physical layer specifications of XG-PON.
6) XgponNetDevice: It is the super class of XgponOltNetDevice and XgponOnuNetDevice. XgponXgemEngine is the only new member since both OLT and ONU need to carry out XGEM (de)encapsulation.
7) XgponXgemEngine: It carries out XGEM (de)encapsulation and segmentation/reassemble. It instantiates the XGEM Engine in Figure 13.
8) XgponLinkInfo: In our simulation, a link is the logical channel between OLT and one ONU, and a connection is the logical channel that carries data traffic between OLT and one client attached to one ONU. Note that one ONU could have many connections. XgponLinkInfo is designed to hold the information related with one ONU, such as keys, burst profiles, equalization delay, etc.
9) XgponConnection: This class is designed to represent one connection in XG-PON. This abstract class mainly contains the IDs related with a connection, such as XGEM-Port, ONU-ID, Alloc-ID (upstream connection only), and IP address of the client that this connection is setup for.
10) XgponConnectionSender: This subclass of XgponConnection is designed as the representation of a connection at the sender-side. It is used to represent an upstream connection at the ONU side or a downstream connection at the OLT side. It contains one queue for packets to be transmitted, the QoS parameters of this connection, and the service record that are used for scheduling.
    a) XgponQueue: It is the queue that holds the packets to be transmitted. Currently, it follows DropTaiQueue with consideration of segmentation.
    b) XgponQoSParameters: It is designed based on T-CONs supported by XG-PON.

c) XgponServiceRecord: It holds the last time that this connection is served and the amount of data that had been transmitted.

11) XgponConnectionReceiver: This subclass of XgponConnection is designed as the representation of a connection at the receiver-side. It is used to represent an upstream connection at the OLT side or a downstream connection at the ONU side.It just contains the information for reassembling.

12) XgponAlloc, XgponAllocOlt, and XgponAllocOnu: In XG-PON, upstream connections are organized into Alloc-TD. For connections of the same Alloc-ID, they should have the same T-CON type. Hence, XgponAlloc should have a list of connections and the aggregated QoS parameters of these connections.

    XgponAllocOlt is the representation of XgponAlloc at OLT. For DBA purpose, it should hold the queue occupancy report from ONU. It should also maintain the current bandwidth allocation for this Alloc-ID (which also indicates whether queue occupancy report will be received) and the time that this allocation is sent to ONU. In case that multiple allocations are granted in one RTT (LR-PON), we may need maintain a list of bandwidth allocations.

    XgponAllocOnu is the representation of XgponAlloc at ONU. For DBA purpose, it should hold the current bandwidth allocation for this Alloc-ID and have the operation for preparing queue occupancy report. It also contains the necessary variables and operations for scheduling connections of the same Alooc-ID. Since it has multiple connections, it also carry out some (de)multiplexing functions.

13) XgponOnuNetDevice: This class is used to simulate the network interface card used by ONU. It is the core of our module for ONU. It fulfils ONU XGTC Engine in Figure 13. The main functions are to generate the upstream burst and parse the downstream frame. To fulfil these functions, it contains the following entities.

    a) XgponOnuConnManager: This class contains the list of its downstream connections and the list of its Alloc-IDs in which upstream connections are organized. Hence, this class fulfil the functions of ONU Downstream/Upstream Connection Manager and the classifier (for upstream traffics) in Figure 13.
    b) XgponOnuDbaEngine: This class is designed to realize the DBA Engine in Figure 13. Since the ONU Upstream Scheduler is tightly coupled with DBA algorithm, it is also instantiated in XgponOnuDbaEngine.
    c) XgponOnuPloamEngine: This class is designed to handle PLOAM messages. It is responsible to maintain ONU-related information. In current phase, it is just used to implement the related logics. The PLOAM messages are not simulated.
    d) XgponOnuOmciEngine: This class is designed to process and generate OMCI packets. In current phase, they are just used to implement the related logics.

