A Novel Dynamic Bandwidth Allocation Algorithm Based on Half Cycling for EPON

Özgür Can TURNA [1], Muhammed Ali AYDIN [1], Tülin ATMACA [2], A. Halim ZAİM [1]
{ ozcantur, aydinali, ahzaim@istanbul.edu.tr} [1]
Department of Computer Engineering, Istanbul University, 34320 Istanbul, Turkey
{tulin.atmaca@it-sudparis.eu} [2]
Department RST, Institut Telecom/ Telecom SudParis, 91011 Evry, France

***Abstract*— The access network solutions based on the fiber infrastructure are examined and developed in the last decade. Ethernet Passive Optical Network (EPON), which has only passive optical units in its' infrastructure, comes front in cost and deliverability of service with high bandwidth and long haul access. In upstream direction, EPON needs a multiple access control mechanism to control the bandwidth allocation among Optical Network Units (ONUs) where Multi-Point Control Protocol (MPCP) is responsible for. In this article we propose a novel dynamic bandwidth allocation algorithm which can increase the link utilization with a fair distribution among ONUs. Our algorithm uses half cycle stops thereby we don't have to wait for calculation while waiting report messages from the entire ONUs. Finally, we simulate an EPON network with mono-service and multi-service traffic in two cases to compare our algorithm with Interleaved Polling with Adaptive Cycle Timing (IPACT) and offline Dynamic Bandwidth Allocation (oDBA) algorithms. Our algorithm gives better performance in byte loss ratio and mean access delay values compared to IPACT and oDBA.**

Keywords*— *Ethernet Passive Optical Network (EPON), Dynamic Bandwidth Allocation (DBA), Half Cycle Dynamic Bandwidth Allocation (hcDBA), Performance Evaluation.

I. INTRODUCTION

By the incredible development in internet and computer technology, users' demands of more bandwidth increase rapidly. While the backbone and local area networks have very fast infrastructures (such as 10Gbit Ethernet LAN), access networks drop down the total network capacity for users while they try to access remote sources [1][2]. In early years of Internet, carried traffic was comprised of plain text pages and images which can be carried by a limited bandwidth capacity. However, nowadays mostly carried traffic over internet is comprised of peer-to-peer file and video sharing, online real-time gaming, video streaming, on demand video and education, IP telephony and IPTV. These applications need more bandwidth in access network area and some also need quality of service in packet delay variation (PDV), packet loss and end-to-end delay cases. To overcome such demands of future internet applications, service providers always research new access technologies. Most of the service providers and network infrastructure designers start to study Fiber-to-the-Home (FTTH) architectures [3]. The most popular FTTH architecture is passive optical network (PON) architecture which has the best cost-effectiveness among fiber access architectures. In PONs for downstream, the data packets are broadcasted from the central office part of the PON, namely Optical Line Terminal (OLT) and the subscriber part, namely Optical Network Unit (ONU) collects the packets sent to itself. In downstream direction, the messages must be encrypted for undesirable access of other subscribers. For upstream direction, since ONUs are connected to the OLT over a single fiber line, a multiple access technology must be used to overcome the congestion conditions. In PON, two different multiple access types are in use; Time Division Multiple Access (TDMA) and Wavelength Division Multiple Access (WDMA).

In TDMA, there are two main standardization branches exist in network area. The first one is Gigabit PON (GPON) which is standardized by ITU-T. The second one is EPON which is standardized by IEEE 802.3ah Task Force [4]. They have published EPON standard in 2000 at 1Gbps up/down transmission capacity. By 2009, 10Gbps up/down transmission capacity has been standardized for EPON architecture [5].

The ease of implementation and cost effectiveness of EPON makes it more popular than GPON in academic studies and industrial world [1]. There are lots of studies to improve the performance, access capacity and service quality in EPON.

The scheme of bandwidth allocation in EPON can be either static or dynamic. In static allocation, a fixed-size transmission window is allocated by OLT, to each ONU, regardless of the traffic requirements at each ONU. On other hand, in the dynamic allocation, a variable-size transmission window is dynamically allocated to the different ONUs, taking into account their traffic which is expressed explicitly by each ONU. The communication between OLT and ONUs is achieved by the multipoint-control protocol (MPCP) which is defined by the IEEE 802.3ah Task Force [4].

In this paper, we present a novel Dynamic Bandwidth Allocation (DBA) algorithm for EPON. We show the basic DBA algorithms for EPON and compare our algorithm with them in terms of mean access delay, byte loss ratio and packet delay variation.

