Parallel Void Thread In Long-Reach Ethernet Passive Optical Networks

Amr Elrasad, Student Member, IEEE and Basem Shihada, Senior Member, IEEE

Abstract—This work investigates void filling (idle periods) in Long-Reach Ethernet Passive Optical Networks (LR-EPONs). We emphasize on reducing grant delays and hence reducing the average packet delay. We introduce a novel approach called, Parallel Void Thread (PVT), which allocates bandwidth grants during voids baseless of bandwidth requests. We introduce three different grant sizing schemes for PVT namely Void Extension (VE), Count Controlled Batch Void Filling (CCBVF), and Size Controlled Batch Void Filling (SCBVF). The proposed approaches can be integrated with almost all of the previously reported dynamic bandwidth allocation schemes. Unlike other void filling schemes, PVT is less sensitive to differential distance between ONUs and can work very well in case of limited differential distances. We have analytically investigated the packet delay and derived a bound condition for PVT to outperform the other competitors. We support our work by extensive simulation study considering bursty traffic with long range dependence for both single class and differentiated services (DiffServ) scenarios. Numerical results show delay reduction up to 35% compared to non-void filling scheme for single class scenario. For DiffServ traffic, PVT achieves delay reduction up to 80% for expedited forward (EF) traffic, 52% for assured forward (AF) traffic, and 56% for best effort (BE) traffic.

Index Terms—Long-Reach Ethernet passive optical networks (LR-EPONs); Optical access networks; Void filling.

I. INTRODUCTION

Ethernet passive optical networks (EPONs) is one of the promising solutions to satisfy the increasing bandwidth demand in access network. It can support a transmission capacity up to 10 Gbps in both upstream and downstream directions. Upstream bandwidth assignment in EPONs is implemented using polling strategy to eliminate collisions. The optical-line-terminal (OLT) unit polls the optical-network-units (ONUs) to transmit according to a certain order. Polling can either be online (interleaved) or offline. In online polling, the grant assignment is carried on immediately after the OLT receives bandwidth request from ONU. On the other hand, the OLT waits until it receives all bandwidth requests before granting the assignment in offline polling. The notations used hereafter are shown in Table I.

The packet delay components $W_{poll}$, $W_{grant}$, and $W_{queue}$ are shown in Fig.1. $W_{poll}$ is the delay between packet arrival and bandwidth request transmission. On average, $W_{poll}$ equals half the duration of the cycle time. $W_{grant}$ is the delay between grant request and grant assignment for certain packet. $W_{grant}$ can span over multiple cycles but it can not be less than $RTT_i + t_c$. Hence the minimum grant start time is

$$t_{s}^{\min}(i, n) = t_{c}(i, n - 1) + RTT_i + t_c.$$  

$W_{queue}$ is the delay between grant start and packet transmission and it depends on both the buffer size seen by the packet, the grant size, and the packet transmission discipline. In general we state that

$$\bar{W} \geq 1.5RTT_{grant}.$$  

Dynamic bandwidth allocation (DBA) is the most vital component in EPON as the network performance heavily relies on it. DBA consists of two main functions namely grant sizing and grant scheduling. Grant sizing determines the amount of bandwidth grant assigned to each ONU during cycle. This decision is based on the bandwidth requests sent from ONU via a report message during the previous grant. This kind of grant will be referred to as request based grant (RBG). Most of the proposed DBA schemes apply a limited grant sizing policy in which the grant assigned can not exceed a certain threshold $C_{i}^{max}$ [1]. Grant scheduling determines both start and end time of bandwidth grant. The most common scheduling approach is to use the horizon (first free) time or non-void filling (NVF) as shown in Fig.2. Assuming cyclic polling, grant start time is given by,

$$t_1 = \left\{ \begin{array}{ll}
max(t_1(N, n - 1), t_1^{\min}(i, n)) & i = 1, \\
max(t_1(i - 1, n), t_1^{\min}(i, n)) & 1 < i \leq N,
\end{array} \right.$$  

In long-reach Ethernet passive optical networks (LR-EPONs), the distance between OLT and ONUs spans over a longer distance (up to 100 Km) and hence longer $RTT_i$. This is reflected on both $W_{grant}$ and void time periods between bandwidth grants [2]. In such a case, $RTT_i$ will be the dominant factor controlling the packet delay at low and medium load [3] such that

$$\bar{W} \approx 1.5RTT_{grant}.$$  

The negative effect of such voids motivates many researchers to propose new DBA schemes to further reduce the delay [2], [4]–[9]. The proposed ideas fall into two categories. The first one aims at reducing the void size while maintaining the cyclic bandwidth allocation order [2], [6]. The second one aims at filling the void with request based grants [4], [5], [8]. The numerical results presented in [3]–[5] show that the proposed schemes could not achieve delay below $1.5RTT$ (see section IV for more details). This is very crucial for delay sensitive traffic. In this paper, we propose a parallel void
TABLE I: Definition of Variables

