Simple and Feasible Dynamic Bandwidth and Polling Allocation for XGPON

Man-Soo HAN
Dept. of Information and Communications, Mokpo National Univ., Korea
mshan@mokpo.ac.kr

***Abstract*—In this paper, we propose a simple and feasible dynamic bandwidth allocation (SFDBA) algorithm in order to utilize the unallocated bandwidth and to achieve the implementation feasibility. SFDBA is based on an immediate allocation with colorless grant (IACG) algorithm but SFDBA uses only a single available byte counter and a single down counter for multiple queues of a same service class. Since multiple queues share the same available byte counter, the unallocated bandwidth of a queue can be utilized by another queues. For better service fairness, SFDBA changes the starting queue of scheduling in a round-robin manner. Using simulations, we show that SFDBA is superior to existing methods in mean delay, frame delay variance and frame loss rate.**

Keywords—Keywords: Passive Optical Network, Dynamic Bandwidth Allocation, Polling, XGPON

I. Introduction

XGPON (10-gigabit-capable Passive Optical Network) consists of an optical line termination (OLT) and multiple optical network units (ONUs) [1]. In order to allocate non-collision transmission slots to ONUs, the OLT receives requests from ONUs and performs dynamic bandwidth allocation (DBA).

In the XGPON technology, the OLT has to produce DBA result in every frame duration which has a fixed to 125 μs [1], [2]. Therefore, it is crucial that a DBA algorithm can be run within the frame duration in XGPON. Many algorithms have been proposed for GPON DBA [5]–[11]. To the best of our knowledge, however, only a GigaPON access network (GIANT), an immediate allocation with colorless grant (IACG), and efficient bandwidth utilization (EBU) have been physically implemented. These algorithms are simple and compliant to the GPON standards.

The GIANT algorithm is the first DBA algorithm that is physically implemented [6], [7]. In GIANT, each queue has a down counter for bandwidth allocation. The OLT can allocate bandwidth to a queue only when the down counter of the queue has expired. The simplicity of GIANT provides the implementation feasibility. Although GIANT provides good performance, it degrades performance since a request of a queue can not be granted until the down counter has expired.

In [8], IACG algorithm had been introduced. In IACG, each queue has an available byte counter and a down counter in order for the fast bandwidth allocation. The OLT can immediately allocate a bandwidth to a queue if its available byte counter has a positive value. The available byte counter is decreased by the grant amount. The available byte counter is recharged when its down counter has expired. It was shown that IACG outperforms GIANT in [8]. Although IACG provides good performance, it does not effectively utilize the unused bandwidth of queues. The unused bandwidth of a queue cannot be used by another queues. It is desirable that the unused bandwidth is utilized by queues whose request sizes are larger than their reserved service bandwidth.

In [9], EBU algorithm had been introduced. EBU improves the unallocated bandwidth problem of IACG. In EBU, each queue has an available byte counter and a down counter like IACG. To utilize the unused bandwidth, the available byte counter can be negative. In addition, at the end of scheduling, the unused remainder of the available byte counter is added to the negative available byte counters. It was shown that EBU increases the utilization of the unused bandwidth compared to IACG in [9]. However, the operation of the available byte counter is complex and an extra stage is required for the update of the available byte counters. Therefore, the implementation complexity of EBU is higher than that of IACG.

In this paper, we propose a simple and feasible dynamic bandwidth allocation (SFDBA) algorithm in order to utilize the unused bandwidth and to achieve the implementation feasibility. SFDBA is based on IACG but it uses only a single available byte counter and a single down counter for a group of queues of a same service class. Since multiple queues share the common available byte counter, the unused bandwidth of a queue can be utilized by another queues. For better service fairness, SFDBA changes the starting queue of scheduling in a round-robin manner. Since only a common counter is used for a group of queues, SFDBA is simpler than IACG. Hence SFDBA is more feasible than IACG. Using simulations, we show that SFDBA is superior to IACG in mean delay, frame delay variance and frame loss rate. In addition, we illustrate that SFDBA is better than EBU in mean delay and frame delay variance despite that SFDBA is simpler than EBU.

In this paper, in the section II, we explain the bandwidth utilization problem of IACG algorithm and describe the proposed SFDBA algorithm. Then, we explain the polling mechanism of the proposed SFDBA algorithm in the section III. In the section IV, we evaluate performance of each algorithm and show that SFDBA algorithm outperforms IACG algorithm using simulations. Also we compare performance of SFDBA

Manuscript received June 6, 2013.
M.-S. Han is with the Dept. of Information and Communication, Mokpo National Univ., Jeonnam, Korea (phone: +82-61-450-2744; fax: +81-61-450-6470; e-mail: mshan@mokpo.ac.kr).

algorithm to that of EBU algorithm using simulations. The conclusions are given in the section V.