This paper is organized as follows. In section II, existing dynamic bandwidth allocation algorithms have been summarized. In section III, our proposed algorithm is introduced. In section IV, our simulation environment is described and the simulation results are presented. Finally, section V concludes the paper.

II. DBA ALGORITHMS

There is lots of dynamic bandwidth algorithms developed for EPON. The early solution for dynamic bandwidth allocation "Interleaved Polling with Adaptive Cycle Time (IPACT)" is proposed by G. Kramer et al. In [6] there are five different conditions which can be used over IPACT and according to the authers, the best variation of IPACT algorithm is to use a maximum window limit approach. In IPACT, OLT sends gate messages to the ONUs one by one in an interleaved fashion without waiting the next report message to arrive from other ONUs. If we increase the maximum window size, this can cause longer waiting time for packets in ONUs local buffers to be sent in next cycle. On the contrary, if we set the window size shorter, this will cause more GATE and REPORT transmission which bring extra overhead to the system.

For fair bandwidth distribution over highly loaded ONUs, another DBA algorithm presented which is based on Interleaved Polling with Stop (also known as offline DBA and here after called as oDBA in this paper). In this scheme, OLT waits for report messages from the entire ONUs in each cycle before it starts to send gate messages to ONUs for next cycle. By doing this, OLT can know the entire bandwidth request from the entire ONUs before it starts to grant bandwidth for ONUs. Thus, OLT can distribute the excess bandwidth fairly among highly loaded ONUs. However, oDBA inserts an idle time ***(T_{idle})*** in upstream channel which consists of Computation Time for the algorithm and Round Trip Time (RTT) between OLT and ONUs (assumed that all ONUs have the same RTT).

oDBA algorithms collect all the bandwidth demands in a cycle. An Excess Bandwidth Distribution (EBD) mechanism allots the excess bandwidth collected from lightly loaded ONUs, among highly loaded ONUs. For EBD, firstly minimum guaranteed bandwidth in a cycle for each ONU B_i^{MIN} is computed for ***N*** ONUs as in formula 1.

$$B_i^{MIN} = \frac{(T_{cycle} - N \times T_g)}{8 \times N} \times R \quad (1)$$

where T_{cycle} is cycle time, T_g is the guard time and ***R*** is the upstream channel capacity. Then, the bandwidth B_i^g needs to attribute to ONU_i is computed as in formula 2.

$$B_i^g = \begin{cases} R_i & if \quad R_i < B_i^{MIN} \\ B_i^{MIN} & if \quad R_i \geq B_i^{MIN} \end{cases} \quad (2)$$

where R_i is the bandwidth requested by the ONU_i.
After, the excess bandwidth which is not yet attributed in the current cycle is fairly distributed among all highly loaded ONUs.

Some previous works for oDBA have been carried out to fill the idle time period. In [7], the authors developed an algorithm that schedule lightly loaded ONUs in idle time period without waiting for entire ONUs to send their REPORT messages for next cycle timing. This approach is good for increasing the throughput in low loads. If the entire ONUs are highly loaded the idle period is still wasted. In [8] authors improved the idle time usage by adding a case to choose one highly loaded ONU to use idle time if no lightly loaded ONU exist in current cycle. Also in [9], another algorithm has been proposed for using idle time period which is capable of highly loaded cases. Contrary to the previous two approaches this one does not use an early allocation; instead it uses the scheme that OLT calculates supplementary granted bandwidth by using the cycle-based arrival rate of client packets.

These proposed algorithms are designed to solve the idle time problem in offline DBA. They change ONUs servicing order which can cause PDV. Proposed algorithms in [9] and [8] send extra control messages which cause extra overhead.

Another approach to decrease delay of packets in ONUs local buffers is to use queue size prediction algorithms. If an ONU is able to predict its buffer size for next cycle then it can demand the necessary bandwidth without waiting for a cycle period [10][11]. However the bursty nature of local traffic sources, the queue size prediction can waste the bandwidth by faulty predictions.

In [6], an exhaustive summary of DBA algorithms for EPON has been presented. For grant sizing there are two main approaches have been studied in literature; online DBA (Interleaved polling with adaptive cycle time approaches) and offline DBA (Interleaved Polling with Stop). Online DBA algorithms give better bandwidth utilization results because they have the capability to allocate all the bandwidth without idle time periods. On the contrary, offline DBA approaches have fair allocation among highly loaded ONUs. However, it introduces an idle time problem which can decrease the bandwidth utilization.