14) XgponOltNetDevice: This class is used to simulate the network interface card used by OLT. It is the core of our module for OLT. It fulfils OLT XGTC Engine in Figure 13. The main functions are to generate downstream frame and parse the upstream burst. To fulfil these functions, it contains the following entities.

    a) XgponOltConnManager: It is designed to manage all downstream connections (broadcast connections included) and the list of all Alloc-IDs in which upstream connections are organized. Hence, this class fulfil the functions of OLT Downstream/Upstream Connection Manager and the classifier (for downstream traffics) in Figure 13. For each ONU, XgponOltConnPerOnu is used to maintain its downstream and upstream connections. These XgponOltConnPerOnu are organized as one vector and the index should equal to the Onu-ID for fast look-up. All downstream connections and upstream XgponAllocs are also organized into several lists based on T-CON type for scheduling.

b) XgponOltDsScheduler: This class is designed to realize the scheduling function of OLT in Figure 13. It is responsible to schedule the traffics of all downstream connections.
c) XgponOltDbaEngine: This class is designed to realize the OLT DBA Engine in Figure 13. It is responsible to handle queue occupancy status report and generate BW$_{map}$ through refering to XgponAllocs maintained in XgponOltConnManager.
d) XgponOltPloamEngine: This class is designed to handle PLOAM messages. It is responsible to maintain ONUs' related information. In current phase, it is just used to implement the related logics. The PLOAM messages are not simulated.
e) XgponOltOmciEngine: This class is designed to process and generate OMCI packets. In current phase, they are just used to implement the related logics.

15) XgponHelper: This class is designed as a helper class. It is responsible to configure the whole XG-PON network. We plan to let it interpret a configuration file, generate the whole network, and pass related information to different entities used by OLT and ONUs. When it assigns XGEM port, ONU-ID, and IP address, the classification algorithm used for mapping between XGEM port and IP address should be considered to speed up the simulation.

In this section, we have briefly presented the functions and the relationship of the main classes designed for simulating XGPON in NS-3. The detailed interfaces, data structures, and algorithms will be refined in the course of code development. Below is the list of current source files.

C. *List of Source Files*

1) Files from Pedro: odn-channel.cc-h, olt-net-device.cc-h, onu-net-device.cc-h, and pedro_xgpon-net-device.cc-h. Some codes should be merged into the new design.

2) pon-channel.h: the definition of PonChannel
3) pon-frame.h: the definition of PonFrame
4) pon-net-device.h: the definition of PonNetDevice
5) pon-phy.h: the definition of PonPhy

6) xgpon-channel.h: the definition of XgponChannel

7) xgpon-phy.h: the definition of XgponPhy
8) xgpon-net-device.h: the definition of XgponNetDevice

9) xgpon-xgtc-ploam.h: the definition of XgponXgtcPloam, the PLOAM message
10) xgpon-xgem-header.cc-h: the definition and implementation of XgponXgemHeader
11) xgpon-xgem-engine.h: the definition of XgponXgemEngine

12) xgpon-xgtc-us-burst.h: the definition of XgponXgtcUsBurst
13) xgpon-xgtc-us-header.h: the definition of XgponXgtcUsHeader, the header of upstream burst.
14) xgpon-xgtc-dbru.h: the definition of XgponXgtcDbru, queue occupancy report.
15) xgpon-xgtc-us-allocation.h: the definition of XgponXgtcUsAllocation, packets of one Alloc-ID and potential queue occupancy report.

16) xgpon-xgtc-ds-frame.h: the definition of XgponXgtcDsFrame
17) xgpon-xgtc-ds-header.h: the definition of XgponXgtcDsHeader, the header of downstream frame.
18) xgpon-xgtc-bw-allocation.h: the definition of XgponXgtcBwAllocation, bandwidth allocation for one Alloc-ID.

