Investigation of the DBA Algorithm Design Space for EPONs

Michael P. McGarry, *Member, IEEE*, and Martin Reisslein, *Senior Member, IEEE*

***Abstract*—The implications of the main components of dynamic bandwidth allocation (DBA) algorithms in ethernet passive optical networks, namely grant scheduling framework, grant sizing, and grant scheduling, have to date been examined in isolation. In contrast, we conduct a comprehensive throughput-delay comparison study of the three main DBA components; whereby, for each of the DBA components, we consider a range of common mechanisms. Our comparison study considers a number of novel combinations of mechanisms for the individual DBA algorithms, such as the double-phase polling (DPP) scheduling framework combined with limited with excess distribution grant sizing, and shortest propagation delay (SPD) first scheduling. We find that this (DPP, Limited with excess, SPD) combination in conjunction with a novel excess sharing mechanism outperforms previously studied DBA algorithms.**

***Index Terms*—Ethernet passive optical network (EPON), grant scheduling, grant sizing, packet delay, propagation delay.**

I. INTRODUCTION

Dynamic bandwidth allocation (DBA) algorithms [1]–[7] for the upstream channel from the optical network units (ONUs) to the optical line terminal (OLT) of ethernet passive optical networks (EPONs) have received considerable interest from the research community over the past several years. During this period of time, DBA algorithms have been reduced to three subproblems: 1) grant scheduling framework, 2) grant sizing, and 3) grant scheduling [2].

During this reductionist period, these subproblems along with the problem of packet scheduling inside ONUs (referred to as intra-ONU scheduling [8]) have been studied almost independently. In most cases, the solutions to two of the subproblems were fixed while one was varied. For instance, a few studies [9]–[13] have focused on the overall scheduling framework that triggers bandwidth allocation decisions. Many studies have focused on sizing the bandwidth grants for the individual ONUs [1], [14]–[22], while others have primarily focused on scheduling the sized grants [23]–[32].

With this study, we wish to usher in a new synergistic era for DBA algorithm research by exploring the three dimensional design space of DBA algorithms whereby each of the three subproblems is a dimension in the design space. This is the first study of its kind. Several of the combinations of algorithms in the design space that we explore are novel DBA algorithms. Importantly, some of these novel DBA algorithms outperform all previously proposed DBA algorithms. For instance, combining the double-phase polling (DPP) scheduling framework [9], [10] with limited with excess distribution grant sizing [14], [33] and shortest propagation delay (SPD) first scheduling [32] achieves better throughput delay performance than previously examined DBA algorithms. Further building on this (DPP, Limited with excess, SPD) DBA algorithm, we propose a novel excess bandwidth sharing strategy that further significantly improves throughput-delay performance.

We focus on DBA mechanisms with a single thread in this paper, i.e., an ONU is polled once per cycle. Polling with multiple threads, which has recently been proposed for long-reach EPONs [34], may alleviate long propagation delays, but introduces additional guard times and complexities for adaptively controlling the appropriate spacing between the different threads. In contrast, in this study, we examine low-complexity single-thread polling mechanisms.

Throughout, we consider single-channel EPONs with different propagation distance ranges so that our results provide insights both for standard distance EPONs as well as long-reach EPONs [34], [35].

This paper is organized as follows. In Section II, we describe the three dimensional DBA algorithm design space. In Section III, we describe our novel proposal for excess bandwidth distribution for the DPP scheduling framework. In Section IV, we present an extensive delay and maximum achievable channel utilization performance analysis of a wide range of DBA algorithms derived from the three dimensional design space, including several novel combinations. Finally, we summarize our conclusions in Section V.

II. DBA ALGORITHM DESIGN SPACE

We classify and identify DBA algorithms along the three design dimensions of:

1) grant scheduling framework, which is characterized by the event triggering a bandwidth allocation,
2) grant sizing policy, which determines the size (duration) of the upstream transmission window allocated to an ONU, and
3) grant scheduling policy, which determines the temporal order of several simultaneously scheduled transmission windows.

Manuscript received December 01, 2011; revised April 13, 2012; accepted April 16, 2012. Date of publication May 03, 2012; date of current version May 16, 2012. A preliminary version of this paper appeared with the title "Impact of DBA components on performance" in the International Conference on Computer Communications and Networks, Aug. 2011.

M. McGarry is with the Department of Electrical and Computer Engineering, University of Texas at El Paso, El Paso, TX 79902 USA (e-mail: mpmcgarry@utep.edu).

M. Reisslein is with the School of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe, AZ 85287-5706 USA (e-mail: reisslein@asu.edu).

Digital Object Identifier 10.1109/JLT.2012.2196023

Thus, we identify a DBA algorithm by the following triple: (grant scheduling framework, grant sizing policy, grant scheduling policy).

A. *Grant Scheduling Framework*

The grant scheduling framework determines when the OLT makes access decisions and send transmission grants to ONUs. We can differentiate the scheduling frameworks according to the event during an arbitrary granting cycle that triggers the production of a granted transmission window schedule for the next granting cycle.

1) Online—triggered by the receipt of a REPORT from any ONU, only that ONU is scheduled.
2) Offline—triggered by the receipt of REPORTs from all ONUs, all ONUs are scheduled.
3) ONU Load Status (OLS) [14]—triggered by the receipt of a REPORT from any ONU, if the REPORT is less than a guaranteed minimum the ONU is scheduled immediately otherwise it is scheduled when REPORTs from all ONUs have been received.
4) Double Phase Polling (DPP) [9], [10]—ONUs are partitioned into two independent groups. Within each group, a schedule is triggered by the receipt of REPORTs from all ONUs in the group, all ONUs in the group are scheduled.

More precise analytical definitions of these frameworks can be found in the Appendix.

B. *Grant Sizing Policy*

The grant sizing policy determines the size (duration) of ONU i's granted transmission window $G(i,n)$ during the next granting cycle n. The following general grant sizing policies have been proposed and evaluated in the literature [1], [2].

1) Fixed, the granted transmission window is a fixed size
2) Gated, the granted transmission window size equals the queue depth $R(i,n)$ reported at the end of the upstream transmission of ONU i in cycle $n-1$
3) Limited, the granted transmission window size equals the previously reported queue depth not to exceed a fixed limit $L(i)$

1) Limited With Excess Distribution: Limited grant sizing can be augmented with a technique called excess distribution [14], [33]. With this technique, in each cycle n, ONUs are divided into two sets: underloaded ONUs $i \in \mathcal{U}(n)$ with $R(i,n) \leq L(i)$, and overloaded ONUs $i \in \mathcal{O}(n)$ with $R(i,n) > L(i)$. The underloaded ONUs receive a grant to satisfy their REPORT

$$G(i,n) = R(i,n), \forall\, i \in \mathcal{U}(n) \tag{1}$$

and contribute their excess bandwidth to a pool of excess bandwidth credits for the granting cycle

$$E(n) = \sum_{i \in \mathcal{U}(n)} L(i) - R(i,n). \tag{2}$$

The overloaded ONUs receive a grant that will include some portion of the excess bandwidth accumulated from the underloaded ONUs. Let $e(i,n)$ be the amount of excess bandwidth distributed to ONU i in granting cycle n; with equitable excess division [14], [33]

$$e(i,n) = \frac{E(n)}{|\mathcal{O}(n)|}, \forall\, i \in \mathcal{O}(n). \tag{3}$$

When controlled excess allocation [33] is used, ONUs receive a grant no larger than their reported queue depth, i.e.,

$$G(i,n) = \min\{L(i) + e(i,n),\ R(i,n)\}, \forall\, i \in \mathcal{O}(n) \tag{4}$$

whereby, the unused excess bandwidth occurring when $R(i,n) < L(i) + e(i,n)$ is ignored. In contrast, iterative excess allocation [33] accumulates the difference $L(i) + e(i,n) - R(i,n)$ into $E(n)$ for distribution to other overloaded ONUs.