Our motivation to do this work is to develop a middle approach between online and offline DBA algorithms that is able to behave fairly among highly loaded ONUs and provide maximum bandwidth utilization. A grant sizing approach has been developed that switches between online and offline mode dynamically.

III. HALF CYCLING DBA ALGORITHM (HCDBA)

In this section, we present a novel DBA algorithm which is based on Interleaved Polling with Stop and use a different cycle timing control for transmission in upstream channel. We named the proposed algorithm as "Half Cycling Dynamic Bandwidth Allocation Algorithm" and hcDBA abbreviation is used in this article to identify our algorithm. hcDBA algorithm works in two modes according to the load of the upstream channel. In low loads the algorithm switch into online DBA mode which is similar to IPACT algorithm and in high loads it switches into offline DBA mode. The working mode changes respectively according to the incoming upstream bandwidth demands to the OLT.

In offline DBA (Interleaved Polling with Stop) algorithm, if OLT is able to know the bandwidth demands from entire ONUs before idle period start time as earlier as the length of idle period "T_{idle} ($T_{computation}$ + RTT)", GATE messages can be sent without any idle period in upstream channel. In hcDBA, to send GATE messages, OLT calculates bandwidth amount to be given for half of ONUs instead of entire list, if more than half of ONUs have reported their demands after the last gated ONU. Otherwise, OLT directly sends a GATE message to the following ONU in polling list. In this case, OLT is in online DBA mode which will continue until number of reported ONUs is more than half of the total. When reported ONU count reaches this amount, OLT again starts working in offline DBA mode and jump in half cycle algorithm to distribute bandwidth among ONUs. The algorithm switches into online DBA mode when the OLT cannot collect enough REPORT messages. If time slots for ONUs are so short, the upstream channel is lowly loaded. Thereby, in online mode we do not have to care about fair distribution because we know that the system have a cycle time below the desired cycle time limit. The maximum window size can be held much bigger than IPACT (limited) approach. Even the cycle time becomes longer; the algorithm changes its form to offline DBA which suppose to give a fair bandwidth distribution among ONUs in a cycle.

hcDBA uses the MPCP in EPON standardization without any upgrade necessity in control messages and ONU side implementation. hcDBA changes the OLT side algorithm for polling. In two cases, operation of the algorithm will be explained. First, the work flow diagram of GATE Timer Expire function is given in Figure 1. Second, the EBD algorithm is going to be introduced.

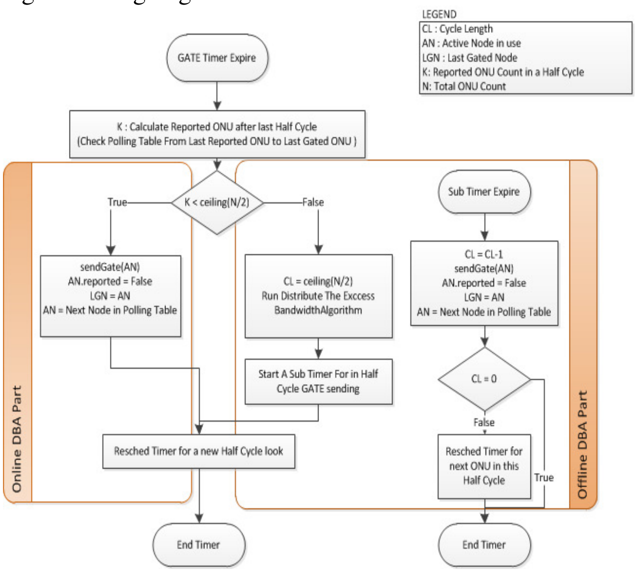

Figure 1. GATE Timer work flow in OLT

GATE timer is responsible for GATE messaging process in OLT. Each time the GATE timer expires, it calculates and prepares the parameters for novel GATE messages.

Addition to offline DBA algorithm, hcDBA algorithm needs demanded and given bandwidth information in previous cycle. Besides, for hcDBA algorithm, the instant monitoring of the last gated ONU and the last reported ONU has to be done. In hcDBA, the servicing order of ONUs during a cycle doesn't change. Thus, the last gated ONU and last reported ONU can be monitored over the polling table simply. The algorithm switches between online and offline mode according to the number of reported ONUs from last gated ONU to gate timer expiration. This parameter is related to the cycle time, requested window size and RTT of ONUs. If the window sizes of half of the ONUs are very small, they can be served in time less than RTT. Thus, when the gate timer expires, if the granted total window size is not more than RTTs for packets, the OLT cannot collect enough report to compute a new half cycle.

