Performance Analysis of QoS Aware Distributed Schemes over a Ring- based EPON Architecture

ASM Delowar Hossain, M. Kouar, M. Ummy, M. Razani

Abstract*— In this work, we introduce decentralized Dynamic Bandwidth Allocation (DBA) schemes capable of supporting upstream Quality of Service (QoS) through differentiated class of service (CoS). In contrast to the centralized approach, the proposed QoS aware distributed DBA supports differentiated services through the integration of both scheduling mechanisms (intra-ONU and inter-ONU) at the Optical Network Unit (ONU). This integration of both scheduling can only be supported by a decentralized architecture. We demonstrate, in addition to the added flexibility and reliability, that the distributed approach has characteristics that make it far better suited than its centralized counterpart for provisioning QoS necessary for properly handling voice, video, and data services over a single line.

Index Terms*— Distributed Control Scheme, Quality of Service, Passive Optical Access Network.

I. INTRODUCTION

The access network bottleneck problem between high-capacity local area networks (LANs) and the backbone network causing a serious problem. Passive optical network (PON) is a feasible solution to this bottleneck [1-11]; therefore, PON (specially Ethernet PON) is expected to serve voice, video and data over a single line with a given Quality of Service (QoS) requirements. Each type of traffic has a different quality constraint and requires differentiated Class of Service (CoS).

Number of centralized Dynamic Bandwidth Allocation (DBA) schemes were recently introduced [2-5]. There are inherent drawbacks with centralized architecture, such as lack of global optimization in upstream DBA, inefficiency in bandwidth utilization, etc[11]. Founded upon these centralized schemes, upstream QoS support was introduced in EPON, where the intra-ONU scheduling of traffic classes takes place in ONU and upstream inter-ONU scheduling (DBA) takes place in Optical Line Terminal (OLT) [6-9]. Since the two scheduling schemes are independent of each other, the final bandwidth allocated to a particular class of traffic for a given ONU may not be the optimum choice.

Manuscript received September, 2013.

ASM Delowar Hossain, Department of Electrical Engineering and Telecommunication Engineering Technology, NYCCT, City University of New York, NY, USA.

M. Kouar, Department of Electrical Engineering and Telecommunication Engineering Technology, NYCCT, City University of New York, NY, USA.

M. Ummy, Department of Electrical Engineering and Telecommunication Engineering Technology, NYCCT, City University of New York, NY, USA.

M. Razani, Department of Electrical Engineering and Telecommunication Engineering Technology, NYCCT, City University of New York, NY, USA.

To address the above mentioned limitations of centralized scheme, we introduce decentralized DBA schemes capable of supporting upstream QoS through differentiated CoS. In contrast to the centralized approach, the proposed QoS aware distributed DBA supports differentiated services through the integration of both scheduling mechanisms (intra-ONU and inter-ONU) at the ONU. This integration of both scheduling can only be supported by a decentralized architecture.

With the support of this decentralized scheme, we develop QoS-based algorithm where intra-ONU (priority queuing) and inter-ONU bandwidth allocation takes place in ONUs.

We demonstrate, in addition to the added flexibility and reliability, that the distributed approach has characteristics that make it far better suited than its centralized counterpart for provisioning QoS necessary for properly handling voice, video, and data services over a single line.

II. OVERVIEW OF CENTRALIZED QOS SCHEME

An OLT-based polling scheme, called *Interleaved Polling with Adaptive Cycle Time* (IPACT) based on *Grant* and *Request* messages, has been presented in [3]. ONUs request OLT for upstream bandwidth; OLT being the upstream arbitrator, allocates upstream bandwidth to each ONU according to an algorithm (Fig.1). Using IPACT, several DBA schemes were studied in [3]; namely fixed, limited, gated, constant credit, and linear credit. Amongst these algorithms, the limited was shown to exhibit the best performance. The OLT based DBA (inter-ONU scheduling) was enhanced in [9] to support QoS through intra-ONU scheduling (priority queuing) at the ONUs. Priority queuing with queue management facilitates class level traffic policing to allow traffic into ONU queues as well as class level transmission scheduling as per OLT's inter-ONU bandwidth allocation. Because the centralized limited IPACT scheme was shown to exhibit the best performance in [3], we will consider the QoS scheme detailed in [9] as a reference model for comparing the performance of our proposed distributed QoS scheme. An overview of QoS enabling mechanisms are detailed next.

A. Scheduling at OLT (inter-ONU)

The limited IPACT DBA scheme is cycle-based, where a cycle (T_{CYC}) is defined as the time that elapses between two executions of the scheduling algorithm. A cycle has a variable length size confined within certain lower and upper bounds, which we denote as T_{MIN} and T_{MAX} (sec) respectively. Thus, the algorithm schedules between B_{MIN} and B_{MAX} (bytes) at a time, where B^i is determined by multiplying T^i with the line rate. In this scheme, the ONU will be granted the requested number of bytes, but no more than a given predetermined maximum B_{MAX}. If R^i is the requested bandwidth of ONU^i, then the granted bandwidth ($B^i_{Granted}$) is equal to:

$$B^i_{Granted} = \begin{cases} R^i & if \quad R^i \le B_{MAX} \\ B_{MAX} & if \quad R^i > B_{MAX} \end{cases}$$

B_{MAX} is determined by the maximum cycle time T_{MAX} : $B_{MAX} = \frac{1}{N}[R_{EPON}\ (\ T_{MAX} - (N*T_G)]$, where N is the number of ONUs, T_G is the guard band time between two consecutive ONU transmission slots, and R_{EPON} is EPON line rate. The bandwidth allocation information ($B^i_{Granted}$) is sent to ONUs by the OLT through a GATE message. $B^i_{Granted}$ is used by ONUs for intra-ONU scheduling to arrange class level transmission.