19) xgpon-connection.h: the definition of XgponConnection, XgponConnectionSender, and XgponConnectionReceiver
20) xgpon-queue.h: the definition of XgponQueue
21) xgpon-qos-parameters.h: the definition of XgponQosParameters
22) xgpon-service-record.h: the definition of XgponServiceRecord
23) xgpon-alloc.h: the definition of XgponAlloc, XgponAllocOlt, and XgponAllocOnu
24) xgpon-onu-conn-manager.h: the definition of XgponOnuConnManager
25) xgpon-olt-conn-manager.h: the definition of XgponOltConnManager
26) xgpon-olt-conn-per-onu.h: the definition of XgponOltConnPerOnu

27) xgpon-ploam-engine.h: the definition of XgponOltPloamEngine and XgponOnuPloamEngine
28) xgpon-link-info.h: the definition of XgponLinkInfo and other related structures
29) xgpon-omci-engine.h: the definition of XgponOltOmciEngine and XgponOnuOmciEngine

30) xgpon-olt-ds-scheduler.h: the definition of XgponOltDsScheduler
31) xgpon-olt-dba-engine.h : the definition of XgponOltDbaEngine
32) xgpon-onu-dba-engine.h : the definition of XgponOnuDbaEngine

33) xgpon-olt-net-device.h : the definition of XgponOltNetDevice

34) xgpon-onu-net-device.h : the definition of XgponOnuNetDevice

VI. Project Roadmap

Since the LR-PON module for NS-3 is needed by the following research, we will put more efforts on it in the near term. We hope that a detailed design will be ready in several weeks and the preliminary implementation can be done in three months. This first version should be enough to study TCP performance in LR-PON. After that, we can start to integrate this LR-PON module with the existing Wi-Max and LTE modules for studying the *LR-PON plus wireless* access networks.

References

[1] D. B. Payne and R. P. Davey, "The future of fibre access systems?" *BT Technology Journal*, vol. 20, no. 4, pp. 104–114, October 2002.

[2] D. P. Shea and J. E. Mitchell, "A 10-gb/s 1024-way-split 100-km long-reach optical-access network," *JOURNAL OF LIGHTWAVE TECHNOLOGY*, vol. 25, no. 3, pp. 685–693, March 2007.

[3] *10-Gigabit-Capable Passive Optical Networks (GPON) Series of Recommendations*, G.987.x, ITU Std., March 2010. [Online]. Available: http://www.itu.int/rec/T-REC-G/e

[4] "The ns-3 network simulator (available at http://www.nsnam.org/)," 2008.

[5] A. Cauvin, A. Tofanelli, J. Lorentzen, J. Brannan, A. Templin, T. Park, and K. Saito, "Common technical specification of the g-pon system among major worldwide access carriers," *IEEE Communications Magazine*, vol. 44, no. 10, pp. 34–40, October 2006.

[6] *IEEE 802.3ah Task Force*, IEEE Std., June 2004. [Online]. Available: http://www.ieee802.org/3/efm

[7] *Broadband optical access systems based on Passive Optical Networks (PON)*, Rec. G.984.x, ITU Std., 2005.

[8] *Gigabit-Capable Passive Optical Networks (G-PON)*, Rec. G.984.x, ITU Std., October 2008.

[9] B. Skubic, J. Chen, J. Ahmed, L. Wosinska, and B. Mukherjee, "A comparison of dynamic bandwidth allocation for epon, gpon, and next-generation tdm pon," *IEEE Communications Magazine*, vol. 47, no. 3, pp. S40–S48, March 2009.

[10] *IEEE 802.3av 10G-EPON Task Force*, IEEE Std., September 2009.

[11] C. Assi, M. Maier, and A. Shami, "Toward quality of service protection in ethernet passive optical networks: challenges and solutions," *IEEE Network*, vol. 21, no. 5, pp. 12–19, 2007.