The main case to think about in our algorithm is the EBD in offline mode. hcDBA uses a similar method like the one used in offline DBA algorithm for EBD. However, in hcDBA the OLT has to make the EBD process with the ***K*** report messages instead of entire list (***N***). For (***N-K***) nodes, the OLT has not received the bandwidth request yet. If OLT distributes the excess bandwidth according to the ***K*** report information, the algorithm may misjudge the bandwidth demand of ONUs in a full cycle and EBD can be unfair. The half cycle that OLT is going to give grants can take more or less "excess bandwidth-highly loaded node" ratio than consecutive cycles. In a situation like this, unfairness takes places between two ONU groups. For this reason, in hcDBA algorithm, while the excess bandwidth distribution is being calculated for a new half cycle for ***N/2*** ONU, the algorithm does not make the decision just over ***K*** report messages (note that always, $K \geq \lceil N/2 \rceil$). It also includes the excess bandwidth and highly loaded ONUs information for (***N-K***) ONUs from the previous bandwidth requests and grants. If the bandwidth demands are less than excess bandwidth in previous half cycle, more bandwidth can be used in current half cycle. Otherwise, the excess bandwidth is distributed in fair for next half cycle according to the situation of current half cycle.

ONUs are examined in two groups in EBD algorithm such as reported and unreported. The algorithm needs entire ONUs requests to distribute the bandwidth fairly. For ONUs of which reports have not arrived to OLT, the needed information will be generated based on their previous cycle. The details about the process of excess bandwidth algorithm are given below. Minimum bandwidth is calculated same as oDBA as in formula 1.

Excess bandwidth calculation is a bit more different than oDBA approach. The usable excess bandwidth in a half cycle cannot be measured just with the ONUs requests in current half cycle. To distribute the excess bandwidth fairly to entire ONUs in PON, OLT also take into consideration the bandwidth requests of ONUs served in previous half cycle. Since the algorithm cannot use all the excess bandwidth (B^{EXCESS}), it will calculate the usable excess bandwidth B^{USABLE} by using B^{EXCESS} values. B^{EXCESS} is calculated as the sum of excess bandwidth amount of reported (***K***) and previous excess bandwidth amount of unreported (***O: N-K***) nodes. The excess bandwidth values are calculated separately for ***K*** ONUs and ***O*** ONUs. Unused bandwidth of ***K*** ONUs from last report information is given in formula 3. Unused bandwidth of ***O*** ONUs (***O: N-K***), last ***O*** ONUs from previous report information kept in polling table is given in formula 4.

 ISBN: 978-1-61208-103-8

$$B_K^{EXCESS} = \sum_{i=1}^{K}[B_i^{MIN} - B_i^{REQ}] \quad (B_i^{MIN} > B_i^{REQ}) \tag{3}$$

$$B_O^{EXCESS} = \sum_{j=K}^{N}[B_j^{MIN} - B_j^{REQ}] \quad (B_j^{MIN} > B_j^{REQ}) \tag{4}$$

B_K^{USABLE} will show the total maximum excess bandwidth available to use for ***K*** ONUs. In processing half cycle, bandwidth arrangement will be done just for $\lceil N/2 \rceil$ ONUs. In a half cycle, despite EBD is done through ***K*** ONUs, just the distribution of $\lceil N/2 \rceil$ ONUs will be determined. B_K^{USABLE} calculation is done as follows.

$$B_K^{USABLE} = \begin{cases} B_K^{EXCESS} + B_L^{UNUSED} & if \quad \frac{R_K^H}{R_K^H + R_O^H} \times B^{EXCESS} \geq B_K^{EXCESS} \\ \frac{R_K^H}{R_K^H + R_O^H} \times B^{EXCESS} + B_L^{UNUSED} & if \quad \frac{R_K^H}{R_K^H + R_O^H} \times B^{EXCESS} < B_K^{EXCESS} \end{cases} \tag{5}$$

Here, B_L^{UNUSED} is the unused excess bandwidth according to extra bandwidth demand and excess bandwidth of ***N/2*** nodes served just before. (This calculation should be done by checking the polling table each time needed, because of the dynamically switching between online DBA and offline DBA modes. There may be some ONUs served according to online DBA between the previous and current half cycle. If the bandwidth demand of overloaded amount exceeds the excess bandwidth, they will be assumed as zero.) R_K^H indicates the total bandwidth demand of highly loaded ones of ***K*** ONUs and R_O^H indicates the total bandwidth demand of highly loaded ones in previous cycle of ***O*** ONUs.