II. Simple and Feasible Dynamic Bandwidth Allocation

Fig. 1 shows the XGPON system. The XGPON system consists of an OLT and N ONUs. To support multiple service classes, ONU i maintains a queue q_{ij} for a service class j. A service class is known as a transmission container (T-CONT) type in XGPON technology. In this paper, we consider three service classes: T-CONT types 2, 3 and 4. Since a static bandwidth allocation is used for T-CONT type 1 in XGPON, we do not consider the T-CONT type 1. An incoming frame from users to ONU i is saved in q_{ij} if its T-CONT type is j. A queue has its unique Allocation Identifier (AllocID).

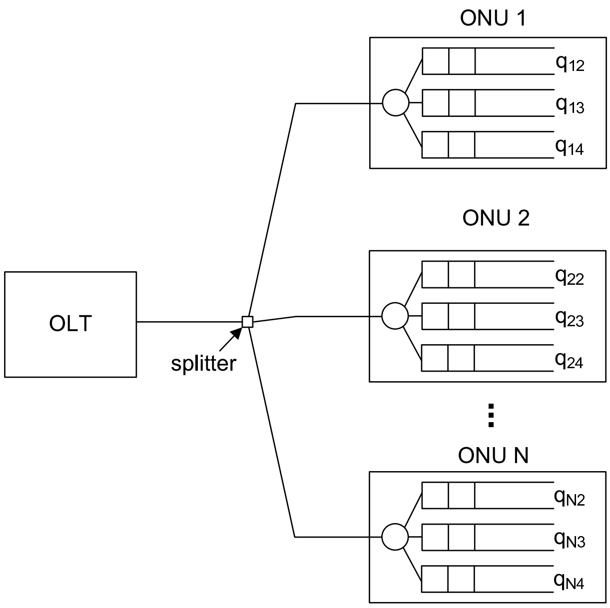

Fig. 1. XGPON system

In the downstream direction, the OLT broadcasts frames to each ONU. When a frame arrives from the OLT, each ONU accepts only if the destination of the frame is matched. In the upstream direction, ONUs transmit frames to the OLT in a time division multiple access manner. To assign a non-overlapping transmission time-slot to each ONU, the OLT performs dynamic bandwidth allocation (DBA).

In XGPON, every operation of OLT and ONU is synchronised with a fixed frame duration (FD) which is 125 μs long. The OLT collects the requests from ONUs, performs the DBA and produces the DBA result in every FD. The DBA result has the transmission slot information for an upcoming single FD. To receive the requests from ONUs, the OLT assigns a dynamic bandwidth report upstream (DBRu) transmission slot to a queue of an ONU. Using the DBRu slot, the queue sends its queue length to the OLT.

In IACG algorithm, a queue q_{ij} has two service parameters: a service interval S_{ij} with unit of FD and a maximum allocation byte A_{ij} that can be maximally allocated to q_{ij} during its service interval S_{ij}. The queue q_{ij} has a down counter T_{ij} which is decreased by 1 for each FD. In addition, the queue q_{ij} maintains an available byte counter V_{ij} which denotes the remaining service bytes during its S_{ij}.

When the down counter T_{ij} has expired, T_{ij} is recharged to S_{ij} and the available byte counter V_{ij} is reset to A_{ij}. Let r_{ij} be the request of queue q_{ij}. In IACG, the OLT grants the minimum of the request r_{ij} and the value of V_{ij}. The OLT immediately decreases each the request r_{ij} and the value of V_{ij} by the grant amount. The maximum grant size of q_{ij} is limited by A_{ij}.

We now explain the bandwidth utilization problem of IACG. In IACG, a queue cannot use an unused bandwidth of another queue. For example, consider two queues of T-CONT type j, q_{1j} and q_{2j}. Suppose that $r_{1j} = 100$, $V_{1j} = 0$, $r_{2j} = 0$, and $V_{2j} = 100$. Than IACG will not grant any bandwidth for queue q_{1j} despite V_{2j} remains unused. This inefficiency of IACG degrades the DBA performance.

In order to solve the utilization problem of IACG, we propose SFDBA algorithm. In SFDBA, a queue does not have the individual counters T_{ij} and V_{ij}. Instead, all queues of the same T-CONT type share a single down counter and a single available byte counter. Let T_j and V_j be the common down counter and the common available byte counter of T-CONT type j, respectively. Also let S_j and A_j be the common service interval and the common maximum allocation bye of T-CONT type j, respectively.