As indicated in (2), $E(n)$ must be computed after the OLT receives reported queue depths $R(i,n)$ from all ONUs at the end of their transmission windows during granting cycle $n-1$. When using an online scheduling framework, the grant size of each ONU for the next granting cycle n is determined without consideration of the reported queue depths of other ONUs. As a result, the computation of $E(n)$ is not possible, and therefore, excess bandwidth distribution cannot be supported when using an online scheduling framework unless an alternative method of accumulating excess bandwidth credits is devised. This possibility was investigated in [18], but the proposed method of accumulating excess bandwidth credits could lead to very large granting cycle lengths that will degrade performance.

When using an offline scheduling framework, the transmission grants for all ONUs are determined after the OLT receives $R(i,n),\ \forall\ i$, readily supporting the computation of $E(n)$. However, an offline scheduling framework does not interleave polling across granting cycles and will result in an idle upstream channel time between granting cycles.

An OLS scheduling framework exploits the fact [see (1)], that the grant size for an underloaded ONU only depends on that ONU's reported queue depth [36]. This permits, if there are underloaded ONUs, some interleaved polling across granting cycles that will fill in the idle gap between granting cycles.

C. *Grant Scheduling Policy*

The grant scheduling policy determines how multiple ONU transmission windows are ordered during a granting cycle. When the OLT uses an online scheduling framework, a grant scheduling policy cannot be used, since only one ONU is scheduled at a time. When the OLT uses an OLS scheduling framework, a scheduling policy only applies to the overloaded ONUs. The following are some grant scheduling policies that have been proposed and evaluated in the literature.

1) Shortest grant or shortest processing time (SPT) first [37] orders the ONUs in ascending order by grant size
2) Largest number of frames (LNF) first [11] orders the ONUs in descending order by number of queued frames
3) Shortest propagation delay (SPD) first [32] orders the ONUs in ascending order by round-trip propagation delay.

III. EXCESS CREDIT ACCUMULATION FOR THE DPP SCHEDULING FRAMEWORK

Excess bandwidth distribution has not yet been investigated for use with the DPP scheduling framework. Given that DPP creates two independent groups of ONUs that are polled separately, a natural proposal is to accumulate, divide, and allocate excess bandwidth separately within each group.

Let $E(k,n)$ be the total excess bandwidth credits from polling group k during granting cycle n, $\mathcal{U}(k,n)$ be the set of underloaded ONUs within polling group k during granting cycle n, and $\mathcal{O}(k,n)$ be the set of overloaded ONUs within polling group k during granting cycle n. Then, within each group k

$$E(k,n) = \sum_{i \in \mathcal{U}(k,n)} L(i) - R(i,n). \tag{5}$$

This method restricts the number of ONUs that share bandwidth to half the total number of ONUs, thereby limiting statistical multiplexing.

To mitigate this limitation, we propose to share excess bandwidth credits between the two groups. If unused credits are simply forwarded from one group to the next, the cycle length will become unbounded. To prevent this, an excess credit sharing mechanism must limit the forwarding of excess credits. Our mechanism limits forwarding to the immediately next polling group, e.g., polling group 2 in cycle $n-1$ can forward credits to polling group 1 in cycle n, but these credits cannot be forwarded on to polling group 2 in cycle n.

Formally, let $S(1,n)$ denote the excess credits forwarded by group 1 in cycle n to group 2 in cycle n, and let $S(2,n)$ denote the excess credits forwarded by group 2 in cycle n to group 1 in cycle $n+1$. The total excess credits $E^*(k,n)$ for a group k are those accumulated within that group $E(k,n)$ (5) plus those forwarded from the preceding group. Specifically, the total excess credits for groups 1 and 2 during granting cycle n are

$$E^*(1,n) = E(1,n) + S(2,n-1) \tag{6}$$

$$E^*(2,n) = E(2,n) + S(1,n) \tag{7}$$

and these total credits $E^*(k,n)$ are considered in the excess division and allocation according to (3) and (4) to obtain the grant sizes $G(i,n)$ for the overloaded ONUs $i \in \mathcal{O}(k,n)$.

Our mechanism prevents a polling group from forwarding credits that the group received from the preceding group. Only excess credits $E(k,n)$ (5) accumulated within a polling group k within a given cycle n can be forwarded to the next group. Therefore,

$$S(k,n) = \min\left\{E^*(k,n) - \sum_{i \in \mathcal{O}(k,n)} G(i,n) - L(i), E(k,n)\right\}. \tag{8}$$

This sharing mechanism bounds the "lifetime" of excess credits to the group in which they are accumulated and the next group (i.e., a full granting cycle). As a result, the cycle length is bounded. Specifically, the new upper bound on the cycle length is 50 % larger than with no sharing. To reduce this upper bound increase, we can parameterize this sharing approach to limit the number of forwarded credits. In our experiments, we refer to this mechanism for excess bandwidth credit accumulation as Excess:Share.

We briefly contrast the Excess:Share mechanism that we propose for DPP [9] with the bandwidth sharing mechanisms in multithread polling [34]. DPP splits the ONUs into two groups and thus restricts the sharing of excess bandwidth to ONUs within a given group. The proposed Excess:Share mechanism overcomes this restriction and permits all ONUs to share excess bandwidth over the time period of one full polling (granting) cycle. In contrast, multithread polling temporally interleaves multiple polling processes (threads), whereby each thread polls all ONUs. Thus, within a given thread, excess bandwidth can be shared by all ONUs, i.e., multithread polling does not impose restrictions on bandwidth sharing among ONUs within a given thread. However, multithread polling introduces the challenge of sharing bandwidth among the multiple interleaved threads, which requires adaptive thread tuning mechanisms [34].

IV. PERFORMANCE ANALYSIS

A. Simulation Setup

We conducted a set of simulation experiments to compare the packet delay and maximum achievable channel utilization of the following DBA algorithms in the three dimensional design space:

1) (Online, Limited)
2) ({Offline, DPP}, Limited, {LNF, SPD})
3) ({Offline, OLS, DPP}, Excess, {LNF, SPD})
4) (DPP, Excess:Share, SPD)

We use an EPON simulator that we have developed using the CSIM discrete event simulation library [38]. We simulated an EPON with a channel capacity, $C = 1$ Gb/s and $M = 32$ ONUs. We varied the maximum propagation delay to represent three different EPON reaches: 1 to 10 km (6.67 to 50 μs), 1 to 50 km (6.67 to 250 μs), and 1 to 100 km (6.67 to 500 μs) (in [32], we illustrate the feasibility of these ranges in practical EPON architectures). A quad modal packet size distribution was used for all simulation experiments: 60% 64 bytes, 4% 300 bytes, 11% 580 bytes, and 25% 1518 bytes. We set the guard time, $t_g = 1$ μs, and $L(i) = 7688$ bytes (i.e., 61.5 μs), $\forall$ i, i.e., $\sum_{i=1}^{32}[L(i) + t_g] = 2$ ms. Initially, each ONU is assigned a grant size to accommodate only the REPORT message.