If the excess bandwidth amount of ***K*** nodes is lower than the bandwidth amount portion of ***K*** nodes in total excess bandwidth (this means that ONUs are overloaded in processing half cycle), algorithm marks the whole excess bandwidth for ***K*** ONUs as usable. If it results in other way, it means that ONUs in previous half cycle are overloaded. In this case, the algorithm will distribute the assigned excess bandwidth to current half cycle considering total needs of entire ONUs in PON, in order to let ONUs in previous cycle to have more excess bandwidth in following cycle. Besides, if unused excess bandwidth exists for ***N/2*** ONUs from previous half cycle, this unused value is also added to excess bandwidth. With combination of these calculations, hcDBA tries to guarantee fair distribution between respective half cycles.

After calculation of B_K^{USABLE}, for each half cycle (just for ***N/2*** ONUs, always ***K ≥ N/2***), the bandwidth assigned for each ONU will be calculated as below:

$$B_i^{g} = \begin{cases} R_i & if \quad R_i \leq B_i^{MIN} \\ R_i & if \quad R_i > B_i^{MIN} \wedge B_K^{USABLE} \geq (R_K^H - B_i^{MIN} \times K) \\ B_i^{MIN} + \frac{R_i}{R_K^H} \times B_K^{USABLE} & if \quad R_i > B_i^{MIN} \wedge B_K^{USABLE} < (R_K^H - B_i^{MIN} \times K) \end{cases} \tag{6}$$

IV. PERFORMANCE EVALUATION

In this section, we present simulation results to verify our analysis and demonstrate the performance of the proposed hcDBA algorithm. We compare the results obtained from the hcDBA algorithm with IPACT and offline DBA algorithms. We use the same basis for each algorithm on the simulation.

We consider an EPON access network consisting of 16 ONUs connected to an OLT through a passive coupler. All ONUs are assigned a downstream and an upstream propagation delay (from ONU to OLT). We fix the distance between the coupler and OLT and distances between ONUs and the coupler about 10 km (about 0.05 ms). We compare the algorithms in two cases; 1Gbps upstream channel for EPON and 10Gbps upstream channel for 10G-EPON. The algorithms are compared with four different priority classes described as below.

TABLE I. TRAFFIC HYPOTHESIS [9]

	CoS1 Premium	*CoS2 Silver*	*CoS3 Bronze*	*CoS4 BE*
Traffic Ratio	10 %	10 %	30 %	50 %
Packet size (in Bytes)	70	70	50,500,1500	50,500,1500
Source and Burstiness	CBR	CBR	PPBR/ μ=1.4	PPBP/μ=1.4
Burst Length (# of Packets)	CBR	CBR	10	20

TABLE II. SIMULATION PARAMETERS

Parameter	*Value(Case1)*	*Value(Case2)*
No. of ONUs	16	16
Upstream Bandwidth, R	1 Gbit/s	10 Gbit/s
Maximum cycle time for hcDBA and oDBA	2ms	2ms
Maximum transfer window size for IPACT and hcDBA	15 KB (IPACT) 30 KB (hcDBA)	150 KB (IPACT) 300 KB (hcDBA)
Guard Time	5μs	5μs

For Premium and Silver traffic, we use CBR (Constant Bit Rates) sources. To generate self-similar traffic of Ethernet LAN (Bronze and Best Effort BE classes), we use an aggregation of multiple sources of Poisson Pareto Burst Process (PPBP), so called Pareto-distributed ON-OFF [12]. In hcDBA, since our first goal is to improve the bandwidth utilization with fairness among ONUs, we also give some results without service classes to show the overall utilization performance of hcDBA.

Simulations were done using discrete event network simulation tool (ns2.34). Table II shows the simulation parameters for each algorithm.

We shall start the performance comparison of hcDBA algorithm with others, considering byte loss ratio. In Figure 2a the byte loss ratios for mono-service traffic are shown. Only oDBA algorithm has byte loss in 0.9 offered load in mono-service traffic condition. Since oDBA provides bad bandwidth utilization compared to others. In Figure 2b the byte drop results of three algorithms for multi-service traffic is given. Drops occur only in the lowest service class in each algorithm after 0.7 offered load. hcDBA gives the best performance while considering byte loss ratio.