B. Scheduling at ONU (intra-ONU)

Queue management and priority queuing are used to divide an ONU's timeslot (allocated by the OLT) to the different classes of traffic supported by that ONU. It provides low delay to high-priority traffic, but it has some performance shortcomings such as better-than-needed performance for high-priority queues and starvation of low-priority queues. Each ONU is equipped with *n* queues serving *n* priority classes (denoted P_0,P_1, . . . ,P_n), with P_0 being the highest priority and P_n being the lowest. When a packet is received at ONU, the ONU classifies its type and places it in the corresponding queue. The queues in each ONU share common memory space. If an arriving packet with priority *Pi* finds the buffer full in the ONU, it can preempt one or more lower-priority packets P_j *(j > i)* from their queues, such that the P_i packet can itself be placed into the P_i queue. Between transmission slots, an ONU stores all the packets received in their respective queues. When the ONU timeslot starts, the ONU serves a higher-priority queue to exhaustion before serving a lower-priority queue. We assume there are three classes of traffic. If total reported queue size of an ONU, $R_t = (R_{p0} + R_{p1} + R_{p2})$, where R_{p0}, R_{p1}, R_{p2} are queue sizes of P_0, P_1, P_2 classes respectively. The class level bandwidth allocations (B_{p0}, B_{p1}, B_{p2}) within an ONU are as follows:

$$B_{p0} = R_{p0}, B_{p1} = R_{p1}, B_{p2} = R_{p2}, \qquad \text{when} \quad R_t \le B_{MAX}$$

$$B_{p0} = R_{p0}$$

$$B_{p1} = \begin{cases} R_{p1} & if \quad R_{p1} \le (B_{MAX} - B_{p0}) \\ (B_{MAX} - B_{p0}) & if \quad R_{p1} > (B_{MAX} - B_{p0}) \end{cases} \qquad \text{otherwise}$$

$$B_{P2} = \begin{cases} R_2 & if\, R_2 \le (B_{MAX} - B_{P0} - B_{P1}) \\ (B_{MAX} - B_{P0} - B_{p1}) & if\ R_2 > (B_{MAX} - B_{P0} - B_{P1}) \end{cases}$$

Note the above referenced DBA scheme is OLT-based and OLT has the centralized intelligence. The (inter-ONU scheduling) performance of most of the centralized schemes, including the limited IPACT scheme, suffers from several limitations, including: (1) the bandwidth granted by the OLT, during cycle n, to ONU^i is only determined by the content of a single REPORT message transmitted in the previous cycle n-1 by ONU^i (i.e., the bandwidth computation module does not take into account the remaining requests of other ONUs). Thus, the process of bandwidth allocation is not globally optimized; (2) due to the bursty nature of Ethernet traffic, some ONUs might have less traffic to transmit while other ONUs may require more bandwidth than B_{MAX}. For instance, assume that ONU^i requests an amount of bandwidth $R^i < B_{MAX}$, while ONU^j requests an amount of bandwidth $R^j > B_{MAX}$. Although there is an excess amount of bandwidth (B_{MAX} - R^i) that can be granted to ONU^j, however, due to limitation # 1 cited above, the maximum bandwidth that may be granted to ONU^j is only B_{MAX}; (3) since the centralized scheme requires typical guard band between two consecutive ONU transmissions, it reduces available upstream bandwidth. These lead to overall inefficient utilization of upstream bandwidth as well as inefficiency in intra-ONU scheduling; (4) furthermore, the intra-ONU scheduling takes place in ONU and upstream inter-ONU scheduling (DBA) takes place in OLT; since the two scheduling schemes are independent of each other, the final bandwidth allocated to a particular class of traffic for a given ONU may not be the optimum choice.

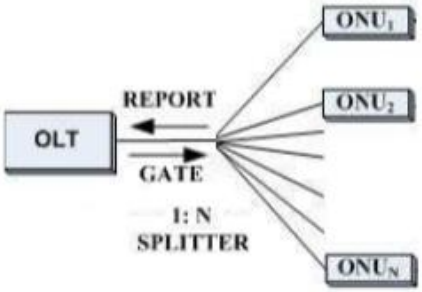

Figure 1: Centralized PON tree configuration

III. PROPOSED QOS SCHEME OVER A DISTRIBUTED RING ARCHITECTURE

The proposed QoS schemes, reliant upon a distributed architecture (distributed control plane), are introduced to address the impediments of centralized schemes. Therefore, it is imperative to understand how the decentralized scheme works. We introduce a short overview of the general principles of decentralized operation[11].

An OLT is connected to N number of ONUs via a 10-20 km trunk feeder fiber, a passive 3 port optical circulator, and a short distribution fiber ring. The set of ONUs are joined by point-to-point links in a closed loop. The links are unidirectional: both downstream and upstream signals (combined signal) are transmitted in one direction only. Fig. 2b shows detailed ONU architecture. Each ONU attaches to the ring at a (n: 1-n) 1x2 passive star coupler (incoming signal at point A in Fig. 2b) and can transmit data onto the ring through the output port of a 2x1 *Coarse Wave Division Multiplexing* (CWDM) combiner (outgoing signal at point C in Fig. 2b). Note that in addition to the conventional transceiver receiver tuned at λ_{up}.

Downstream signal is coupled to the ring at port 2 of the optical circulator. After recombining with the re-circulated upstream signal via another 2x1 CWDM combiner (Fig. 2a) placed on the ring directly after the optical circulator, the combined signal then circulates around the ring (ONU 1 through ONU N) in a Drop-and-Go fashion. The downstream signal is then removed at the end of the ring using a filter (located directly after the last ONU) that passes only the 1310 nm upstream signal. The upstream signal emerging from the filter at the end of the ring is split into two components via a 1x2 passive splitter (Fig. 2a) placed on the ring directly after the filter. The first component is directed towards the OLT via circulator ports 1 and 3, while the second component is allowed to re-circulate around the ring after recombining with

the downstream signal (originating from the OLT) via the 2x1 CWDM combiner Fig. 2a.

The (n: 1-n) 1x2 coupler (n is a small arbitrary percentage assumed here to be 10%) splits the incoming combined signal at each node into a small (10%) ―Drop-signal-portion‖ and a large (90%) ―Go-signal-portion‖ . The small portion of the circulating combined signal dropped at each node (Drop-signal) is passed through a filter that removes the upstream signal and passes only the downstream broadcast signal, which is then received and processed by the 1490 nm downstream receiver. The remaining portion of the combined signal emerging from the 90% coupler's port (Go-signal) is first separated into its two constituent: downstream and upstream signals via a CWDM filter. The separated upstream signal (second component) is received and processed via the 1310 nm upstream optical receiver housed at the ONU, where it is then regenerated and retransmitted along with the ONU's own local control and data traffic.