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$N$</td>
<td>Number of ONUs, indexed $i = 1, 2, ..., N</td>
</tr>
<tr>
<td>$C$</td>
<td>Transmission capacity in bps</td>
</tr>
<tr>
<td>$T_{max}$</td>
<td>Maximum cycle time</td>
</tr>
<tr>
<td>$G_i$</td>
<td>Maximum allocated grant per cycle $i$</td>
</tr>
<tr>
<td>$R_i$</td>
<td>ith ONU bandwidth request for cycle $i$</td>
</tr>
<tr>
<td>$G_i$</td>
<td>ith ONU bandwidth grant for cycle $i$</td>
</tr>
<tr>
<td>$w_i$</td>
<td>ith ONU weight according to service level agreement (SLA)</td>
</tr>
<tr>
<td>$t_g$</td>
<td>Guard time between consecutive grants</td>
</tr>
<tr>
<td>$t_e$</td>
<td>Time needed to transmit Gate or report message</td>
</tr>
<tr>
<td>$D_i$</td>
<td>Distance between OLT and ith ONU</td>
</tr>
<tr>
<td>$RTT_i$</td>
<td>Round-trip-time delay between OLT and ith ONU</td>
</tr>
<tr>
<td>$RTT_{max}$</td>
<td>Average round-trip-time delay</td>
</tr>
<tr>
<td>$RTT_{min}$</td>
<td>Minimum round-trip-time delay</td>
</tr>
<tr>
<td>$t_s(i,n)$</td>
<td>ith ONU request based grant start time in cycle n</td>
</tr>
<tr>
<td>$t_e(i,n)$</td>
<td>ith ONU request based grant end time in cycle n</td>
</tr>
<tr>
<td>$t_{min}(i,n)$</td>
<td>ith ONU request based grant minimum start time in cycle n</td>
</tr>
<tr>
<td>$V_t$</td>
<td>The horizon time after ith ONU grant allocation in cycle n</td>
</tr>
<tr>
<td>$V_e$</td>
<td>Void succeeding ith ONU request based grant</td>
</tr>
<tr>
<td>$V_{s}(i,n)$</td>
<td>Start time of void succeeding ith ONU request based grant</td>
</tr>
<tr>
<td>$V_{e}(i,n)$</td>
<td>End time of void succeeding ith ONU request based grant</td>
</tr>
<tr>
<td>$W$</td>
<td>Packet delay</td>
</tr>
<tr>
<td>$W_{poll}$</td>
<td>Polling delay</td>
</tr>
<tr>
<td>$W_{grant}$</td>
<td>Grant delay</td>
</tr>
<tr>
<td>$W_{queue}$</td>
<td>Queuing delay</td>
</tr>
<tr>
<td>$K_B$</td>
<td>The batch size in CCBVF</td>
</tr>
<tr>
<td>$V_{max}$</td>
<td>The maximum allocated void based grant for SCBVF</td>
</tr>
</tbody>
</table>

Upon receiving bandwidth request and employing grant sizing policy to determine $G(i,n)$, the OLT begins to search for eligible void that fits the bandwidth grant. A void $V_e$ is considered usable if it matches two conditions described as follows,

$$V_e(k,n) - V_e(k,n) ≥ G(i,n)$$ (5)


and

$$V_e(k,n) - G(i,n) ≥ t_{min}(i,n).$$ (6)

If $V_e$ is satisfies the above conditions, the grant start and end times are given by

$$t_s(i,n) = max(t_{min}(i,n), V_e(k,n))$$ (7)


and

$$t_e(i,n) = t_s(i,n) + G(i,n).$$ (8)

The selected void is the one that has minimum $t_s(i,n)$ among the set of eligible voids. The computation complexity of RBVF is $O(N)$. If binary search tree is used, the computation complexity is reduced to $O(log N)$ [4]. The results in [4] show that RBVF reduces the average delay compared to NFV. However, it does not reduce the average delay below the $1.5RTT$ bound. In the best case each ONU can not transmit less than every $RTT$, RBVF performance improvement decreases with ONUs with small distance variations. If all ONUs have the same $D_i$, RBVF can not reduce the average delay compared to NFV [4].