[12] H. Song, B.-W. Kim, and B. Mukherjee, "Multi-thread polling: A dynamic bandwidth distribution scheme in long-reach pon," *IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATION*, vol. 27, no. 2, p. 134, Feburary 2009.

[13] M. V. Dolama and A. G. Rahbar, "Multi group dynamic bandwidth allocation scheme in long reach pons," in *International Symposium on Telecommunications*, 2010.

[14] H. C. Leligoun, C. Linardakis, K. Kanonakis, J. D. Angelopoulos, and T. Orphanoudakis, "Efficient medium arbitration of fsan-compliant gpons," *International Journal of Communication Systems*, vol. 19, pp. 603–617, 2006.

[15] M.-S. Han, H. Yoo, B.-Y. Yoon, B. Kim, and J.-S. Koh, "Efficient dynamic bandwidth allocation for fsan-compliant gpon," *Journal of Optical Networking*, vol. 7, no. 8, pp. 783–795, August 2008.

[16] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Transactions on Networking*, vol. 1, no. 4, pp. 397–413, 1993.

[17] J. Postel, "Transmission control protocol - darpa internet program protocol specification," RFC 793, DARPA, Sep. 1981.

[18] R. Stewart, "Stream control transmission protocol," RFC 4960, Sep. 2007.

[19] E. Kohler, M. Handley, and S. Floyd, "Designing dccp: Congestion control without reliability," in *SIGCOMM*, 2006.

[20] J. D. Angelopoulos, H.-C. Leligou, T. Argyriou, S. Zontos, E. Ringoot, and T. V. Caenegem, "Efficient transport of packets with qos in an fsan-aligned gpon," *IEEE Communications Magazine*, vol. 42, no. 2, pp. 92–98, Feburary 2004.

[21] B. Skubic, J. Chen, J. Ahmed, B. Chen, L. Wosinska, and B. Mukherjee, "Dynamic bandwidth allocation for long-reach pon: Overcoming performance degradation," *IEEE Communications Magazine*, vol. 40, no. 11, pp. 100–108, November 2010.

[22] S. Ha, I. Rhee, and L. Xu, "Cubic: a new tcp-friendly high-speed tcp variant," *Operating Systems Review*, vol. 42, pp. 64–74, Jul. 2008.

[23] K. Tan, J. Song, Q. Zhang, and M. Sridharan, “A compound tcp approach for high-speed and long distance networks,” in *INFOCOM*, 2006.

[24] H. Balakrishnan, V. N. Padmanabhan, G. Fairhurst, and M. Sooriyababdara, “Tcp performance implications of network path asymetry,” RFC 3449, 2002.

[25] H. Ikeda and K. Kitayama, “Dynamic bandwidth allocation with adaptive polling cycle for maximized tcp throughput in 10g-epon,” *JOURNAL OF LIGHTWAVE TECHNOLOGY*, vol. 27, no. 23, pp. 5508–5516, December 2009.

[26] E. C. Perkins, “Ip mobility support for ipv4,” RFC 3344, Aug. 2002.

[27] “Opnet modeler.” [Online]. Available: http://www.opnet.com/solutions/network_rd/modeler.html

[28] “Qualnet.” [Online]. Available: http://www.scalable-networks.com/content/products/qualnet

[29] “Ns2 network simulator,” http://www.isi.edu/nsnam/ns/.

[30] Z. Peng and P. Radcliffe, “Modeling and simulation of ethernet passive optical network (epon) experiment platform based on opnet modeler,” in *ICCSN*, 2011.

[31] C.-H. Chang, “Dynamic bandwidth allocation mac protocols for gigabit-capable passive optical networks,” Ph.D. dissertation, University of Hertfordshire, 2008.

[32] S. Jansen and A. McGregor, “Simulation with real world network stacks,” in *Winter Simulation Conference*, 2005.

[33] “Tcptrace,” http://jarok.cs.ohiou.edu/software/tcptrace/.

[34] “Wireshark.” [Online]. Available: http://www.wireshark.org/