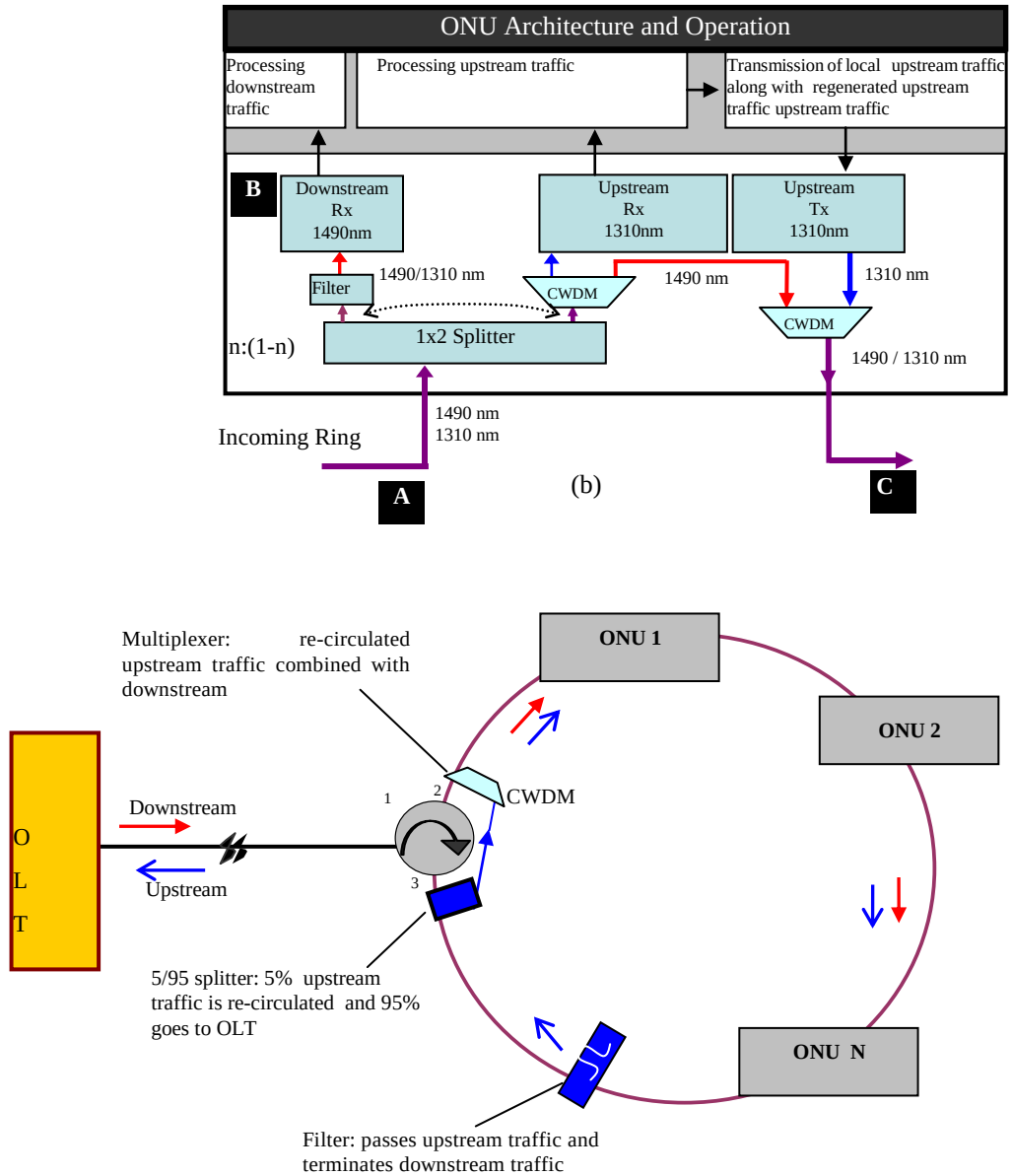

Figure 2: (a) Distributed ring-based architecture (b) ONU architecture

Finally, the separated downstream signal is re-combined again with the retransmitted upstream signal (regenerated plus local) via the 2x1 CWDM of Fig. 2b to form the outgoing combined signal (incoming combined signal for next ONU) that circulates around the ring.

Since upstream transmission is based on a TDMA scheme, inter-ONU traffic (LAN data and control messages exchanged among ONUs) is transmitted along with upstream traffic destined to the OLT (MAN/WAN data) within the same pre-assigned time slot. The first component of the upstream signal destined to the OLT is received and processed by the 1310 nm upstream optical receiver (housed at the OLT), which accepts only MAN/WAN traffic, discards LAN traffic, and may discard or process (for reasons to be given below) the control messages. On the other hand, the second component of upstream signal is transmitted sequentially around the ring from one node to the next where it is regenerated and retransmitted at each node.

Since the ring is a closed loop, upstream traffic will circulate indefinitely unless removed. The process of removing, regenerating and retransmitting the second component of the upstream signal at each node (ONU) is implemented as follows: first, the 1310 nm upstream optical receiver (housed at each ONU) terminates all upstream traffic, examines the destination MAC address of each detected Ethernet frame, and then performs one or more of the following functions: (1) all re-circulated upstream traffic addressed to the OLT is removed by the first ONU (ONU that is physically located on the ring directly after the 2x1 CWDM coupler of Fig. 2a); (2) all control messages (REPORTs) must be processed, regenerated, and then retransmitted by each node; (3) the source node removes its own transmitted inter-ONU control messages that complete one trip around the ring through re-circulation; (4) transient LAN traffic, terminated at an intermediate node, but destined to other nodes are regenerated and then retransmitted along with the node's own local upstream traffic within the designated proper time slot; (5) once the destination address of the LAN traffic matches the node's MAC address, it is copied and delivered to the end users and then discarded (not retransmitted to the next ONU).

A. Integrated Scheduling at ONU

Priority Queuing: it is a simple method for supporting differentiated service classes as discussed in section II.B.

Transmission Scheduling: Based on bandwidth demands, ONUs can be classified into two groups, namely: lightly loaded ONUs that have bandwidth demands less than B_{MAX}; and heavily loaded ONUs that have bandwidth demands more than B_{MAX}. Note each ONU is allowed up to B_{MAX} without any arbitration scheme.

During each cycle, the DBA module must now keep track of the unclaimed bandwidth from the set of lightly loaded ONUs. It then must redistribute (in addition to B_{MAX}) this excess bandwidth to other heavily loaded ONUs based on certain scheme[7].

During each cycle, the lightly loaded ONUs with $R_t^i < B_{MAX}$ will contribute a total cycle bandwidth: $B_{Cycle\_Remainder} = \sum_{i}^{L} (B_{MAX} - R_t^i)$, L: is the number of lightly loaded ONUs. The heavily loaded ONUs with $R_t^i > B_{MAX}$ will require a total over the limit cycle bandwidth: $B_{Cycle\_OverLimit} = \sum_{i}^{H} (R_t^i - B_{MAX})$, H: is the number of heavily loaded ONUs. An ONU can transmit as much as reported queue size when any of (a) or (b) is true:

a) $R_t \le B_{MAX}$, note ONU can transmit without waiting for DBA calculation, as per reporting sequence [11].

b) $R_t > B_{MAX}$ & $B_{Cycle\_Remainder} \ge B_{Cycle\_OverLimit}$

It implies that an ONU will be allowed bandwidth (B_{p0}, B_{p1}, B_{p2}) to transmit all traffic from each class as reported, as shown below $B_{p0} = R_{p0}, B_{p1} = R_{p1}, B_{p2} = R_{p2}$.