The common down counter T_j is decreased by 1 for each FD. When the down counter T_j has expired, T_j is recharged to S_j and the common available byte counter V_j is reset to A_j. The common maximum allocation byte A_j is calculated by

$$\frac{A_j}{S_j} = \sum_{i=1}^{N} \frac{A_{ij}}{S_{ij}}. \tag{1}$$

In SFDBA, the OLT grants the minimum of the request r_{ij} and the value of V_j. The OLT immediately decreases each of the value of V_j and the request r_{ij} by the grant amount.

Since the single available byte counter is used for multiple queues, the upstream channel can be monopolized by a queue. To prevent the monopolization, the maximum grant size of a queue can be limited. In this case, the OLT grants the minimum of the request r_{ij}, the maximum grant size and the value of V_j. Another method to prevent the monopolization is traffic policing using the leaky bucket method [13] to control the traffic arrival rate of a queue. Since traffic policing is required in XGPON, we assume that the leaky bucket method is used in this paper for simplicity.

Fig. 2 shows the pseudo code of SFDBA for T-CONT type j where $j = 2, 3, 4$. Scheduling is performed in the order of T-CONT types 2, 3, and 4. In Fig. 2, the round robin pointer P_j denotes the ONU number at which scheduling starts. To provide the service fairness, the starting ONU number in the scheduling is changed in the round robin fashion. The variable F is the remaining bytes of FD and its initial value is 38,880 bytes when the upstream channel speed is 2.5 Gbps. The variable g_{kj} is the grant amount of queue q_{kj}. The

```
// k = ONU number
// j = T-CONT type
k = stop = P_j;
while(1){
    if (V_j > 0 and F > 0){
        g_kj = min(V_j, r_kj, F);
        V_j = V_j - g_kj;
        r_kj = r_kj - g_kj;
        F = F - g_kj;
    } else if (alloc_end = 0 and F = 0){
        P_j = k;
        alloc_end = 1;
    }
    T_j = T_j - 1;
    if (T_j = 0){
        T_j = S_j;
        V_j = A_j;
        PF_kj = 0;
    }
    k++;
    k = k mod N;
    if (k = stop) break;
}
```

Fig. 2. Pseudo code of SFDBA for T-CONT type 2, 3 and 4

variable $alloc\_end$ means the end of the allocation. When $alloc\_end = 1$, the allocation has ended. The initial values of g_{kj} and $alloc\_end$ are zero. The variable PF_{kj} denotes the polling flag of q_{kj}. The polling flag PF_{kj} is used in the polling operation of SFDBA and will be explained in Section III.

For the previous utilization example of IACG, we apply SFDBA by assuming $V_j = V_{1j} + V_{2j} = 100$. Then SFDBA will grant 100 byte for queue q_{1j}. Unlike IACG, the unused bandwidth of q_{2j} is used by q_{1j} in SFDBA.

If the total sum of requests is less than the size of FD, then the upstream channel cannot be fully allocated. To utilize the unallocated remainder of FD, the remainder is evenly distributed to each ONU in IACG [8]. To distinguish the remainder from normal grants, the remainder has T-CONT type 5 and is called as the colorless grant. An ONU can send packets arrived after the ONU sent its request using the colorless grant. Note that ONU does not has a queue with T-CONT type 5. Using the colorless grant, ONU transmits packets from queues in the order of T-CONT types 2, 3 and 4. We use the colorless grant scheme of IACG in SFDBA. At the end of DBA, the colorless grant of ONU k is given by F/N where F is the remaining bytes of FD.

III. POLLING OPERATION OF SFDBA

Now we describe the polling method of SFDBA. Fig. 3 describes the transmission timing diagram of XGPON. In Fig. 3, the maximum distance between an OLT and and ONU is 20 Km. Also, U_i means the i-th upstream FD and D_i denotes the i-th downstream FD. RTT is the round trip time between an ONU and an OLT. If the distance between the OLT and the ONU is 20 Km, RTT is 200 μs. The variable T_O is the ONU response time to receive the grant result from the OLT and to prepare an upstream response. In this paper, $T_O = 35$ μs as in [1].