B. DBA Model and Notation

For the interpretation of the simulation results, we employ the DBA model notation in Table I. The critical quantity for the packet delay interpretation is the channel idle time $\Delta(j,n)$ which based on the definitions in Table I is

$$\Delta(j,n) = \begin{cases} \alpha(j,n) - \beta(M,n-1), & \text{for} \quad j = 1 \\ \alpha(j,n) - \beta(j-1,n), & \text{for} \quad j = 2,\ldots,M. \end{cases} \tag{9}$$

Toward the evaluation of $\Delta(j,n)$, we examine the start time $\alpha(j,n)$ of the arrival of the upstream transmission of the jth

TABLE I
NOTATION OF THE DBA MODEL

M	Total number of ONUs, which are numbered $i,\ i = 1, \ldots, M$
j	ONU index ordered by upstream transmission position for given cycle n, i.e., ONU j has j^{th} upstream grant in cycle n
$o(i, n)$	Operator that returns the upstream transmission order of ONU i during granting cycle n
$o^{-1}(j, n)$	Operator that returns the number i of the jth ONU to transmit upstream during granting cycle n
$\gamma(j, n)$	Time instant when OLT makes scheduling decision for granted transmission window of jth ONU in cycle n
$\tau(i)$	Propagation delay (symmetric) between the OLT and ONU i
t_g	guard time between ONU upstream transmissions
t_G	transmission time of a GATE message
$T(j, n)$	GATE signaling delay: Time duration from instant of OLT scheduling decision to end of GATE transmission to jth ONU in cycle n plus round-trip propagation delay
$G(j, n)$	Duration of granted transmission window of jth ONU in cycle n
$\alpha(j, n)$	Time instant when upstream transmission of jth ONU in cycle n begins to arrive at OLT
$\beta(j, n)$	Time instant when end of upstream transmission of jth ONU in cycle n arrives at the OLT, $\beta(j, n) = \alpha(j, n) + G(j, n)$
$\eta(j, n)$	Time instant when upstream channel at OLT becomes free prior to arrival of upstream transmission of jth ONU in cycle n, i.e., a guard time after end of arrival of preceding upstream transmission, $\eta(j, n) = \beta(M, n-1) + t_g$ for $j = 1$ and $\eta(j, n) = \beta(j-1, n) + t_g$ for $j = 2, \ldots, M$
$\Delta(j, n)$	Channel idle time preceding the arrival of upstream transmission of jth ONU in cycle n at OLT

TABLE II
AVERAGE PACKET DELAY VALUES (IN MILLISECONDS), EXPRESSED AS (CI LOWER BOUND, AVERAGE, CI UPPER BOUND). (a) 50 μS MAX. PROP. DELAY, I.E., UP TO 10 KM AND (B) 500 μS MAX. PROP. DELAY, I.E., UP TO 100 KM

(a)

Load (in Gbps)	0.5	0.7
(Online, Limited)	(3.18, 3.24, 3.30)	(4.30, 5.85, 7.41)
(Offline, Limited, LNF)	(5.78, 5.91, 6.04)	(8.64, 11.25, 13.87)
(Offline, Limited, SPD)	(3.82, 3.89, 3.95)	(6.68, 6.72, 6.77)
(DPP, Limited, LNF)	(3.57, 3.63, 3.70)	(4.44, 6.06, 7.67)
(DPP, Limited, SPD)	(3.25, 3.31, 3.37)	(4.32, 5.87, 7.43)
(Offline, Excess, LNF)	(1.54, 1.57, 1.60)	(2.34, 3.34, 4.35)
(Offline, Excess, SPD)	(1.33, 1.36, 1.38)	(2.52, 2.75, 2.99)
(OLS, Excess, LNF)	(1.27, 1.29, 1.32)	(2.41, 2.64, 2.86)
(OLS, Excess, SPD)	(1.27, 1.29, 1.32)	(2.41, 2.64, 2.86)
(DPP, Excess, LNF)	(1.30, 1.32, 1.35)	(2.43, 2.65, 2.88)
(DPP, Excess, SPD)	(1.30, 1.33, 1.35)	(2.43, 2.66, 2.89)
(DPP, Excess:Share, SPD)	(1.25, 1.27, 1.30)	(2.34, 2.56, 2.78)

(b)

Load (in Gbps)	0.5	0.7
(Online, Limited)	(18.27, 22.53, 26.80)	(17.34, 24.88, 32.43)
(Offline, Limited, LNF)	(44.94, 61.01, 77.08)	(∞)
(Offline, Limited, SPD)	(24.12, 24.24, 24.36)	(19.69, 27.82, 35.96)
(DPP, Limited, LNF)	(19.94, 30.33, 40.73)	(35.79, 47.59, 59.39)
(DPP, Limited, SPD)	(23.16, 23.28, 23.39)	(18.07, 25.84, 33.61)
(Offline, Excess, LNF)	(4.49, 7.39, 10.29)	(∞)
(Offline, Excess, SPD)	(3.21, 3.25, 3.30)	(3.70, 4.81, 5.91)
(OLS, Excess, LNF)	(3.75, 3.80, 3.86)	(6.93, 7.05, 7.18)
(OLS, Excess, SPD)	(3.38, 3.42, 3.46)	(5.19, 5.50, 5.81)
(DPP, Excess, LNF)	(3.99, 4.07, 4.16)	(6.15, 6.61, 7.07)
(DPP, Excess, SPD)	(3.17, 3.22, 3.26)	(3.06, 4.39, 5.72)
(DPP, Excess:Share, SPD)	(2.75, 2.78, 2.81)	(2.72, 3.93, 5.15)

ONU in granting cycle n. At the earliest, this upstream transmission can begin to arrive a guard time after the arrival of the end of the preceding upstream transmission, i.e., with the notation in Table I, no earlier than $\eta(j, n)$. Furthermore, this upstream transmission of the jth ONU in cycle n can arrive no earlier than the time instant $\gamma(j, n)$ when the scheduling decision for this upstream transmission was made, plus the time duration $T(j, n)$ for GATE transmission and subsequent round trip propagation (of GATE from OLT to ONU and upstream transmission from ONU to OLT). Based on these two constraints

$$\alpha(j, n) = \max\{\eta(j, n),\ \gamma(j, n) + T(j, n)\}. \tag{10}$$

Inserting (10) in (9) gives

$$\Delta(j, n) = \max\{t_g,\ \Delta_s(j, n)\} \tag{11}$$

whereby, we defined for notational convenience the schedule notification delay

$$\Delta_s(j, n) = \begin{cases} \gamma(j, n) + T(j, n) - \beta(M, n-1), & \text{for} \quad j = 1 \\ \gamma(j, n) + T(j, n) - \beta(j-1, n), & \text{for} \quad j > 1. \end{cases} \tag{12}$$