On the contrary, when none of the above holds, then it requires the ONUs to invoke detail algorithm to distribute cycle bandwidth among the ONU classes considering fairness and QoS restrictions. Six variations of these algorithms are introduced in this section. We will call the decentralized DBAs (DDBA) as DDBA1 through DDBA6.

Note the transmission of various classes of traffic within an ONU can be scheduled in any of the two sequences:

Option 1: ONU_n transmits all three classes of traffic that fits within its allocated bandwidth, then ONU_{n+1} repeats the process until all ONUs complete their transmissions in that cycle. Note, with in an ONU transmission timeslot, the high priority class is always transmitted first.

Option2: Unlike previous scheme, ONU_n transmits its high priority P_0 traffic only, followed by the P_0 traffic of ONU_{n+1} and this process continues until all the ONUs complete their transmission of P_0 traffic in that cycle. Then ONU_n starts to transmit its P_1 traffic followed by ONU_{n+1} P_1 traffic, until all ONUs complete P_1 traffic transmission in that cycle. Then this process continues for P_2 traffic as well. The cycle ends when all classes of traffic of all ONUs are transmitted.

Note the ONU queue report transmission can be accomplished in two ways (a) an ONU can send its report within its time slot as in [11], saving the DBA time but report will be untimely (b) ONUs can send report at the end of each cycle, costing DBA time but benefiting from up-to-date report.

B. Distributed DBAs

1. DDBA1: In this scheme, the queue reporting is scheduled within the ONU time slot. Generally, the P_0 class demand is low ($B_{max} >> P_0$) and it results in granted bandwidth equal to the report, $B_{p0} = R_{p0}$. Rest of the bandwidth of B_{max} and cycle remainder is allocated to P1 first and any left over is allocated to P2. First, the assessment of P_1 demands of all ONUs that exceed (B_{max}-B_{p0}) is calculated as:

$$B_{p2}^i = \begin{cases} 0, & if\ (B_{\max} - R_{po}^i - R_{p1}^i) \le 0\ \&\ \bar{B}_{Cycle\_Remainder} \le 0 \\ \min(R_{p2}^i, (B_{\max} - R_{po}^i - R_{p1}^i)), & if\ (B_{\max} - R_{po}^i - R_{p1}^i) > 0\ \&\ \bar{B}_{Cycle\_Remainder} \le 0 \\ (B_{\max} - R_{po}^i - R_{p1}^i + B_{p2\_extra}^i), & if\ (B_{\max} - R_{po}^i - R_{p1}^i) > 0\ \&\ R_{p2}^i > (B_{\max} - R_{po}^i - R_{p1}^i)\ \&\ \bar{B}_{Cycle\_Remainder} > 0 \\ \min(R_{p2}^i, B_{p2\_extra}^i), & if\ (B_{\max} - R_{po}^i - R_{p1}^i) \le 0\ \&\ R_{p2}^i > 0\ \&\ \bar{B}_{Cycle\_Remainder} > 0 \end{cases}$$

$B_{P1\_Cycle\_OverLimit} = \sum_{i \in H1} Y^i$, where $Y^i = (R^i_{p0} + R^i_{p1}) - B_{MAX}$ and H1 is the number of heavily loaded ONUs with $(R^i_{p0} + R^i_{p1}) > B_{MAX}$. Then P1 class is allocated bandwidth as follows:

$$B^i_{p1} = \begin{cases} R^i_{p1} & if\ R^i_{p1} \le (B_{\max} - B^i_{p0}) \\ R^i_{p1} & if\ B_{P1\_Cycle\_OverLimit} \le B_{Cycle\_Remainder} \\ (B_{\max} - B^i_{p0}) + B^i_{p1\_extra} & if\ R^i_{p1} > (B_{\max} - B^i_{p0})\ \& \\ & B_{P1\_Cycle\_OverLimit} > B_{Cycle\_Remainder} \end{cases}$$

and $B^i_{p1\_extra}$ = Max-Min Fair ($B_{Cycle\_Remainder}$, Y^i), where $\forall i \in H1$.

A short description of Max-Min Fair algorithm is as follows**.** *Max-Min Fair [12–16] is a* resource distribution scheme. The principle of this scheme is as follows: intuitively, a fair share allocates a queue with a "small" demand that it wants, and evenly distributes unused resources to the "high-demand" queues. While sharing C bandwidth among n queues, where $\sum_{i=1}^{n} Q^i > C$, the order of calculation is as follows: 1) resources are allocated ***in order of increasing demand*** ($Q^1 \le Q^2 \le \ldots \le Q^n$) . 2) no queue gets a resource share larger than its demand $B^i_{Share} = \min(Q^i, C/n)$ 3) no other allocations satisfying (2) has a higher minimum allocation 4) condition (3) ***recursively holds*** as we remove the minimal user and reduce the total resource accordingly. 5) queues with unsatisfied demands get ***an equal share of the resources*** ***(*** $B^i_{Share} = B^{i+1}_{Share}$***)***. Note that, $C = \sum_{i=1}^{n} B^i_{Share}$

Then the P_2 demands of all ONUs which exceeds (B_{max}-R_{p0}-R_{p1}) is derived: $B_{P2\_Cycle\_OverLimit} = \sum_{i \in H2} X^i$ where H2: # of heavily loaded ONUs and X is defined as

$$X^i = \begin{cases} R^i_{p2} & if\ (B_{\max} - R^i_{po} - R^i_{p1}) \le 0 \\ R^i_{p2} - (B_{\max} - R^i_{po} - R^i_{p1}) & if\ (B_{\max} - R^i_{po} - R^i_{p1}) > 0 \end{cases}$$

Then the present cycle remainder is derived as follows:

$$\bar{B}_{Cycle\_Remainder} = B_{Cycle\_Remainder} - \sum_{i \in H1} Y^i$$

This leads to the P2 class allocations as follows:

where $B^i_{p2\_extra} = \bar{B}_{Cycle\_Remainder} \left[\frac{X}{B_{P2\_Cycle\_OverLimit}} \right]$

Note that transmissions of traffic classes are scheduled as per transmission option 1 (as discussed in this section earlier) .

2. DDBA2: It is the same as DDBA1, except the transmission is not scheduled based on ONU, but based on classes (transmission option 2).

3. DDBA3: Note that reporting is scheduled within ONU time slot. Since the P0 demand is small, $B^i_{p0} = R^i_{p0}$ and then the rest of Bmax is divided between P1 and P2. Then the cycle remainder is distributed among P1 of all the ONUs through Max-Min Fair allocation.