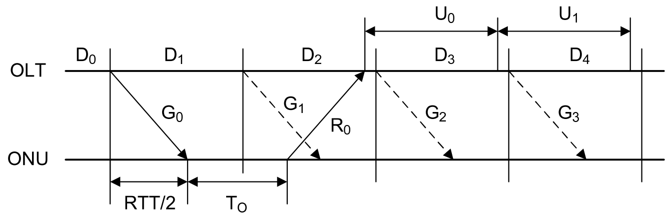

Fig. 3. Transmission timing diagram

In this paper, we modify the polling mechanism of IACG [8]. In Fig. 3, G_i is the i-th grant result and R_0 is the first request of the ONU. The DBRu field is used by the ONU to transmit its request. The i-th grant result is produced by the DBA operation in D_i and is transmitted to the ONU at the beginning of the downstream FD, D_{i+1}. When the ONU receives the grant result, it waits for T_O, then sends its request R_0 and data packets during the upstream FD, U_0. The request R_0 is the total length of packets in a queue. Since the request R_0 is sent before data packets, it does not reflect the grant result G_0. The requests received during U_0 are used in the DBA at the downstream FD, D_4. Since SFDBA may produces grant results for the ONU during D_1, D_2, and D_3, the grant results must be reflected in R_0. Thus, in order to obtain the correct request, the OLT has to subtract the grant results $G_0, \cdots, G_3$ from R_0 at the beginning of D_4. Therefore, the OLT needs to remember the four most recent grant results for obtaining the correct request.

We now explain the DBRu field allocation mechanism of SFDBA. Queue q_{kj} has a polling flag PF_{kj}. The OLT can allocate the DBRu field to queue q_{kj} when $PF_{kj} = 0$. If the OLT allocate the DBRu field to q_{kj}, the flag PF_{kj} is set to 1. The flag PF_{kj} is reset to 0 when the down counter T_j has expired. Therefore, queue q_{kj} has a chance to receive the DBRu field once per the service interval S_j.

Fig. 4 describes the pseudo code of the SFDBA polling operation. In Fig. 4, L_j is the ONU number at which the polling operation starts for T-CONT type j. The variable DBR_{kj} means whether or not the DBRu field is assigned to q_{kj}. The variable DS is the size of the DBRu field. In the XGPON standard, the size DS is 4 bytes. The variable $poll\_end$ represents the end of polling operation. The polling flag PF_{kj} is set to 0 when the down counter T_j has expired as shown in Fig. 2.

```
// k = ONU number
// j = T-CONT type
k = stop = L_j;
while(1){
    DBR_kj = 0;
    if (PF_kj = 0 and F > 0){
        PF_kj = 1;
        DBR_kj = 1;
        F = F − DS;
    } else if (poll_end = 0 and F = 0){
        L_j = k;
        poll_end = 1;
    }
    k++;
    k = k  mod N;
    if (k = stop) break;
}
```

Fig. 4. Pseudo code of SFDBA polling operation

IV. PERFORMANCE EVALUATION

A. Comparison to IACG

Now we compare performance of SFDBA, IACG and predictive-colorless-grant offset-based scheduling with flexible intervals (PCG-OSFI) [10] using simulations. We consider an XGPON system with $N = 16$, the maximum ONU line rate of 200 Mbps, the upstream channel rate of 2.5 Gbps. Also suppose that the maximum round trip time between the OLT and ONUs is 200 μs, and the ONU response time is 35 μs. The size of a queue q_{ij} is 1 Mbytes.

For the T-CONT type 2 of IACG, we set $A_{i2} = 7812$, $S_{i2} = 5$, which is equivalent to 100 Mbps. For the T-CONT type 3 of IACG, we set $A_{i3} = 15624$, $S_{i3} = 10$. That is, 100 Mbps is given to the T-CONT type 3. For the T-CONT type 4 of IACG, $A_{i4} = 15624$, and $S_{i4} = 10$, which is equivalent to 100 Mbps. For the T-CONT type 2 of SFDBA, we set $S_2 = 5$ and we obtain $V_2 = 7812 \times N$ from Eq. (1). For the T-CONT types 3 and 4, we set $S_3 = S_4 = 10$. Then we have $V_3 = V_4 = 15624 \times N$. The reserved bandwidth for each T-CONT type of PCG-OSFI is equal to that of IACG.

We use the self-similar traffic model of [12] where each ONU is fed by a number of Pareto distributed on-off processes. The shape parameters for the on and off intervals are set to 1.4 and 1.2, respectively. Hence, the Hurst parameter is 0.8. The frame size follows the tri-modal distribution [12], where the frame sizes are 64, 500, and 1500 bytes and their load fractions are 60%, 20% and 20%, respectively as in [6]. Each simulation is performed until the total number of frames transmitted by ONUs exceeds 10^9 for each algorithm.

Traffic is balanced so that each ONU and each queue has an identical traffic load fraction. The traffic load of each ONU varies from 0.1 to 0.99. Figs. 5 and 6 show the mean delay of each algorithm. Note that the offered load means the input traffic load of a single ONU in the Figs. As we can see from Figs. 5 and 6, the proposed method outperforms other methods in mean delay.