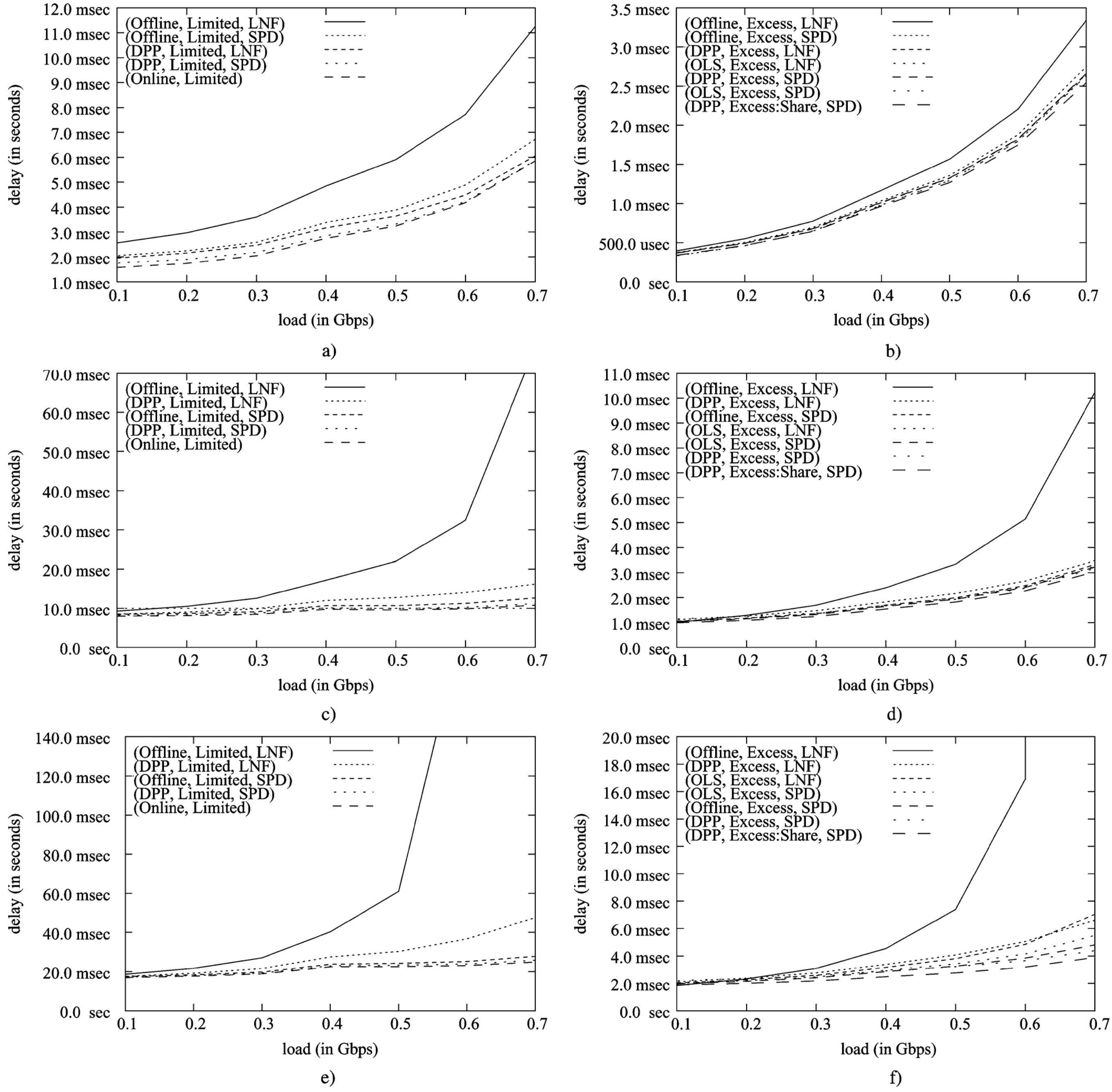
Note that if the constraint $\eta(j, n)$ determines the arrival start time $\alpha(j, n)$, then the channel idle time is the guard time, i.e., $\Delta(j, n) = t_g$. On the other hand, if the constraint $\gamma(j, n) + T(j, n)$ determines $\alpha(j, n)$, then the channel idle time depends on the GATE signaling delay $T(j, n)$, which starts to elapse at the instant of the scheduling decision $\gamma(j, n)$; however, the channel is occupied by the upstream transmission of the preceding ONU up to time instant $\beta(M, n-1)$ for the first ONU $(j = 1)$ in cycle n and $\beta(j-1, n)$ for the subsequent ONUs $(j = 2, \ldots, M)$, reducing the idle time. Expressions for $\gamma(j, n)$ and $T(j, n)$ for the different scheduling frameworks are provided in the Appendix and are employed for the interpretation of the packet delay in Section IV-C2.

C. Packet Delay

Fig. 1 shows the average packet delay values for the 12 different DBA algorithms for the three different EPON reach configurations. DBA algorithms with limited grant sizing are in the left column of the figure and those with limited with excess distribution grant sizing are in the right column. Table II records the delay values for two specific load points.

We first summarize and then elaborate on our primary observations from the experimental data in Fig. 1 and Table II.

1) Limited with excess distribution grant sizing results in a very significant reduction in average packet delay compared to limited grant sizing. The difference increases with increasing maximum propagation delay. Our new method of sharing excess credits among polling groups within the DPP scheduling framework reduces average packet delay even further compared to not sharing the excess credits.
2) The SPD scheduling policy results in much lower average queueing delay than the LNF scheduling policy; especially with the offline scheduling framework. This difference increases significantly with increasing maximum propagation delay.
3) The DPP scheduling framework results in an average queueing delay very close to the online scheduling framework.



a) b) c) d) e) f)

Fig. 1. Average packet delay for different combinations of (grant scheduling framework, grant sizing policy, grant scheduling policy) for three different propagation delay ranges. (a) Limited, 50 μs max. prop. delay (i.e., up to 10 km). (b) Excess, 50 μs maximum propagation delay (i.e., up to 10 km). (c) Limited, 250 μs maximum propagation delay (i.e., up to 50 km). (d) Excess, 250 μs maximum propagation delay (i.e., up to 50 km). (e) Limited, 500 μs maximum propagation delay (i.e., up to 100 km). (f) Excess, 500 μs maximum propagation delay (i.e., up to 100 km).

4) For long-reach EPON, the packet delay difference between the worst algorithm and the best algorithm is quite dramatic. This dramatic difference illustrates the significant impact DBA algorithms can have on packet delay performance.

1) Observation 1: Limited with excess distribution grant sizing provides lower average queueing delay because grant sizes of limited with excess distribution for overloaded ONUs as given by (4) are larger than or equal to the grant sizes $G(i, n) = \min\{R(i, n), L(i)\}$ with limited grant sizing. Increased grant sizes means that more queued packets can be dequeued and transmitted during the next granting cycle after they are REPORTed. As a result, average queueing delay are lower when using limited with excess distribution grant sizing.

Although limited with excess distribution grant sizing allows for large grant sizes to overloaded ONUs, the cycle length upper bound is still approximately 2 ms (the true cycle length also accommodates the idle periods between grants). For our excess credit sharing mechanism, it is 50% larger, 3 ms. Our simulation experiments confirmed that the maximum cycle length was 2 ms for limited with excess distribution grant sizing and between 2.8 and 3 ms for our new excess credit sharing mechanism, Excess:Share.

From the experimental data in Table II, we can quantify the differences in average delay among the various DBA algorithms. For a load of 0.7 Gb/s and an EPON reach up to 100 km (DPP, Limited, SPD) yields an average queueing delay of 25.84 ms, whereas our novel (DPP, Excess:Share, SPD) yields an average queueing delay of 3.93 ms, a dramatic 85% decrease. For the same presented load, (Offline, Excess, SPD) yields an average queueing delay of 4.81 ms; (DPP, Excess:Share, SPD) provides a 19% decrease.

Also, for the 0.7 Gb/s load, (DPP, Excess, SPD) yields an average queueing delay of 4.39 ms; (DPP, Excess:Share, SPD) provides an 11% decrease. With a presented load of 0.5 Gb/s, the data indicate that the average delay difference between (DPP, Excess, SPD) and (DPP, Excess:Share, SPD) is statistically significant. As a result, there is a significant benefit using our excess credit sharing mechanism for the DPP scheduling framework.

2) Observation 2: Since the SPD scheduling policy sorts the ONUs by propagation delay, the time duration between transmission grants for each ONU will be shortened. As a result, the queueing delay at each ONU will be reduced. With the DBA model from Section IV-B, we illustrate this analytically.