Finally, the cycle remainder is distributed among P2 of all the ONUs through proportional distribution. The overall DDBA3 process is as follows:

$B^i_{p0} = R^i_{p0}$

$$B^i_{p1} = \begin{cases} R^i_{p1} & if\ R^i_{p1} \le \frac{(B_{\max} - R^i_{p0})}{2} \\ \frac{(B_{\max} - R^i_{p0})}{2} + B^i_{p1\_extra} & if\ R^i_{p1} > \frac{(B_{\max} - R^i_{p0})}{2} \end{cases}$$

$B^i_{p1\_extra}$ = Max-Min Fair ($B_{Cycle\_Remainder}$, Y^i), $\forall i \in H1$ where H1: # of heavily loaded ONUs

and $Y^i = R^i_{p1} - \frac{(B_{\max} - R^i_{p0})}{2}$, Y>0. Now the P2 allocation is

$$B^i_{p2} = \begin{cases} R^i_{p2} & if\ R^i_{p2} \le \frac{(B_{\max} - R^i_{p0})}{2} \\ (B_{\max} - R^i_{p0})/2 + B^i_{p2\_extra} & if\ R^i_{p2} > \frac{(B_{\max} - R^i_{p0})}{2} \end{cases}$$

where $B^i_{p2\_extra} = \bar{B}_{Cycle\_Remainder} \left[\frac{X^i}{B_{P2\_Cycle\_OverLimit}} \right]$ and accumulative over demand of P2 is

$B_{P2\_Cycle\_OverLimit} = \sum_{i \in H2} X^i$, H2: # of heavily loaded ONUs, where $X^i = R^i_{p2} - \frac{(B_{\max} - R^i_{p0})}{2}$, X>0 and $\bar{B}_{Cycle\_Remainder} = B_{Cycle\_Remainder} - \sum_{i \in H1} Y^i$

Note that all transmissions scheduled as per transmission option 2.

4. DDBA4: It is the same as DDBA2, but the difference is in ONU reporting scheme. Unlike reporting at the beginning of slot of each ONU, reporting of all ONUs takes place at the end of cycle. It costs DBA idle time, but facilitates up-to-date reporting.

5. DDBA5: It is the same as DDBA3, but the difference is in ONU reporting scheme. Unlike reporting at the beginning of slot of each ONU, reporting of all ONUs takes place at the end of cycle as in DDBA4.

6. DDBA6: Reporting of all ONUs take place at the end of cycle. The needy ONUs are proportionally divided the cycle remainder bandwidth (in addition to Bmax). Then P_0,P_1,P_2 classes share the allocated bandwidth to the ONU utilizing the Max-Min Fair scheme. Note this is the only DDBA where P1 class has no preference over P2.

$B_{Cycle\_Remainder} = \sum_{i}^{L} (B_{MAX} - R^i_t)$, L: is the number of lightly loaded ONUs and where $R^i_t = R^i_{p0} + R^i_{p1} + R^i_{p2}$.

The heavily loaded ONUs with $R_t^i > B_{MAX}$ will require a total over the limit cycle bandwidth:

$B_{Cycle\_OverLimit} = \sum_{i}^{H} (R_t^i - B_{MAX})$, H: is the number of heavily loaded ONUs. Each needy ONU's proportional share of the cycle remainder is: $B_{extra}^i = B_{Cycle\_Remainder} \left[\frac{R_t^i - B_{MAX}}{B_{Cycle\_OverLimit}} \right]$ and $B_{Granted}^i = B_{extra}^i + B_{MAX}$.

$B_{Granted}^i$ is shared among the queues (R_k^i) in ONU^i (intra-ONU allocation) using Max-Min Fair scheme as follows: B_k^i = Max-Min Fair ($B_{Granted}^i$, R_k^i), where $\forall k \in P$ and P=$\{P_0, P_1, P_2\}$.

Note that, in contrast to centralized IPACT where the order of ONUs transmission is fixed (i.e., sequential) in each cycle, the distributed schemes has the added flexibility of varying the order of ONUs transmission according to ONUs traffic demands and priority.

IV. PERFORMANCE EVALUTION

In this section, we compare the simulation performance of the proposed QoS aware decentralized schemes with that of the centralized one. Two simulation programs with identical network parameters were developed, one for the QoS aware centralized IPACT scheme and the other for the decentralized QoS scheme. The performance metrics used here are average packet queuing delay, average queue size and packet loss ratio.

To compare the performance results of the proposed distributed scheme with that of the centralized scheme, we used identical network parameters: a system with 16 ONUs, access link data rate from users to an ONU of 100 Mbps, and a 1 Gbps upstream link data rate (from an ONU to the OLT). The distance between the OLT and the ONUs is ~21 km for the centralized tree architecture and 20km to 23km (ring circumference 3km) for decentralized architecture. Maximum cycle time is 2 ms. The guard time for centralized scheme, separating two consecutive ONU transmissions, is 5 µs. There is no guard time for the decentralized architecture. Buffer size in each ONU is 10 MB.

The traffic model used here is the same as that reported in [11] where each ONU has a number of ON/OFF sources, each with a Pareto distribution governing the lengths of the ON/OFF periods, in order to capture the self-similar nature of Ethernet traffic [17-18]. Note we generated uneven loads from the ONUs, where half of the ONUs are heavily loaded and other half are lightly loaded.

We consider three priority classes P0, P1, and P2. Here P0 is the highest priority and P2 is the lowest. These classes are used for delivering voice, video stream, and data. Each ONU maintains three separate priority queues that share the same buffering space: i) Class P0 is used to emulate a circuit over packet connection. P0 traffic has CBR. In our model, we chose to emulate a Tl connection. The Tl data arriving from the user is packetized at the ONU by placing 24B of data in a packet. Including Ethernet and UDP/IP headers, it results in a 70 bytes frame generation every 125us. Hence, the P0 data consumed 4.48 Mbps of bandwidth [9]. This is the highest priority traffic. ii) Class P1 consisted of VBR video streams that exhibit properties of self-similarity. Packet sizes in P1 streams is standard Ethernet frame ranged from 64 to 1518 B. iii) Class P2 is same as P1, but not time sensitive. This class has the lowest priority. As we varied the ONU offered load, P0 was always kept constant [4.48Mbps/100 Mbps=0.0448 of an ONU offered load (OOL)]. The remaining load was split equally between P1 and P2. For n ONUs, the total network load (TNL) = $\sum_{i}^{n} OOL_i$.

Figure 3 shows the queuing delays of the highest priority class (P0). Note the P0 traffic demand is very low, therefore under any scheme, this class always receives bandwidth equal to its report. In most cases all variations of DDBAs outperform centralized scheme, because:

(i) DDBA has more available bandwidth (no inter-ONU guard time).
(ii) Their DBA decisions are globally optimized due overall network demand analysis.
(iii) Redistribution of cycle remainder.
(iv) Since the ONUs decide their inter-ONU and intra-ONU bandwidth, their class level allocations are more efficient.