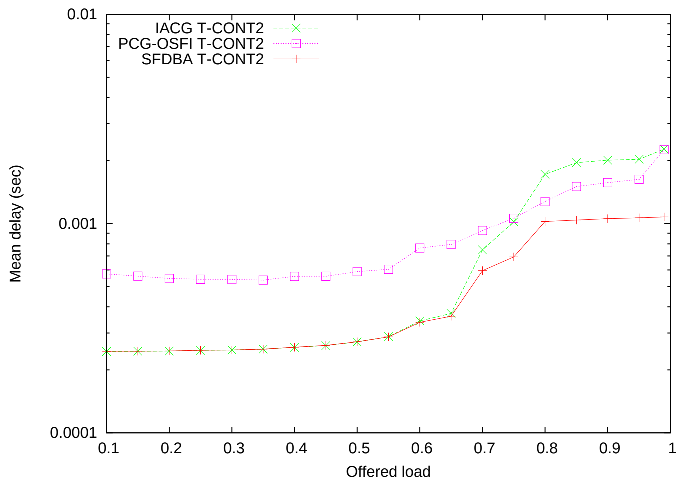

Fig. 5. Mean delay of T-CONT type 2

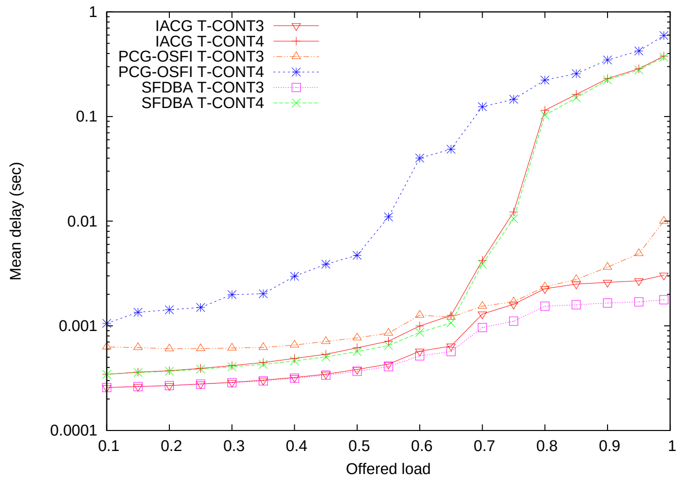

Fig. 6. Mean delay of T-CONT types 3 and 4

Figs. 7 and 8 depict the frame delay variance of each algorithm. Figs. 7 and 8 show that the proposed method outperforms other methods in frame delay variance. Fig. 9 illustrates the frame loss rate of each algorithm for T-CONT types 2 and 3. Thanks to the utilization of unused bandwidth, the loss rate of SFDBA is better than those of other methods.

B. Comparison to EBU

To increase the DBA efficiency, EBU can allocate an additional DBRu field to a queue [9]. In EBU, the OLT allocates the DBRu field once per its service interval like IACG. Also the OLT can allocate an additional DBRu field to a queue. If the grant $g_{ij} > 0$ and the DBRu field is not allocated to the