In [5], authors proposed a request based partial void filling (RBPVF) scheme to further improve RBVF. Their algorithm is a mix between RBVF and multi-thread polling (MTP) [6]. Upon receiving bandwidth request, the OLT invokes RBVF to fit the grant within one of the available voids. If the OLT can not find suitable void, it invokes RBPVF that can divide $G(i,n)$ into at most $P$ chunks and switch the corresponding ONU to multi-thread mode. During multi-thread mode, the OLT can not invoke RBPVF and it can use RBVF only. The OLT starts killing the threads requesting Zero bandwidth till there is only one thread left. The OLT switches this ONU back to single-thread mode again. This method enables long grants that do not fit in a single void to be divided into two or more grants. Their results shows a delay improvement compared to EFT-RBVF [4], but the minimum delay bound can be reached is 1.5$RTT$ for the same reasons listed above.

II. REQUEST BASED VOID FILLING

In [4], a request based void filling was proposed to fit grant requests from relatively nearer ONUs into available voids as shown in Fig.3.

III. PARALLEL VOID THREAD

Parallel void Thread (PVT) relies on void detection after an RBG is assigned by the OLT as shown in Fig.4. Upon receiving grant request from ith ONU, the OLT schedules the RBG grant start time as in (3) and updates the horizon time, $V_e(i,n)$. In order to detect if there is a void succeeding ith ONU, the OLT compares $V_e(i,n)$ with $t_{min}(i+1,n)$. If

$$V_e(i,n) < t_{min}(i+1,n)$$ (9)

then the OLT detects a void, $V_e$, with start and end times as,

$$V_e(i,n) = t_s(i,n)$$ (10)


and

$$V_e(i,n) = t_{max}(i+1,n).$$ (11)

If $i = N$, then the above equations are modified to,

$$V_e(N,n) = t_{max}(N,n)$$ (12)


and

$$V_e(N,n) = t_{max}(1,n+1).$$ (14)
Upon void detection, the OLT invokes PVT immediately to fill the detected voids with void-based grant(s) VBG(s). Since ONUs are polled in cyclic order, the void detection computation complexity is of $O(1)$. The size of VBGs only depends on the detected void duration. It is worth noting that during VBG, the ONU will not send bandwidth requests. In fact, bandwidth requests are only sent during RBG. We propose three different VBG sizing schemes to fill the detected voids, namely void extension (VE), count control batch void filling (CCBVF), and size control batch void filling (SCBVF).

**A. Void Extension**

Void extension (VE) assigns ith ONU a VBG with a duration of the detected void $V_i$ as shown in Fig.5. It is worth noting that VE is different from LOHEDA [8] in two ways. First, VE fills the void with a separate grant rather than merging both the RBG and VBG into a single grant. This is because enlarging the grant will further delay the bandwidth request (report message) and creates another void in the next cycle [2]. Second, VE does not delay the grant of ONU ($i - 1$) till it receives the bandwidth request of ith ONU as LOHEDA does.

**B. Count Controlled Batch Void Filling**

In CCBVF, a batch of $K_B$ ONUs are assigned VBGs during the detected void. Fig.6 demonstrates CCBVF for $K_B = 2$. Regarding VBG assignment, ONUs are polled according to cyclic order that is independent of the cyclic order used for RBG. For CCBVF, an eligible void must satisfy the following condition:

$$V_e(i, n) - V_s(i, n) \geq K_B \times (t_s + t_g). \quad (15)$$

This condition is necessary to ensure that the allocated VBGs are sufficient to send the shortest packet but not sufficient to ensure that all VBGs will be utilized. As $K_B$ increases, VBG becomes shorter due to increased guard periods. The amount of VBG assigned to each ONU is,

$$VBG_i = \frac{w_i}{\sum_{j=1}^{K_B} w_j} (V_e(i, n) - V_s(i, n) - K_B \times t_g). \quad (16)$$

**C. Size Controlled Batch Void Filling**

SCBVF assigns VBGs based on size rather than on the batch count ($K_B$) as CCBVF. In SCBVF, each VBG does not exceed a certain threshold $V_{max}$ which is set based on each ONU relative weight $w_i$ such that:

$$\frac{V_{max}}{w_j} \geq w_i, \quad (17)$$

where $i$ and $j \in \{1, 2, ..., N - 1\}$. SCBVF continues to allocate VBGs with the maximum void duration until a polled ONU can not be granted its maximum void duration. In order to maximize the benefit of void filling, this ONU will be granted the remaining grant rather than leaving it unscheduled. The operation of SCBVF is shown in Fig.7. In this figure, we observe that ONU 3 has been allocated it's maximum VBG size while ONU 4 was granted the remaining part.