The exception to that is at network saturation (TNL ~1), when DDBA1-3 show slightly more delay than the centralized one.

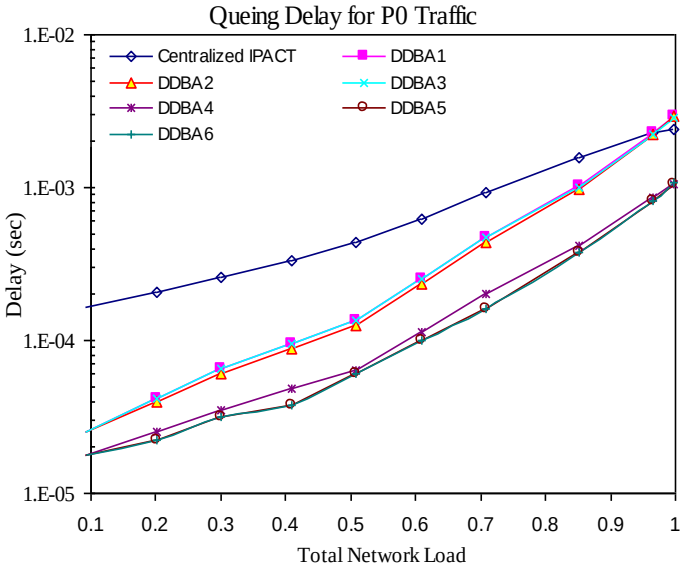

Figure 3: Queuing delays for P0 traffic vs. TNL

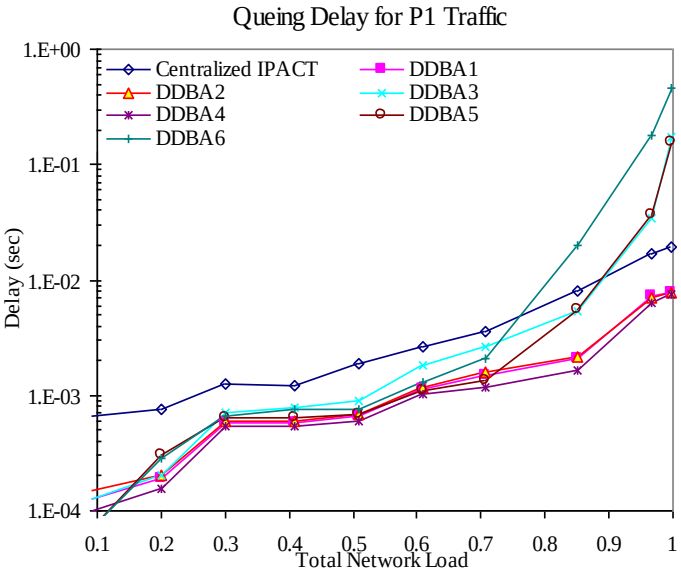

Figure 4: Queuing delays for P1 traffic vs. TNL

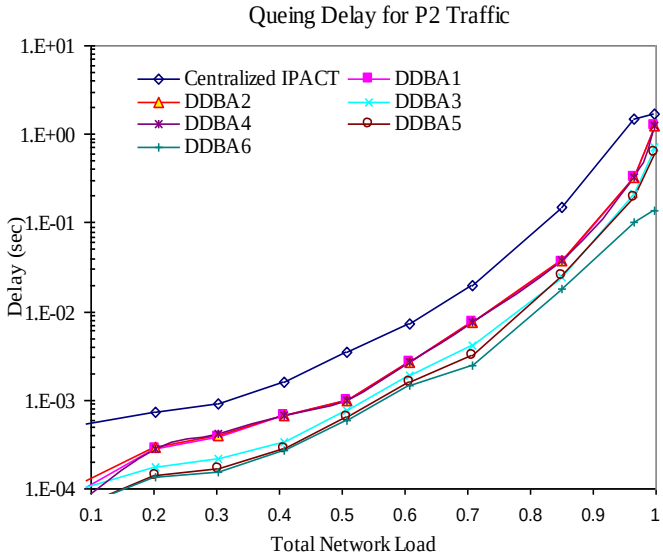

Figure 5: Queuing delays for P2 traffic vs. TNL

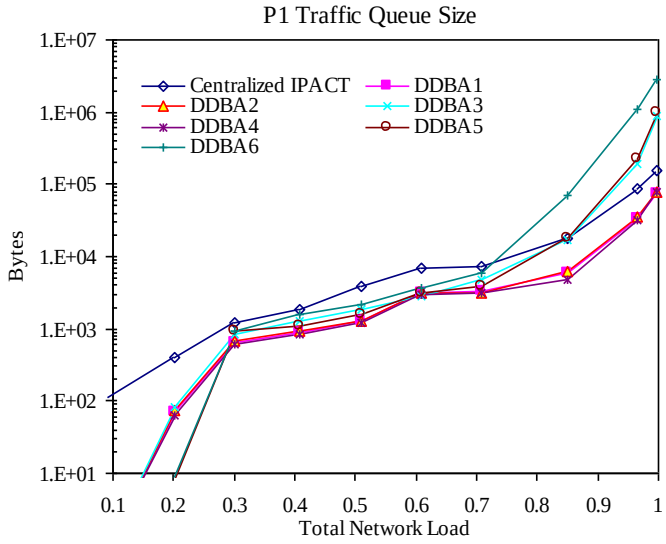

Figure 6: Queue size of P1 traffic vs. TNL

It is because, the DDBA1-3 queue reports are sent at the start of the ONU slot, not as timely as the end of cycle reporting like DDBA4-6. At network saturation the cycle length gets longer and the reports do not reflect the present queue status of the ONUs . Therefore, P0 allocation is not timely enough to outperform centralized scheme. Note the centralized scheme used strict priority, where an ONU first serves the P0 class regardless of the reported queue size to the OLT. It shows better performance for P0, but does not establish fairness among the queues. On the other hand, we are using Fair Queuing scheme [9]; once DBA allocates class level bandwidth, newly arrived P0 traffic are not considered at transmission time. They have to wait until next cycle, causing further delay until next cycle. It slightly lags the P0 performance at network saturation, but establishes fairness among queues. To keep the fairness as well as enhance the P0 performance, we introduced DDBA4, 5, and 6, where ONUs exchange reports at the end of cycle. Thus the reports are timely and allows efficient transmission of traffic, resulting in lower delay of P0 traffic than DDBA1, 2, 3 and IPACT. Note it will cost some idle upstream time, but the overall performance improves.