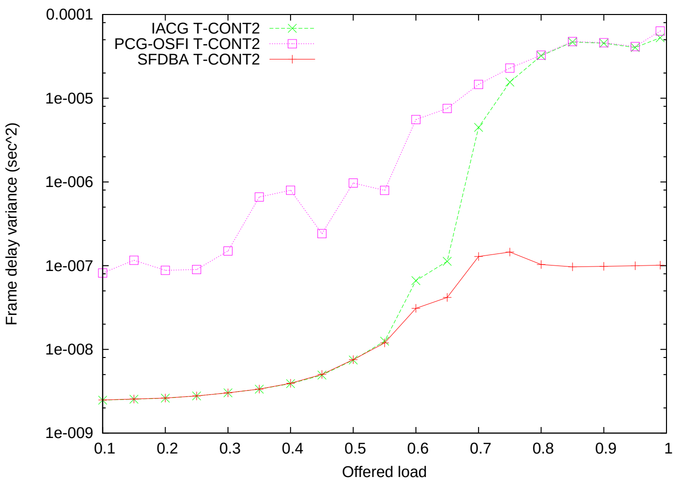

Fig. 7. Frame delay variance of T-CONT type 2

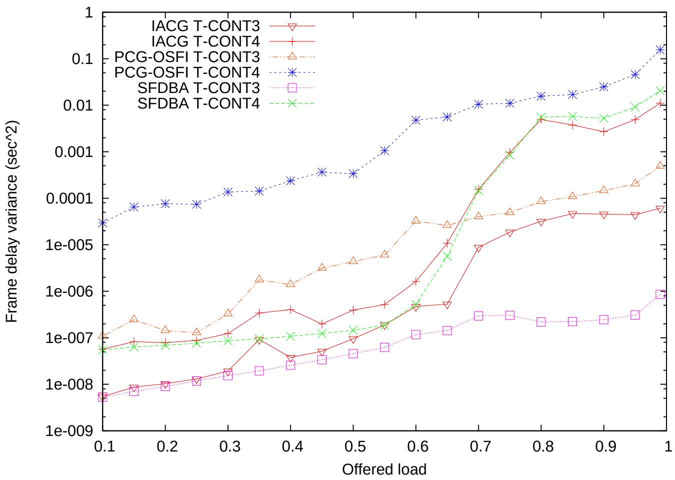

Fig. 8. Frame delay variance of T-CONT types 3 and 4

queue q_{ij} then the OLT allocates the additional DBRu field to the queue q_{ij}. It was shown that the additional DBRu is very effective in [9]. To compare performance of EBU with that of SFDBA, we apply the additional DBRu mechanism to the polling operation of SFDBA in this subsection.

We use the same system, the same parameters, and the same traffic of the subsection IV-A. For the T-CONT type 2 of EBU, we set $A_{i2} = 7812$, $S_{i2} = 5$, which is equivalent to 100 Mbps. For the T-CONT type 3 of EBU, we set $A_{i3} = 15624$, $S_{i3} = 10$. That is, 100 Mbps is given to the T-CONT type 3. For the T-CONT type 4 of EBU, $A_{i4} = 15624$, and $S_{i4} = 10$, which is equivalent to 100 Mbps. We do not use the non-assured T-CONT type 3 of EBU.

Figs. 10 and 11 show the mean delay of each algorithm. As we can see from Figs. 10 and 11, the proposed SFDBA outperforms the EBU algorithm in mean delay. In EBU, an unused bandwidth of a queue q_{ij} is added to the available byte counter of another queue only when the down counter T_{ij} has

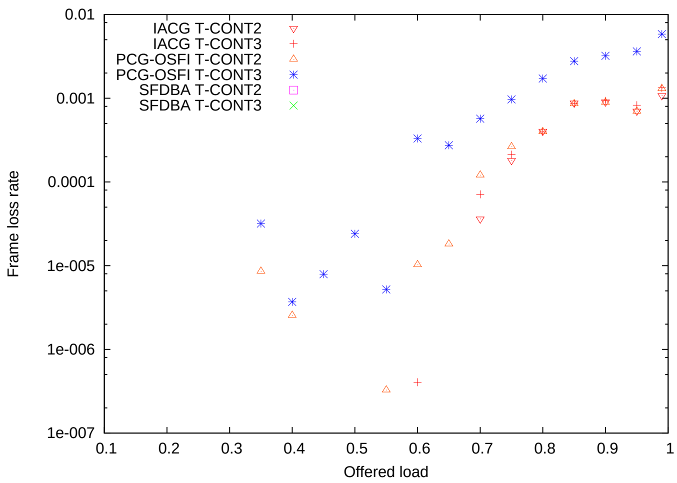

Fig. 9. Frame loss rates of T-CONT types 2 and 3

expired. It means that the unused bandwidth of the queue q_{ij} cannot be utilized until its down counter T_{ij} has expired. In contrast, since SFDBA uses a common available byte counter, an unused bandwidth of a queue can be immediately used by another queues. This difference makes the proposed SFDBA algorithm works better than the EBU algorithm.

Figs. 12 and 13 depict the frame delay variance of each algorithm. Figs. 12 and 13 show that the proposed SFDBA is better than the EBU method in frame delay variance. Fig. 14 illustrates the frame loss rate of each algorithm for T-CONT type 4. The frame loss rates of T-CONT types 2 and 3 are zero in both algorithm thanks to the utilization of the unused bandwidth and the additional polling mechanism. The frame loss rates of both algorithms are comparable to each other for T-CONT type 4 as we can see in Fig. 14.