For the offline scheduling framework, we obtain by inserting (21) and (22) in (12)

$$\Delta_s(j,n) = \begin{cases} t_G + 2\tau(o(1,n)), & \text{for } j = 1 \\ \beta(M,n-1) + jt_G & \\ \quad +2\tau(o(j,n)) - \beta(j-1,n), & \text{for } j > 1. \end{cases} \tag{13}$$

Recall from (10) that $\Delta_s(j,n)$ governs the channel idle time $\Delta(j,n)$ if $\alpha(j,n) = \gamma(j,n) + T(j,n)$; with (21) and (22), we thus obtain for $j > 1$

$$\Delta_s(j,n) = t_G - G(j-1,n) \\ +2\tau(o(j,n)) - 2\tau(o(j-1,n)). \tag{14}$$

By ordering ONUs in ascending order by their propagation delay, SPD minimizes the propagation delay of the first ONU $\tau(o(1,n))$, and minimizes the propagation delay difference between subsequent ONUs $\tau(o(j,n)) - \tau(o(j-1,n))$. As a result, SPD reduces Δ further than other scheduling methods. For a presented load of 0.5 Gb/s and maximum propagation delay of 500 μs, (Offline, Limited, LNF) provides an average packet delay of 61.01 ms, while (Offline, Limited, SPD) provides an average packet delay of 24.24 ms, a very significant 60% reduction.

3) Observation 3: By analyzing $\Delta(j,n)$ we can illustrate why the DPP scheduling framework results in average packet delay that is similar to the online scheduling framework. For the online scheduling framework, we obtain by inserting (19) and (20) in (12)

$$\Delta_s(j,n) = \begin{cases} [\beta(1,n-1) - \beta(M,n-1)] & \\ \quad +t_G + 2\tau(o(j,n)), & \text{for } j = 1 \\ [\beta(j,n-1) - \beta(j-1,n)] & \\ \quad +t_G + 2\tau(o(j,n)), & \text{for } j > 1. \end{cases} \tag{15}$$

For the DPP scheduling framework, we obtain for the first ONU in the first polling group ($j = 1$) from inserting (25) and (26) in (12)

$$\Delta_s(1,n) = [\beta(M/2,n-1) - \beta(M,n-1)] \\ +t_G + 2\tau(o(1,n)). \tag{16}$$

For the subsequent ONUs in the first polling group ($j = 2, \ldots, M/2$), the scheduling dynamics are analogous to the ONUs with $j > 1$ in the offline scheduling framework, and we consequently obtain the same delay expression as in (14). For the first ONU in the second polling group ($j = M/2 + 1$), we obtain from (25), (26), and (12)

$$\Delta_s(M/2+1,n) = [\beta(M,n-1) - \beta(M/2,n)] \\ +t_G + 2\tau(o(M/2+1,n)) \tag{17}$$

while the subsequent ONUs in the second polling group ($j = M/2 + 2, \ldots, M$) experience effectively the offline scheduling framework dynamics, i.e., (14). The first square bracketed term in both (16) and (17) are clearly negative and will diminish the value of the other two terms in both equations.

Comparing the scheduling notification delay Δ_s from (16) and (17) for the DPP scheduling framework with the corresponding delays (15) for the online scheduling framework, we observe that the first bracketed terms in all equations are negative. Therefore, unless the GATE signaling delay exceeds the first bracketed term, the beginning of the upstream transmission α will be determined by the channel available time η, and therefore, Δ will be a mandatory guard time t_g. The experimental data indicate that the DPP scheduling framework is allowing for Δ to be determined by t_g nearly as frequently as the online scheduling framework, resulting in similar values for α. For instance, for a load of 0.5 Gb/s and maximum propagation delay of 500 μs, (Online, Limited) provides an average packet delay of 22.53 ms while (DPP, Limited, SPD) provides an average packet delay of 23.28 ms (which is statistically nonsignificantly larger than the (Online, Limited) delay).

4) Observation 4: Combining the effects of scheduling framework, grant sizing policy, and scheduling policy leads to dramatic average packet delay differences. The grant sizing policy clearly has the strongest impact, but the scheduling framework and scheduling policy also have significant impact. For a load of 0.5 Gb/s and maximum propagation delay of 500 μs, the average packet delay is 61 ms with (Offline, Limited, LNF) and only 2.78 ms with (DPP, Excess:Share, SPD), a reduction by a factor of 22.

D. Maximum Achievable Channel Utilization

Fig. 2 shows the delay values for high loads to determine the point at which the delay becomes asymptotically unstable (i.e., the point at which the maximum achievable channel utilization is reached) for the 12 different DBA algorithms for the three different EPON reach configurations. Those DBA algorithms that used limited grant sizing are in the left column of the figure and those that used limited with excess grant sizing

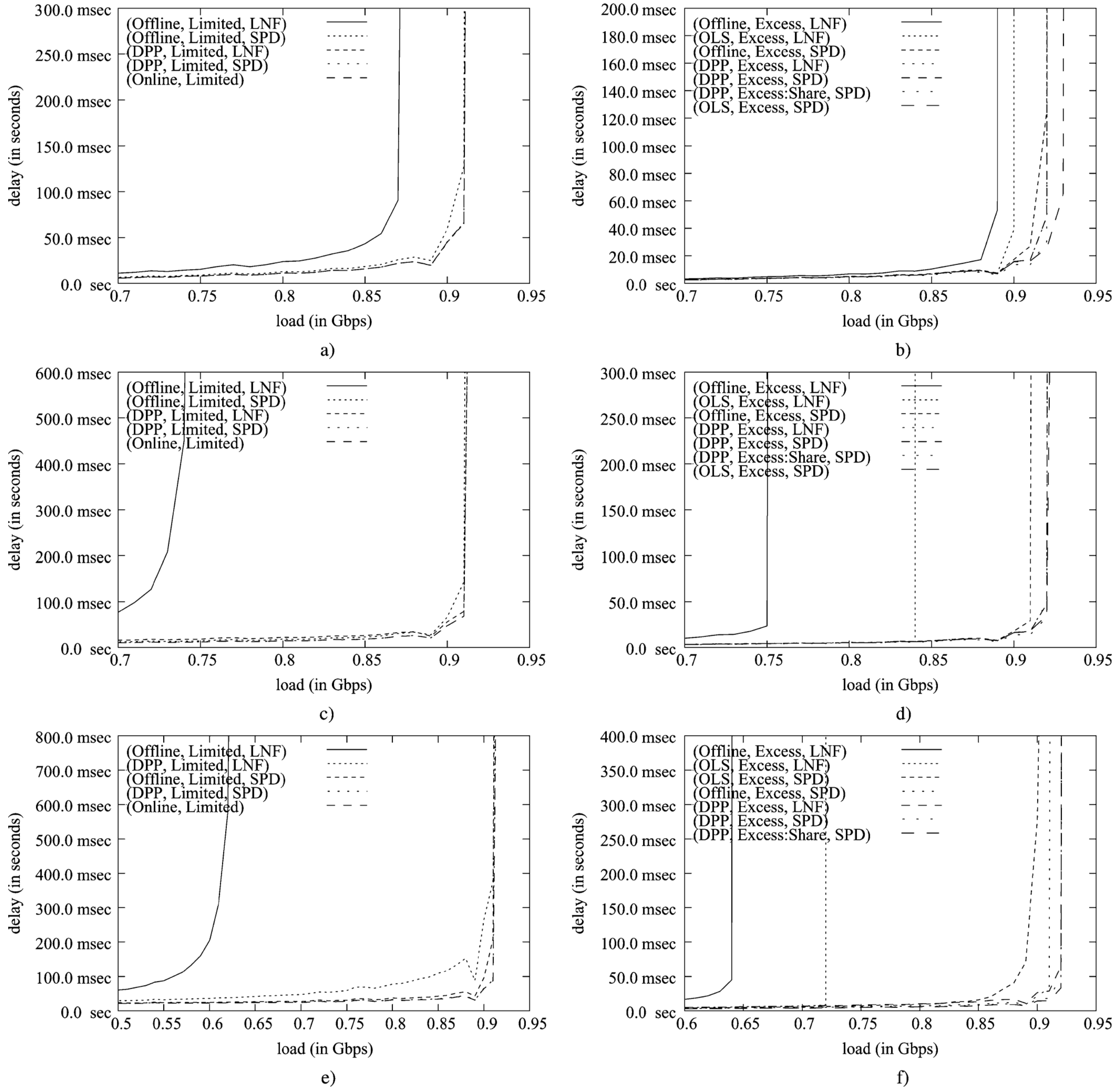


Fig. 2. Stability limit for different combinations of (grant scheduling framework, grant sizing policy, grant scheduling policy) for three different propagation delay ranges. (a) Limited, 50 μs maximum propagation delay (i.e., up to 10 km). (b) Excess, 50 μs maximum propagation delay (i.e., up to 10 km). (c) Limited, 250 μs maximum propagation delay (i.e., up to 50 km). (d) Excess, 250 μs maximum propagation delay (i.e., up to 50 km). (e) Limited, 500 μs maximum propagation delay (i.e., up to 100 km). (f) Excess, 500 μs maximum propagation delay (i.e., up to 100 km).

are in the right column. We summarize and then elaborate on our primary observations from the experimental data visualized in Fig. 2.