Figure 4 shows the P1 traffic delays for centralized scheme and various decentralized schemes. Generally, all DDBAs outperforms centralized scheme due to the reasons stated previously. The exception to that is DDBA3, 5 and 6; at TNL~0.75, DDBA6 has higher delay than the centralized scheme. DDBA3, 5 also has similar effect at TNL~0.9. At low TNL, all traffic classes are allocated bandwidth equal to their report and the bandwidth sharing preferences have little impact. But at higher TNL, when the traffic demands exceeds available bandwidth, the sharing preferences among traffic classes impact the delays of each traffic class. As we see in higher TNL, since DDBA6 allocates comparatively lesser bandwidth to P1 (than other DBAs), it results in higher delay

of P1 traffic. DDBA3 and 5 allocate more bandwidth to P1 compare to DDBA6 but less than DDBA1, 2 and 4. Consequently, in higher TNL, DDBA3 and 5 perform better than DDBA6, but worse than DDBA1, 2 and 4. Note, normally P1 traffic gets only 50% of (B_{max}-P0) in DDBA3, and 5, contrary to other DBAs (DBA1, 2, 4) where P1 gets the most bandwidth out of B_{MAX}.

The P1 performance in DDBA6 is worst than all DDBAs; but at low TNL, the DDBA6 performance improves, because of the up-to-date reporting at the end of cycle. It also applies to DDBA5. Despite the DDBA3 and 5 used same bandwidth sharing scheme, the DDBA5 outperforms DDBA3 until TNL~0.85 due to the up-to-date reporting at the end of cycle. After that they both perform the same (and delay exceeds the centralized scheme). Because, at higher load, regardless of freshness of report, the queues are mostly full and exceed B_{MAX}; the timely report cannot help P1 traffic any further. Among the better performing DDBAs (1, 2, 4), all of them gives preference to P1 traffic without any consideration to P2. DDBA4 has the best performance among them, because it uses fresh reporting. Note that this advantage will cause P2 higher delays. For the opposite reason, DDA6 will cause P2 to have lower delays.

Figure 5 shows the P2 traffic queuing delays for all DBAs. Generally, any scheme which gave preference to P1 traffic over P2 has to pay the price with higher delay of P2 traffic. All DDBAs outperform centralized scheme, because it allocates bandwidth to P1 first and then any left over bandwidth is allocated to P2. In addition to that, the other contributing factors are untimely reporting, lack of global optimization, no reuse of cycle remainder and independent intra-ONU and inter-ONU scheduling.

Unlike DDBA3, 5 and 6, the DDBA1, 2, and 4 gave less preference to P2 traffic resulting in a comparatively higher delay of P2 traffic. Among DDBA3, 5, 6, DDBA6 perform slightly better. Because, DDBA3 and DDBA5 equally divide bandwidth to P1 and P2, but gives cycle leftover only to P1; but DDBA6, without giving any preference to P1, always shares all bandwidth among P0, P1 and P2 using Max-Min Fair Distribution scheme. That gives P2 (in DDBA6) a slight advantage. DDBA5 performs slightly better than DDBA3, because of end of cycle fresher report. Note that we are using priority queuing in a fixed buffer size. At higher load, it causes P2 class traffic to be dropped (Fig. 8) resulting in lower delays than expected. For an infinite buffer case the delay would be higher.

Figure 6 shows the P1 traffic queue size. The queue size is a direct reflection of how long the packets stay in the buffer. In another word, longer the queuing delay, larger is the queue size. Figure 4 clearly relates to Figure 6. The centralized scheme along with DDBA1, 2 and 4 first allocate bandwidth to P1 first. After the P1 allocation, P2 may get bandwidth when available. It clearly gives an advantage to P1 and causes smaller P1 queue size. DDBA1, 2 and 4 have smaller queue size than centralized scheme due to inherent advantages of decentralized scheme, specially the redistribution of cycle remainder in the presence of heavily loaded and lightly loaded ONUs.

DDBA3, 5 and 6 do not give total preference to P1 over P2 as in rest of the DBAs. Therefore, they are going have larger queue size than rest of them. But the bandwidth sharing scheme may not have much impact at low load, because of ample availability of bandwidth. Also due to decentralized advantages, DDBA 3, 5, and 6 have smaller queues size than centralized scheme at low TNL. As the TNL grows, the decentralized advantages fade out at the face of massive P1 demand. DDBA3 and 5 allocate 50% of (B_{MAX} –P0) to P1 and other 50% to P2 and any cycle remainder is also allocated to P1. Due to that, under high demand, DDBA3 and 5 queue sizes grow more than DDBA1, 2, 4 and the centralized scheme. DDBA6 is even worse for P1 but fairer than all; it gives no preference between P1 and P2, causing the largest P1 queue size among all DBAs (in high load).

Figure 7 shows the queue size of P2 traffic. Figure 7 relates to Figure 5, due to the proportional relation of queuing delay and queue size. The centralized scheme has the largest queue size because of no preference to P2 traffic. P2 traffic waits for cycles before it is transmitted (especially at higher load). Note, at higher load, it causes P2 class traffic to be dropped and resulting in smaller than expected buffer size. For an infinite buffer case the queue size would be higher. DDBA1, 2 and 4 gives no preference to P2 as the centralized scheme, but due to inherent advantages of the decentralized scheme, they have lesser queue size than centralized scheme. Since DDBA3 and 5 give more preference to P2 than DDBA1, 2 and 4, their queue sizes are smaller than DDBA1, 2 and 4. Between DDBA3 and 5, the DDBA5 performs slightly better due to fresher reports. DDBA6 has the smallest queue size,