 ISBN: 978-1-61208-103-8

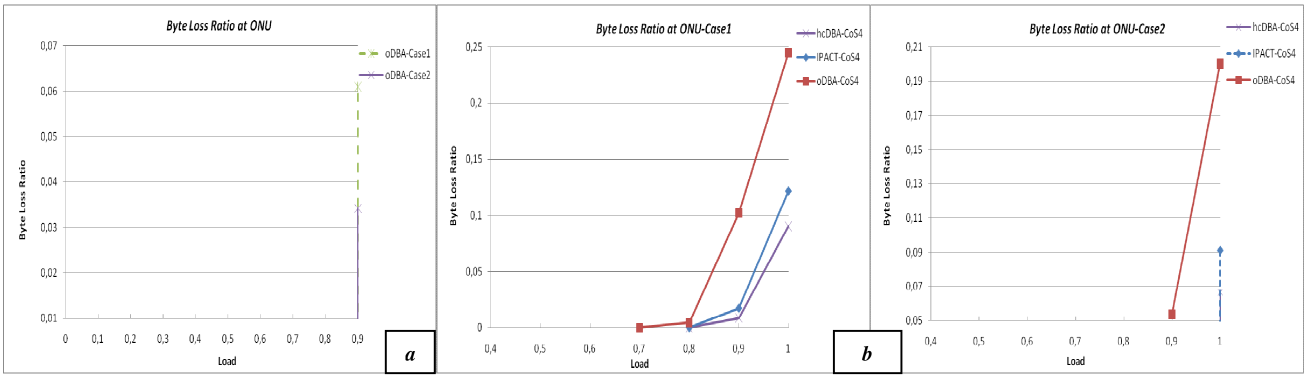

Figure 2. a) Byte Loss Ratio in Case1 and Case2 with mono-service class b) Byte Loss Ratio in Case1 and Case2 with multi-service classes

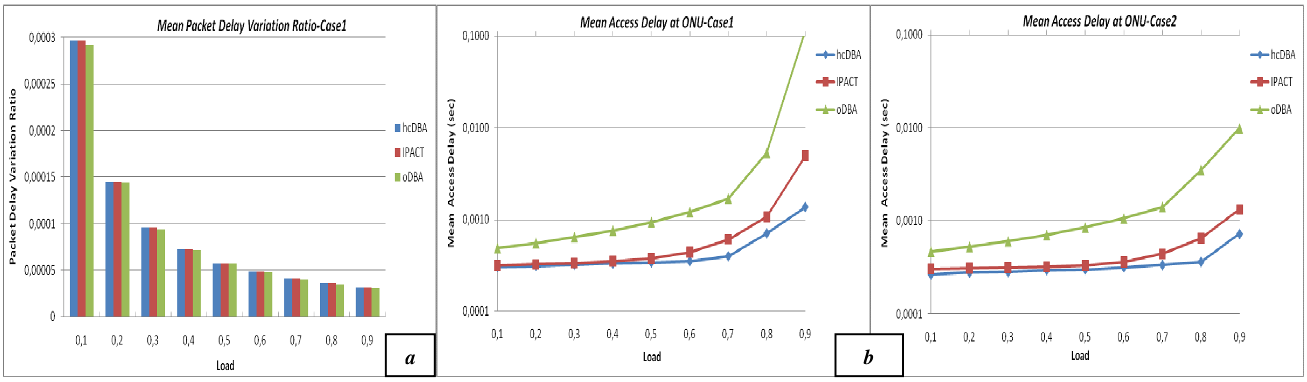

Figure 3. a) Packet Delay Variations in Case1 b) Mean Access Delay with Mono-Service Traffic