1) The SPD scheduling policy results in a higher stability limit than LNF for the offline and OLS scheduling frameworks. The difference increases dramatically with increasing maximum propagation delay.
2) The DPP scheduling framework provides the highest stability limits. Additionally, the scheduling policy loses its impact on the stability limit when used with the DPP scheduling framework.
3) The limited with excess distribution grant sizing scheme provides a stability limit only slightly higher than limited grant sizing. (Offline, Excess, SPD), which is examined here for the first time, provides a stability limit slightly higher than (Online, Limited) and (DPP, {Excess, Excess:Share}, SPD) provides the highest stability limit.

1) Observation 1: As was illustrated in our discussion of Observation 2 in Section IV-C, SPD scheduling by ordering ONUs in ascending order of propagation delay minimizes the value of the propagation delay of the first ONU and the values of propagation delay differences between subsequent ONUs. As a result, SPD scheduling minimizes the values of Δ for the Offline and OLS scheduling frameworks. As a further result, the channel utilization

$$\rho(n) = \frac{\sum_{j=1}^{M} G(j,n)}{\sum_{j=1}^{M} \Delta(j,n) + G(j,n)} \tag{18}$$

is maximized each granting cycle by minimizing Δ. For instance, for a maximum propagation delay of 500 μs, (Offline, Limited, LNF) results in a stability limit of 0.62 Gb/s, while (Offline, Limited, SPD) provides a stability limit of 0.91 Gb/s. This 290 Mb/s difference is nearly a third of the channel capacity. This additional channel utilization is sufficient to provide 10 Mb/s service to 29 additional subscribers.

2) Observation 2: As was illustrated in our discussion of Observation 3 in Section IV-C, with DPP, the idle time Δ is determined by the mandatory guard time t_g nearly as often as with the online scheduling framework. As a result, both scheduling frameworks produce similar channel utilization that will produce the same stability limit. Additionally, the impact of the subsequent propagation delay differences is diminished, resulting in the scheduling policy having an unnoticeable impact on the stability limit. For a maximum propagation delay of 500 μs, (DPP, Limited, SPD), (Online, Limited), and (DPP, Limited, LNF) achieve a stability limit of 0.91 Gb/s.

3) Observation 3: Equation (18) clearly indicates that a grant sizing scheme that produces larger grant sizes will result in higher channel utilization. Equations (1) and (4) clearly illustrate that limited with excess distribution grant sizing will produce grant sizes that are larger than or equal to those produced by limited grant sizing. Therefore, limited with excess distribution grant sizing will result in a channel utilization that is higher than or equal to that of limited grant sizing. However, increasing grant sizes through excess distribution is not as significant as reducing the Δ values through scheduling or enhanced interleaving as with the DPP scheduling framework. For a maximum propagation delay of 500 μs, (Offline, Excess, SPD) results in a stability limit of 0.92 Gb/s, while (Offline, Limited, SPD) provides a stability limit of 0.91 Gb/s, only a 10 Mb/s difference.

V. CONCLUSION

The grant scheduling framework has a significant impact on average packet delay. By enhanced interleaved polling, granting cycle lengths are shortened given a particular set of grant sizes thereby reducing queueing delays. Decreased cycle lengths also increase channel utilization. Therefore, the grant scheduling framework has a significant impact on channel utilization as well.

The grant sizing policy has the largest impact on average packet delay. Larger grant sizes allow more packets to be served in a single granting cycle, thereby reducing queueing delay. However, increasing granting cycle lengths can counteract this and increase the queueing delay. The limited with excess distribution grant sizing policy allows for larger grant sizes while still maintaining a fixed upper bound on the granting cycle length. In contrast to the average packet delay, the grant sizing policy has a very small impact on the maximum achievable channel utilization.

The grant scheduling policy also has a significant impact on average packet delay. Using the SPD scheduling policy minimizes the propagation delay difference between subsequently polled ONUs. This minimizes the idle time between grants to these subsequent ONUs and thereby reduces the granting cycle lengths given a set of grant sizes. The grant scheduling policy also has a significant impact on the channel utilization as a result of the decreased cycle lengths.

Overall, our novel enhancement to the DPP scheduling framework that incorporates a limited with excess distribution grant sizing policy with excess credit sharing between polling groups, (DPP, Excess:Share, SPD), provides the overall best packet delay and maximum achievable channel utilization performance.

Avenues for future investigations include generalizations of the DPP scheduling framework to more than two scheduling groups as well as investigation of multichannel EPONs. Another important avenue is to examine the impact of the DBA design dimensions on delay variations and to explore mechanisms that jointly reduce mean delay as well as delay variations [39]–[41]. Moreover, the internetworking of EPON access networks, which were considered in this study in isolation, with metro networks, e.g., [42]–[45], poses several open research challenges for efficient interoperability and bandwidth management [46]–[48].

APPENDIX

In this appendix, we formally define the considered grant scheduling frameworks and specify their scheduling instants $\gamma(j,n)$ and the GATE signaling times $T(j,n)$.

A. Online Scheduling Framework

The granted transmission window for the jth ONU in granting cycle n is scheduled upon receipt of its REPORT message at the end of its granted transmission window for granting cycle $n-1$. Therefore,

$$\gamma(j,n) = \beta(j,n-1). \tag{19}$$

Only a single ONU is polled at $\gamma(j,n)$. Therefore,

$$T(j,n) = t_G + 2\tau(o(j,n)). \tag{20}$$

$o(j,n) = j$ assuming that the ONUs are initially polled by their ONU number.

B. Offline Scheduling Framework

The schedule for the entire granting cycle n is produced when the REPORT message is received from the last ONU to transmit upstream during granting cycle $n-1$. Therefore,

$$\gamma(j,n) = \beta(M,n-1), \forall\, j. \tag{21}$$

Subsequently, all GATE messages are sent back-to-back

$$T(j,n) = jt_G + 2\tau(o(j,n)). \tag{22}$$

$o(j,n)$ is determined by the scheduling policy (discussed in Section II-C).