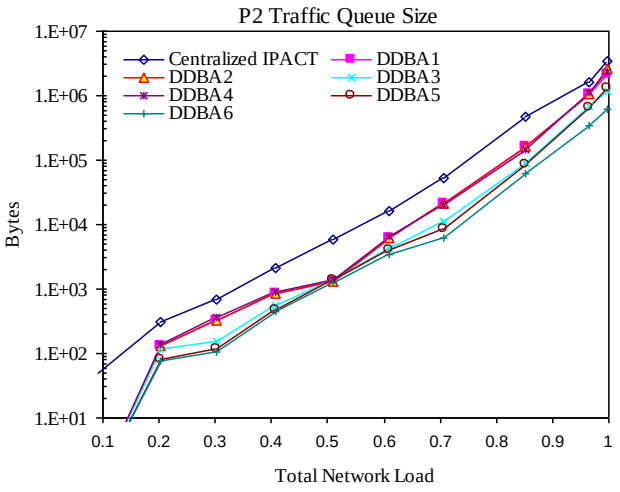

Figure 7: Queue size of P2 traffic vs. TNL

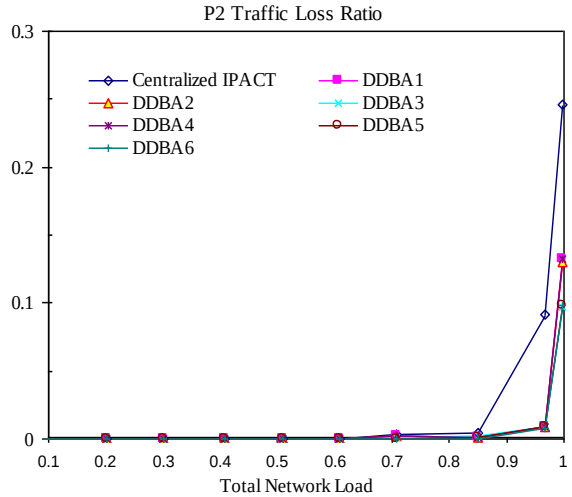

Figure 8: Packet loss ratio of P2 traffic vs. TNL

because it is fairer to P2 than any other DBAs. Note that the accumulative average queue size of P1 and P2 do not reach the maximum size due to the two types of ONU loads (lightly loaded and heavily loaded ONUs). At TNL~1, the queues of the heavily loaded ONUs are saturated and start dropping traffic; on the other hand, the queues of the lightly loaded ONUs are not full, bringing the average queue size down. For the evenly loaded ONUs, at TNL~1, all ONUs fills up almost evenly and starts to drop traffic around the same time, increasing the average queue size to maximum.

Figure 8 shows packet loss ratio of P2 traffic for all DBAs. Note due to priority queuing, the lower priority traffic is dropped to make space for higher priority traffic. Centralized P2 queue size is the largest among all (see Fig. 7). At higher load, the arrival of higher priority traffic (P0 and P1) displaces the P2 traffic. It results in highest P2 packet loss ratio. The decentralized schemes (P2) perform better, because they have shorter P2 queuing delay resulting in smaller P2 queue sizes than centralized scheme. Among the DDBAs, the DDBA 3, 5 and 6 perform slightly better, due to their fairer bandwidth allocation towards P2 traffic.

V. CONCLUSION

We presented QoS aware distributed DBAs supported by decentralized ring architecture. They facilated collision-free upstream data transmission without resorting to the typical use of guard time. Furthermore, in contrast to the centralized approach, the proposed QoS-based distributed DBA supported differentiated services through the integration of both scheduling mechanisms (intra-ONU and inter-ONU) at the ONUs. This integrated scheduling feature that can only be supported by a decentralized architecture, provides better QoS guarantees for properly handling voice, video, and data services over a single line. The higher available upstream bandwidth, redistribution of unused cycle bandwidth, and integrated scheduling resulted in minimized queuing delay, buffer size and packet loss ratio. Among the variations of distributed DBAs, some have a specific advantage over the others. Rather than adapting to only one solution, we demonstrated that according to system needs one solution could be chosen over the others to handle specific circumstances.

REFERENCES

[1] M. Ngo, et al, "Controlling QoS in EPON-based FTTX access networks ", Telecommunication Systems, Volume 48, Issue 1-2, pp 203-217, October 2011.

[2] M. Ma, Y. Zhu, and T. H. Cheng, “A bandwidth guaranteed polling MAC protocol for Ethernet passive optical networks,” in Proc. IEEE INFOCOM’ 03, San Francisco, CA, Mar.–Apr. 2003, pp. 22–31.

[3] G. Kramer, B. Mukherjee, and G. Pesavento, “IPACT: a dynamic protocol for an Ethernet PON (EPON),” IEEE Commun. Mag., pp. 74–80, Feb. 2002.

[4] G. Kramer et al., “Ethernet PON: design and analysis of an optical access network,” Photon. Network Commun. J., vol. 3, no. 3, July 2001.

[5] Monika Gupta et.al**,** “Performance Analysis of FTTH at 10 Gbit/s by GEPON Architecture,” IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 5, September 2010, pp. 265-271.

[6] M. Radivojevic, et al, "Implementation of Intra-ONU Scheduling for Quality of Service Support in Ethernet Passive Optical Networks", IEEE Journal of Lightwave Technology, vol. 27, no. 18, Sept 15, 2009.

[7] Assi et al., “Dynamic bandwidth allocation for quality of service over Ethernet PONs,” IEEE J. Select. Areas Commun., Dec. 03.

[8] Ahmad R. Dhaini, “Per-Stream QoS and Admission Control in Ethernet Passive Optical Networks (EPONs),” IEEE Journal of Lightwave Technology, Vol. 25, No. 7, July 2007, pp.1659-1669.

[9] G. Kramer et al., “On supporting differentiated classes of service in EPON-based access network,” J. Opt. Networks, 2002.

[10] ASM Delowar Hossain, et. al, "Downstream Bandwidth Allocation Scheme in Local Access over a Distributed Control Plane", in Proceedings of IEEE ICCIT 2011, Jordan, March 29-31, 2011

[11] A. Hossain, R. Dorsinville, M. Ali, A. Shami, and C. Assi, "Ring-based local access PON architecture for supporting private networking capability," J. Opt. Network. 5, 26-39, 06.

[12] J.M. Jaffe “Bottleneck Flow Control.”, IEEE Transactions on Communications, 29(7), 1981.

[13] Hou, H. Tzeng and S. Panwar, “A generalized max-min rate allocation policy and its distributed implementation using the ABR flow control mechanism,” IEEE INFOCOM ’98, pp. 1366-1375, Apr. 1998.

[14] A. Malla, M. El-Kadi, S. Olariu and P. Todorova, “A fair resource allocation protocol for multimedia wireless networks,” IEEE Trans. Parallel and Distributed Systems, vol. 14, no. 1, pp. 63-71, Jan. 2003.

[15] Y. Zhou and H. Sethu, “On achieving fairness in the joint allocation of processing and bandwidth resources: principles and algorithms,” IEEE/ACM Trans. Networking, vol. 13, no. 5, pp. 1054 - 1067, Oct. 2005.

[16] M. Hosaagrahara and H. Sethu, “Max-min fairness in input queued switches,” ACM SIGCOMM Student Poster Session, August 2005, Philadelphia, PA, USA.

[17] V. Paxson and S. Floyd, “Wide area traffic: the failure of Poisson modeling,” IEEE/ACM Trans. Networking, vol. 3, pp. 226–244, June 1995.

[18] W. Willinger, M. S. Taqqu, and A. Erramilli, “A bibliographical guide to self-similar traffic and performance modeling for modern high-speed networks,” in Stochastic Networks. Oxford, U.K.: Oxford Univ., 1996, pp. 339–366.