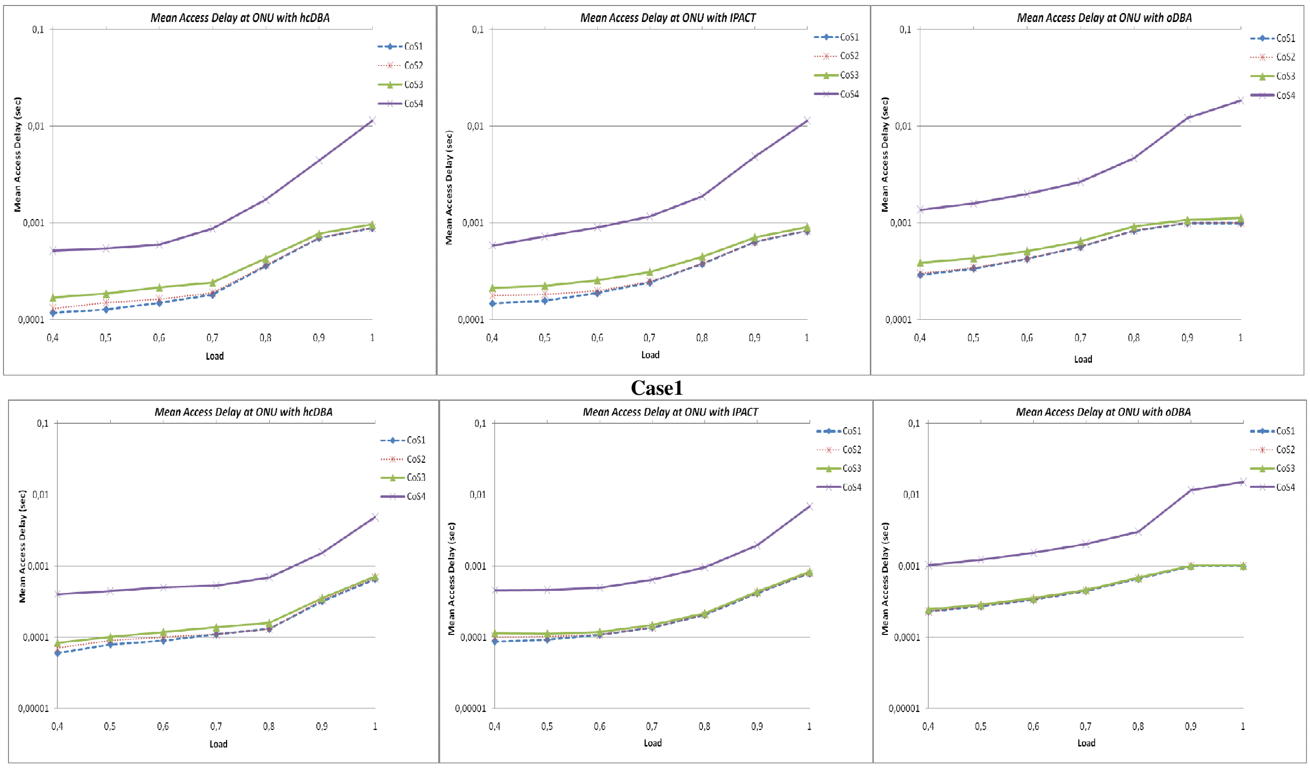

Figure 4. Mean Access Delay with Multi-Service Traffic

 ISBN: 978-1-61208-103-8

Figure 3a shows the PDV of each algorithm in Case1. From the figure it can be seen that all the algorithms have similar PDV values in the same loads. The used algorithm has no significant affect over PDV in PON network. Thus, Case2 results are not given.

Figure 3b shows mean access delay values for mono-service traffic of both cases. hcDBA and IPACT algorithms are better than oDBA. In all algorithms, the mean access delay is below 1ms in low loads. When the load increases, the mean access delays increase as expected. hcDBA gives better performance in both cases in terms of access delay. When the bandwidth rate is 10Gbps, hcDBA algorithm success increases as seen in Figure 4, and hcDBA algorithm gives mean access delay below 1ms at 0.9 load.

Figure 4 shows mean access delays in Case1 and Case2 with multi-service traffic. For each priority classes except CoS4 (Best Effort traffic) hcDBA and IPACT stays under 1ms in every offered load values. hcDBA algorithm is better in all cases of each service class. The lowest priority class is worst in all conditions. In simulations, we use Head of Line (HoL) priority scheduling at ONUs. Thus, lower priority traffic has to wait in buffers each time if there is not enough bandwidth has been given by the OLT. hcDBA gives better mean access delay results than IPACT because that it can give more bandwidth to the highly loaded ONU if there is excess bandwidth exists thanks to the low demands of other ONUs.

We also check fairness of the proposed algorithm among ONUs. When hcDBA works in offline mode, in each cycle time, ONUs are separated into two groups. We must be sure that the algorithm distributes the excess bandwidth fairly between two ONU groups. For this reason, in each cycle, we check the difference of EBD values of hcDBA algorithm with the EBD values if it distributes excess bandwidth as standard offline DBA. This difference ratio is 0 in low loads. Since, in low loads excess bandwidth is enough for all highly loaded nodes. There is only a small difference occurs among 0.7 to 0.9 offered loads (%1 at 0.9 load, %0.01 at 0.8 load and %0.0006 at 0.7 load).