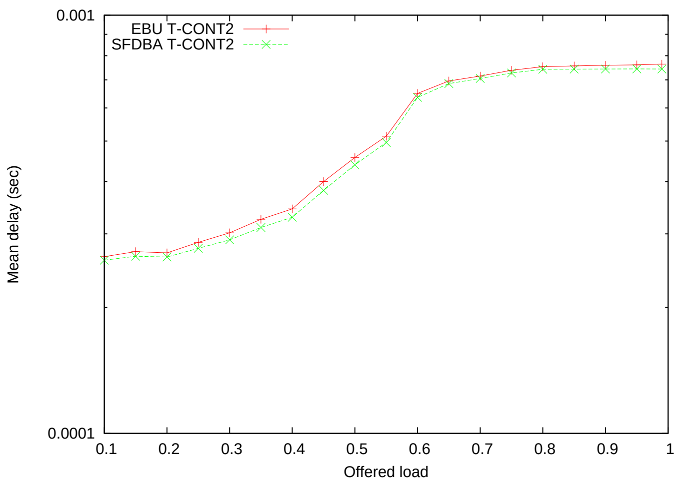

Fig. 10. Mean delay of T-CONT type 2

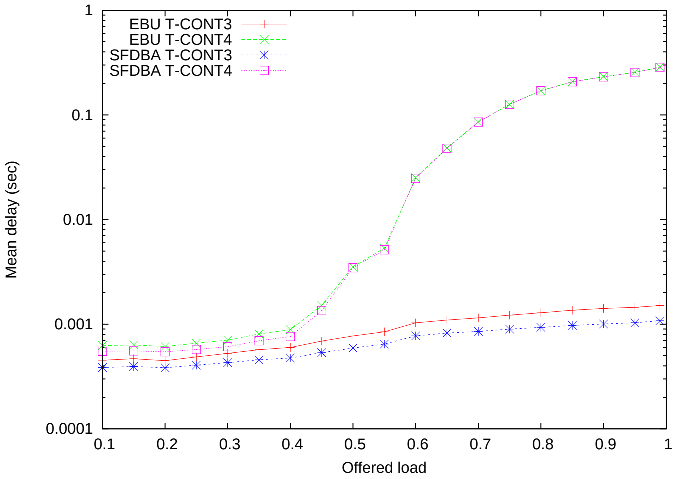

Fig. 11. Mean delay of T-CONT types 3 and 4

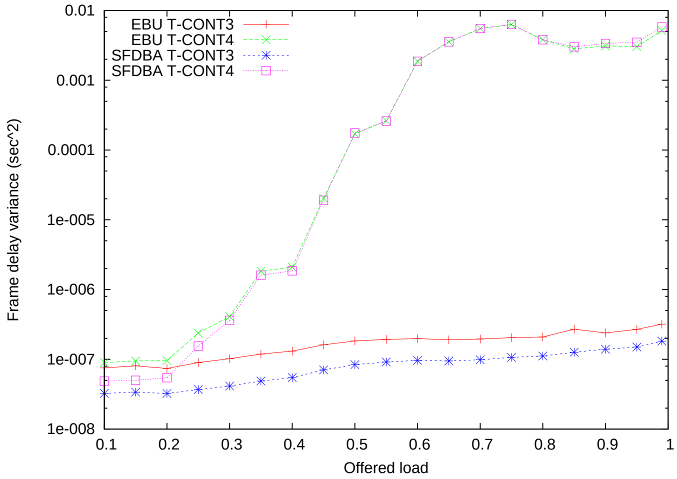

Fig. 13. Frame delay variance of T-CONT types 3 and 4

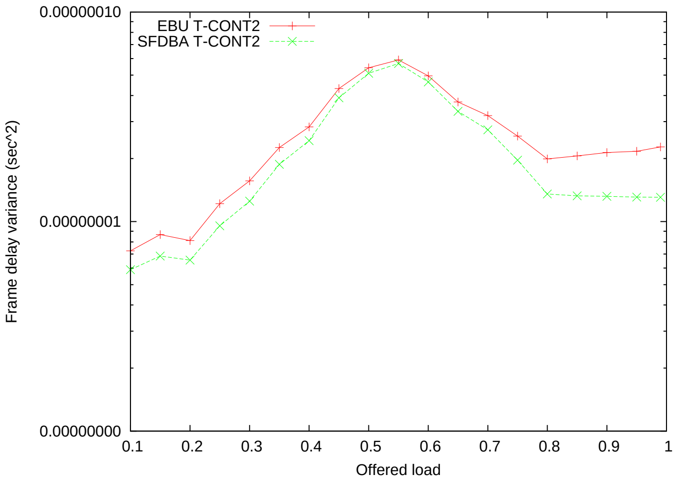

Fig. 12. Frame delay variance of T-CONT type 2

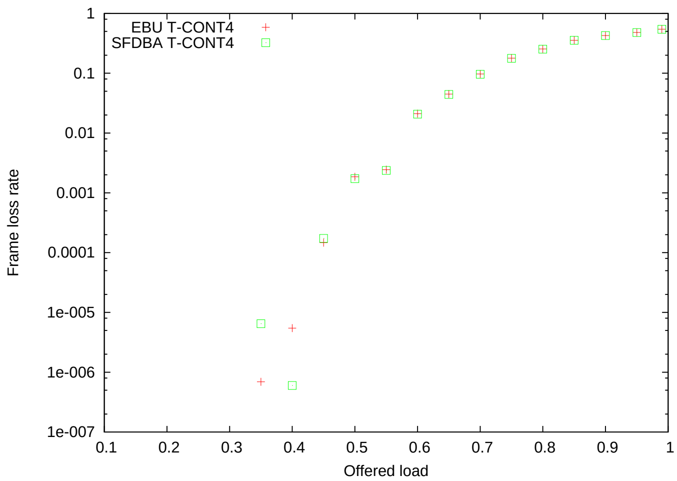

Fig. 14. Frame loss rate of T-CONT type 4

V. Conclusions

We have proposed the SFDBA algorithm for the dynamic bandwidth allocation of XGPON by modifying the IACG algorithm. In IACG, a queue has a down counter to count its service interval and has an available byte counter to control its service amount. Because of the individual counters, a queue can not utilize the unused bandwidth of another queues in IACG. In order to solve the utilization problem of IACG, in SFDBA, all queues of a same T-CONT type use a common down counter and a common available byte counter. Thanks to the common counters, a queue can utilize the unused bandwidth of another queues. Using simulations, we have shown SFDBA outperforms existing algorithms in mean delay, frame delay variance and frame loss rate.

References

[1] ITU-T Rec. G.987.3 Rev.2, "10-Gigabit-capable passive optical networks (XG-PON): Transmission convergence (TC) specifications," 2010.

[2] ITU-T Rec. G.987.1, "10 Gigabit-capable passive optical network (XG-PON): General requirements," 2010.

[3] ITU-T Rec. G.984.3, "Gigabit-capable passive optical networks (G-PON): Transmission convergence layer specification," 2008.

[4] ITU-T Rec. G.984.1, "Gigabit-capable passive optical networks (G-PON): General characteristics," 2008.

[5] J. D. Angelopoulos, H. C. Leligou, and T. Argyriou, "Prioritized multiplexing of traffic accessing an FSAN-compliant GPON," Third IFIP-TC6 Networking Conference, Arthens, Greece, 2004.