C. OLS Scheduling Framework

The OLS scheduling framework [14] is directly coupled to limited with excess distribution grant sizing (see Section II-B.1). An ONU's load status determines when its granted transmission window in granting cycle n is scheduled. If it is underloaded, its granted transmission window is scheduled upon receipt of its REPORT message at the end of its granted transmission window for granting cycle $n-1$. Otherwise, its granted transmission window is scheduled when the REPORT message is received from the last ONU $(j=M)$ during granting cycle $n-1$. Thus,

$$\gamma(j,n)=\begin{cases}\beta(j,n-1): & j\in\mathcal{U}(n)\\ \beta(M,n-1): & j\in\mathcal{O}(n)\end{cases} \tag{23}$$

$$T(j,n)=\begin{cases}t_G+2\tau(o(j,n)): & j\in\mathcal{U}(n)\\ (j-|\mathcal{U}(n)|)t_G+2\tau(o(j,n)): & j\in\mathcal{O}(n)\end{cases}. \tag{24}$$

$o(j,n)$ is lower bounded by $|\mathcal{U}(n)|+1$ for ONUs in $\mathcal{O}(n)$ and upper bounded by $|\mathcal{U}(n)|$ for ONUs in $\mathcal{U}(n)$. Further, $o(j,n)$ is influenced by $o(j,n-1)$ for ONUs in $\mathcal{U}(n)$ and determined by the scheduling policy for ONUs in $\mathcal{O}(n)$.

D. DPP Scheduling Framework

ONUs are divided evenly into two polling groups; the first half $(j=1,\ldots,M/2)$ in polling group 1 and the second half $(j=M/2+1,\ldots,M)$ in polling group 2. Each polling group is scheduled independently. The schedule for the entire granting cycle n for a polling group is produced when the REPORT message is received from the last ONU for that polling group during granting cycle $n-1$. Therefore,

$$\gamma(j,n)=\begin{cases}\beta(M/2,n-1): & j\le M/2\\ \beta(M,n-1): & j>M/2.\end{cases} \tag{25}$$

The ONUs in a polling group are notified back-to-back. Therefore,

$$T(j,n)=\begin{cases}jt_G+2\tau(o(j,n)): & j\le M/2\\ (j-M/2)t_G+2\tau(o(j,n)): & j>M/2.\end{cases} \tag{26}$$

Here, $o(j,n)$ is determined by the scheduling policy within each group. A lower bound on $o(j,n)$ for the ONUs in group 2 is $M/2+1$.

ACKNOWLEDGMENT

The authors would like to acknowledge the assistance of D. Zhao of Arizona State University with the simulation experiments.

REFERENCES

[1] G. Kramer, B. Mukherjee, and G. Pesavento, "Interleaved polling with adaptive cycle time (IPACT): A dynamic bandwidth distribution scheme in an optical access network," *Photon. Netw. Commun.*, vol. 4, no. 1, pp. 89–107, Jan. 2002.

[2] M. McGarry, M. Reisslein, and M. Maier, "Ethernet passive optical network architectures and dynamic bandwidth allocation algorithms," *IEEE Commun. Surveys Tuts.*, vol. 10, no. 3, pp. 46–60, Third Quarter, 2008.

[3] J. Zheng and H. Mouftah, "A survey of dynamic bandwidth allocation algorithms for ethernet passive optical networks," *Opt. Switch. Netw.*, vol. 6, no. 3, pp. 151–162, Jul. 2009.

[4] M. Andrade, G. Kramer, L. Wosinska, J. Chen, S. Sallent, and B. Mukherjee, "Evaluating strategies for evolution of passive optical networks," *IEEE Commun. Mag.*, vol. 49, no. 7, pp. 176–184, Jul. 2011.

[5] R. Roy, G. Kramer, M. Hajduczenia, and H. Silva, "Performance of 10G-EPON," *IEEE Commun. Mag.*, vol. 49, no. 11, pp. 78–85, Nov. 2011.

[6] K. Fouli, M. Maier, and M. Medard, "Network coding in next-generation passive optical networks," *IEEE Commun. Mag.*, vol. 49, no. 9, pp. 38–46, Sep. 2011.

[7] O. C. Turna, M. A. Aydin, T. Atmaca, A. Zaim, and T.-D. Nguyen, "Traffic characterization study on EPON upstream channel," in *Proc. Int. Wireless Commun. Mobile Comput. Conf.*, 2011, pp. 1601–1606.

[8] M. Radivojevic and P. Matavulj, "Implementation of intra-ONU scheduling for quality of service support in Ethernet passive optical networks," *J. Lightw. Technol.*, vol. 27, no. 18, pp. 4055–4062, Sep. 2009.

[9] S. Choi, S. Lee, T. Lee, M. Chung, and H. Choo, "Double-phase polling algorithm based on partitioned ONU subgroups for high utilization in EPONs," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 1, no. 5, pp. 484–497, Oct. 2009.

[10] G. Levit, "Dynamic bandwidth allocation based on partial cycles," U.S. Patent 8 000 234B2, 2011.

[11] M. McGarry, M. Reisslein, C. Colbourn, M. Maier, F. Aurzada, and M. Scheutzow, "Just-in-time scheduling for multichannel EPONs," *J. Lightw. Technol.*, vol. 26, no. 10, pp. 1204–1216, May 2008.

[12] N. Jana, G. Sankaran, K. Sivalingam, and G. Parr, "Performance analysis of dynamic bandwidth allocation algorithms for long-reach PONs," in *Proc. IEEE Int. Symp. Adv. Netw. Telecommun. Syst.*, Dec. 2010, pp. 28–30.

[13] O. C. Turna, M. A. Aydin, T. Atmaca, and A. Zaim, "A novel dynamic bandwidth allocation algorithm based on half cycling for EPONs," in *Proc. Int. Conf. Emerg. Netw. Intell.*, Oct. 2010, pp. 38–43.

[14] C. Assi, Y. Ye, S. Dixit, and M. Ali, "Dynamic bandwidth allocation for quality-of-service over ethernet PONs," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 9, pp. 1467–1477, Nov. 2003.

[15] Y. Luo and N. Ansari, "Bandwidth allocation for multiservice access on EPONs," *IEEE Commun. Mag.*, vol. 43, no. 2, pp. S16–S21, Feb. 2005.

[16] Y. Luo and N. Ansari, "Limited sharing with traffic prediction for dynamic bandwidth allocation and QoS provisioning over EPONs," *OSA J. Opt. Netw.*, vol. 4, no. 9, pp. 561–572, 2005.

[17] Y. Zhu and M. Ma, "IPACT with grant estimation (IPACT-GE) scheme for ethernet passive optical networks," *J. Lightw. Technol.*, vol. 26, no. 14, pp. 2055–2063, Jul. 2008.

[18] J. Ferguson, M. Reisslein, and M. McGarry, "Online excess bandwidth distribution for ethernet passive optical networks," *OSA J. Opt. Netw.*, vol. 8, no. 4, pp. 358–369, Apr. 2009.

[19] W. Lim, C. Yun, Y. Yang, H. Lee, and K. Kim, "Burst-polling-based dynamic bandwidth allocation using adaptive minimum guaranteed bandwidth for EPONs," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 1, no. 7, pp. 594–599, Dec. 2009.

[20] H. Ikeda and K. Kitayama, "Dynamic bandwidth allocation with adaptive polling cycle for maximized TCP throughput in 10G-EPON," *J. Lightw. Technol.*, vol. 27, no. 23, pp. 5508–5516, Dec. 2009.

[21] A. Razmkhah and A. G. Rahbar, "OSLG: A new granting scheme in WDM ethernet passive optical networks," *Opt. Fiber Technol.*, vol. 17, no. 6, pp. 586–593, Dec. 2011.

[22] I.-S. Hwang, J.-Y. Lee, Y.-S. Liang, and Z.-D. Shyu, "A bipartitional dynamic bandwidth allocation mechanism for differentiated services to support state report ONUs in GPON," in *Proc. Int. Multiconf. Eng. Comput. Sci.*, 2011, pp. 1–5.

[23] J. Zheng and H. Mouftah, "Adaptive scheduling algorithms for ethernet passive optical networks," *IEE Proc. Commun.*, vol. 152, no. 6, pp. 643–647, Oct. 2005.