V. CONCLUSION

In this paper, we have proposed a novel dynamic bandwidth allocation algorithm that stays between offline DBA and online DBA algorithms. Our first aim is to eliminate idle time problem in offline DBA algorithm while keeping fair EBD scheme of offline DBA algorithm. In hcDBA, we distribute the excess bandwidth in two half cycle and we also switch to online DBA mode according to the incoming traffic load in each cycle time. Besides, by simulation results, we evaluate the performance of the proposed algorithm with mono-service and multi-service traffic under two cases as 1Gbps and 10Gbps upstream channel bandwidth rates. The performance improvement is measured in terms of mean access delay and byte loss ratio.

We have compared our DBA algorithm with IPACT and oDBA algorithms. hcDBA shows better performance both in mean access delay and byte loss ratio values. The simulation studies provide that hcDBA is almost as fair as offline DBA and has better bandwidth utilization than IPACT algorithm.

Our algorithm's advantages compared to other algorithms proposed to eliminate idle time problem in offline DBA algorithm can be listed as;

- Uses the standard MPCP control messages defined in EPON standard and does not need any change in ONUs.
- Can be combined with different QoS approaches.
- Does not change the service order in polling table therefore, it does not cause additional PDV.
- While it eliminates idle time period in offline DBA, hcDBA does not need extra GATE/REPORT messages, as a result hcDBA introduce less overhead in upstream and downstream channels.

As a future work the hcDBA algorithm can be improved with addition of intra-ONU and inter-ONU quality of service approaches to obtain better results in multi-service environments.

ACKNOWLEDGMENT

This work is supported partially by the European EURO-NF project. This work is also a part of ongoing PhD thesis titled "Next Generation Optical Access Networks" at Istanbul University, Institute of Physical Sciences.

REFERANCES

[1] C. Lam, "Passive Optical Networks: Principles and Practice", Burlington, MA: Academic, 2007.

[2] F. Effenberger and T.S. El-Bawab, "Passive Optical Networks (PONs): Past, present, and future", Elsevier, Optical Switching and Networking 6, pp 143-150, 2009.

[3] F. Effenberger, D.Clearly, O. Haran, G.Kramer, R.D. Li, M. Oron, and T. Pfeiffer, "An introduction to PON technologies", IEEE Commun. Mag., vol.45, no.3, pp.s17–s25, Mar. 2007.

[4] IEEE 802.3ah task force webpage: http://www.ieee802.org/3/efm [accessed on February 2010]

[5] G. Krammer, "10G-EPON: Drivers, Challenges, and Solutions", Optical Communication, 2009. ECOC '09. 35th European Conference on , vol., no., pp.1-3, 20-24 Sept. 2009, Vienna Austria.

[6] G. Kramer, B. Mukherjee, and G. Pesavento, "IPACT: a dynamic protocol for an Ethernet PON (EPON)", IEEE Commun. Mag., 2002, 40, (2), pp. 74–80.

[7] C.M. Assi, Y.Ye, S. Dixit, and M.A. Ali, "Dynamic bandwidth allocation for quality-of-service over Ethernet PONs", IEEE Journal on Selected Areas in Communications, Vol. 21, No. 9, Nov. 2003, pp. 1467-1476.

[8] J. Zheng, "Efficient bandwidth allocation algorithm for Ethernet passive optical networks" Communications, IEE Proceedings- , vol.153, no.3, pp.464-468, 2 June 2006

[9] T.D. Nguyen, T. Eido, and T. Atmaca, "An Enhanced QoS-enabled dynamic bandwidth allocation mechanism for Ethernet PON", Emerging Network Intelligence, pp.135-140, 11-16 Oct. 2009

[10] H.J. Byun, J.M. Nho, and J.T. Lim, "Dynamic bandwidth allocation algorithm in Ethernet passive optical networks," Electronics Letters, vol. 39, no. 13, June 2003, pp. 1001–2.

[11] Y. Luo and N. Ansari, "Limited sharing with traffic prediction for dynamic bandwidth allocation and QoS provisioning over Ethernet passive optical networks," OSA J. Opt. Net., vol. 4, no. 9, Sept. 2005, pp. 561–72.

[12] W. Willinger, M.S. Taqqu, R. Sherman, and D.V Wilson, "Self-Similarity through High-Variability: statistical analysis of Ethernet LAN traffic at the source level", Networking, IEEE/ACM Transactions on , vol.5, no.1, pp.71-86, Feb 1997

 ISBN: 978-1-61208-103-8