[6] H. C. Leligou, Ch. Linardakis, K. Kanonakis, J. D. Angelopoulos, and Th. Orphanoudakis, "Efficient medium arbitration of FSAN-compliant GPONs," Int. J. Comm. Syst., vol. 19, no. 5, pp. 603-617, June 2006.

[7] J. D. Angelopoulos, T. Argyriou, S. Zontos, and T. Van Caenegem, "Efficient transport of packets with QoS in an FSAN-aligned GPON," IEEE Comm. Mag., vol. 42, no. 2, pp. 92-98, Feb. 2004.

[8] M.-S. Han, H. Yoo, B.-Y. Yoon, B. Kim, and J.-S. Koh, "Efficient dynamic bandwidth allocation for FSAN-Compliant GPON," OSA J. Opt. Netw., vol. 7, no. 8, pp. 783-795, Jul. 2008.

[9] M.-S. Han, H. Yoo, and D.-S. Lee, "Development of Efficient Dynamic

Bandwidth Allocation Algorithm for XGPON," ETRI Journal, vol. 35, no. 1, pp. 18-26. 2013.

[10] K. Kanonakis and I. Tomkos, "Offset-Based Scheduling With Flexible Intervals for Evolving GPON Networks," Journal of Lightwave Technology, vol. 27, no. 15, pp. 3259-3268, Aug. 2009.

[11] C. H. Chang, N. M. Alvarez, P. Kourtessis, R. M. Lorenzo, and J. M. Senior, "Full-Service MAC Protocol for Metro-Reach GPONs," Journal of Lightwave Technology, vol. 28, no. 7, pp. 1016-1022, April 2010.

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

[13] A. S. Tanenbaum, "Computer Networks, Fourth Edition," Prentice Hall PTR, 2003,

Man Soo Han received the B.S., M.S. and Ph.D. degrees in electrical engineering from Korea Advanced Institute of Science and Technology (KAIST), Korea in 1992, 1994 and 1999 respectively. He was a senior researcher of the Electronics and Telecommunications Research Institute (ETRI) Daejon, Korea from 1999 to 2003. He is an associate professor in the Department of Information and Communications Engineering at Mokpo National University, Korea. His research interest includes scheduling in high speed networks, passive optical networks and wireless networks. He is a member of IEEE, OSA, IEICE, and KICS.