[24] K. Kim, D. Gutierrez, F. An, and L. Kazovsky, "Design and performance analysis of scheduling algorithms for WDM-PON under SUCCESS-HPON architecture," *J. Lightw. Technol.*, vol. 23, no. 11, pp. 3716–3731, Nov. 2005.

[25] M. Ma, L. Liu, and T. Cheng, “Adaptive scheduling for differentiated services in an ethernet passive optical network,” *OSA J. Opt. Netw.*, vol. 4, no. 10, pp. 661–67, 2005.

[26] B. Chen, J. Chen, and S. He, “Efficient and fine scheduling algorithm for bandwidth allocation in ethernet passive optical networks,” *IEEE J. Sel. Topics Quantum Electron.*, vol. 12, no. 4, pp. 653–660, Jul./Aug. 2006.

[27] M. De Andrade, L. Gutierrez, and S. Sallent, “A distributed scheduling approach for ethernet-based passive optical networks,” in *Proc. IEEE Conf. Local Comput. Netw.*, Oct. 2007, pp. 205–206.

[28] A. Dhaini, C. Assi, M. Maier, and A. Shami, “Dynamic wavelength and bandwidth allocation in hybrid TDM/WDM EPON networks,” *J. Lightw. Technol.*, vol. 25, no. 1, pp. 277–286, Jan. 2007.

[29] F. Melo Pereira, N. L. S. Fonseca, and D. S. Arantes, “A fair scheduling discipline for ethernet passive optical networks,” *Comput. Netw.*, vol. 53, no. 11, pp. 1859–1878, Jul. 2009.

[30] K. Kanonakis and I. Tomkos, “Improving the efficiency of online upstream scheduling and wavelength assignment in hybrid WDM/TDMA EPON networks,” *IEEE J. Sel. Areas Commun.*, vol. 28, no. 6, pp. 838–848, Aug. 2010.

[31] P. Sarigiannidis, S. Petridou, G. Papadimitriou, and M. Obaidat, “IGFS: A new MAC protocol exploiting heterogeneous propagation delays in the dynamic bandwidth allocation on WDM-EPON,” *IEEE Syst. J.*, vol. 4, no. 1, pp. 49–56, Mar. 2010.

[32] M. McGarry, M. Reisslein, F. Aurzada, and M. Scheutzow, “Shortest propagation delay (SPD) first scheduling for EPONs with heterogeneous propagation delays,” *IEEE J. Sel. Areas Commun.*, vol. 28, no. 6, pp. 849–862, Aug. 2010.

[33] X. Bai, C. Assi, and A. Shami, “On the fairness of dynamic bandwidth allocation schemes in ethernet passive optical networks,” *Comput. Commun.*, vol. 29, no. 11, pp. 2125–2135, Jul. 2006.

[34] H. Song, B. Kim, and B. Mukherjee, “Multi-thread polling: A dynamic bandwidth distribution scheme in long-reach PON,” *IEEE J. Sel. Areas Commun.*, vol. 27, no. 2, pp. 134–142, Feb. 2009.

[35] B. Kantarci and H. Mouftah, “Bandwidth distribution solutions for performance enhancement in long-reach passive optical networks,” *IEEE Commun. Surveys Tuts.*, 2012, to be published.

[36] A. Dhaini, C. Assi, and A. Shami, “Quality of service in TDM/WDM ethernet passive optical networks (EPONs),” in *Proc. IEEE Int. Symp. Comput. Commun.*, Jun. 2006, pp. 616–621.

[37] S. Bhatia and R. Bartos, “IPACT with smallest available report first: A new DBA algorithm for EPON,” in *Proc. IEEE Int. Conf. Commun.*, Jun. 2007, pp. 2168–2173.

[38] Csim (Mesquite Software) [Online]. Available: http://www.mesquite.com

[39] T. Berisa, Z. Ilic, and A. Bazant, “Absolute delay variation guarantees in passive optical networks,” *J. Lightw. Technol.*, vol. 29, no. 9, pp. 1383–1393, May 2011.

[40] A. Dixit, G. Das, B. Lannoo, D. Colle, M. Pickavet, and P. Demeester, “Jitter performance for QoS in ethernet passive optical networks,” in *Proc. Eur. Conf. Opt. Commun.*, Sep. 2011, pp. 1–3.

[41] A. Shami, X. Bai, C. Assi, and N. Ghani, “Jitter performance in ethernet passive optical networks,” *J. Lightw. Technol.*, vol. 23, no. 4, pp. 1745–1753, Apr. 2005.

[42] A. Carena, V. D. Feo, J. M. Finochietto, R. Gaudino, F. Neri, C. Piglione, and P. Poggiolini, “RingO: An experimental WDM optical packet network for metro applications,” *IEEE J. Sel. Areas Commun.*, vol. 22, no. 8, pp. 1561–1571, Oct. 2004.

[43] M. Maier, M. Reisslein, and A. Wolisz, “A hybrid MAC protocol for a metro WDM network using multiple free spectral ranges of an arrayed-waveguide grating,” *Comput. Netw.*, vol. 41, no. 4, pp. 407–433, Mar. 2003.

[44] M. Scheutzow, M. Maier, M. Reisslein, and A. Wolisz, “Wavelength reuse for efficient packet-switched transport in an AWG-based metro WDM network,” *J. Lightw. Technol.*, vol. 21, no. 6, pp. 1435–1455, Jun. 2003.

[45] D. Wonglumsom, I. M. White, K. Shrikhande, M. S. Rogge, S. M. Gemelos, F.-T. An, Y. Fukashiro, M. Avenarius, and L. G. Kazovsky, “Experimental demonstration on an access point for HORNET—A packet-over-WDM multiple-access MAN,” *J. Lightw. Technol.*, vol. 18, no. 12, pp. 1709–1717, Dec. 2000.

[46] A. Ahmed and A. Shami, “RPR-EPON-WiMAX hybrid network: A solution for access and metro networks,” *IEEE/OSA J. Opt. Commun. Netw.*, vol. 4, no. 3, pp. 173–188, Mar. 2012.

[47] H.-C. Leligou, E. Kosmatos, and A. Stavdas, “Future internet infrastructure based on the transparent integration of access and core optical transport networks,” *IEEE/OSA J. Opt. Commun. Netw.*, vol. 1, no. 2, pp. A205–A218, Jul. 2009.

[48] J. Segarra, V. Sales, and J. Prat, “An all-optical access-metro interface for hybrid WDM/TDM PON based on OBS,” *J. Lightw. Technol.*, vol. 25, no. 4, pp. 1002–1016, Apr. 2007.

Michael P. McGarry (M’98) received the B.S. degree in computer engineering from the Polytechnic Institute of New York University, Brooklyn, in 1997, and the M.S. and Ph.D. degrees in electrical engineering from Arizona State University, Tempe, in 2004 and 2007, respectively.

He is currently an Assistant Professor in the Department of Electrical and Computer Engineering, University of Texas at El Paso, El Paso. His research interests include the optimization of medium access control protocols, error recovery mechanisms for video communication, bandwidth forecasting, and congestion control.

Dr. McGarry is a co-recipient of the 2009 IEEE Communications Society Best Tutorial Paper Award.

Martin Reisslein (A’96–S’97–M’98–SM’03) received the Ph.D. degree in systems engineering from the University of Pennsylvania, Philadelphia, in 1998.

He is currently a Professor in the School of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe. From July 1998 to October 2000, he was a Scientist with the German National Research Center for Information Technology (GMD FOKUS), Berlin, Germany, and a Lecturer at the Technical University Berlin.

Dr. Reisslein is an Associate Editor for the IEEE/ACM TRANSACTIONS NETWORKING and for *Computer Networks*. His research interests include multimedia networking, optical access networks, and engineering education.