**IV. VOID FILLING SCHEMES DELAY ANALYSIS**

In this section, we present delay analysis for the proposed void filling techniques. We would like to highlight that:

- The delay analysis is presented for low to moderate loads to be conformant with the assumptions presented in Section IV-A.
- The delay analysis purpose is to analytically explain the delay reduction mechanism of PVT and to provide a comparative analysis to NVF, RBPVF, and RBPVF.
- The outcome of the delay analysis is considered sufficient but not necessary condition for PVT to achieve delay lower than its competitors.

**A. Preliminaries**

In order to ensure that the existence of voids, we assume that,

$$\sum_{i=0}^{N-1} G(i, n) < RTT^{max} - Nt_g, \quad (18)$$

and

$$\sum_{i=0}^{N-1} G_{i}^{max} > RTT^{max} - Nt_g. \quad (19)$$

We also assume packets arrive according to poisson process. For LR EPONs, we neglect $W_{queue}$ as it is very small compared to $W_{poll}$ and $W_{grant}$. We also assume that $W_{grant}$ only spans over one cycle as in the case of low load. The offered load is uniformly distributed among ONUs. This leads to uniformly distributed voids along $W_{grant}$. 
It is clear from Fig. 1 that $W_{poll}$ depends on $W_{grant}$. For the best case, $W_{poll} = 0$ and for the worst case, $W_{poll} = W_{grant}$. On average, we can state that

$$W_{poll} = 0.5W_{grant}, \quad (20)$$

and

$$W = 1.5W_{grant}, \quad (21)$$

B. Non Void Filling (NVF)

For NVF, $W_{grant} = RTT^{max}$. Based on (21), the average delay of NVF is

$$\overline{W} = 1.5RTT^{max}. \quad (22)$$

C. Request Based Void Filling and Request Based Partial Void Filling

Both RBVF and RBPVF try to fit the bandwidth grants as close to $RTT_i$ as possible. For $i$th ONU, the grant delay is bounded by $RTT_i$ and $RTT^{max}$ as lower and upper bounds respectively. On average, it can be stated that:

$$RTT \leq W_{grant} \leq RTT^{max}. \quad (23)$$

Based on (21) and (23), the average delay of RBVF at low load is bounded by

$$1.5RTT \leq \overline{W} \leq 1.5RTT^{max}. \quad (24)$$

D. Count Controlled Batch Void Filling (CCBVVF) and Size Controlled Batch Void Filling (SCBVVF)

Let $A(i,n)$ be the number of packets arrive between $t_s(i,n-1) - t_e$ and $t_s(i,n) - t_e$. Those packets are decomposed according to their transmission time as,

- $B(i,n)$: packets transmitted at the end of RBG slot during cycle $n$.
- $V(i,n)$: packets transmitted during VBG slot(s) allocated uniformly between $t_s(i,n)$ and $t_s(i,n+1)$.
- $Y(i,n)$: packets transmitted at the start of RBG slot during cycle $(n+1)$.

The timing diagram of PVT packets is shown in Fig. 8. $B(i,n)$ mainly depends on both $V(i,n)$ and $A(i,n)$. $B(i,n)$ packets are transmitted during the unutilized bandwidth at the end of RBG slot of cycle $n$ resulting from the transmission of $V(i,n)$ during the VBG slot(s) before the RBG slot. In order to calculate $\overline{W}$, the delay of each type should be calculated.

In order to make the analysis trackable, we consider the case where there is always remaining packets to be sent during RBG slot, (i.e., $Y(i,n) > 0$). This implies that the calculated average delay is an upper bound of the real average delay.

Let $\overline{B}$, $\overline{V}$, $\overline{Y}$, and $A$ be the average values of $B(i,n)$, $V(i,n)$, $Y(i,n)$, and $A(i,n)$ respectively. It is also worth noting that for any two consecutive RBG slots, the following relation is valid

$$t_s(i,n) - t_e(i,n-1) \approx RTT^{max}. \quad (25)$$

$t_e$ and report message processing time are neglected as they are very small compared to round-trip-time in LR-EPONs.

$B(i,n)$ packets arrive starts at $t_s(i,n-1) - t_e$ and is on average $\frac{3}{2}RTT^{max}$ long. They are transmitted at the end of RBG slot of cycle $n$. This makes their average arrival time is $t_s(i,n-1) - t_e + \frac{3}{2}RTT^{max}$ and their average transmission time is $t_s(i,n) - t_e$. Hence, the average delay of these packets ($\overline{W}_B$) is

$$\overline{W}_B = (1 - \frac{B}{A})RTT^{max}. \quad (26)$$

$V(i,n)$ packets arrival on average starts at $t_s(i,n-1) - t_e + \frac{3}{2}RTT^{max}$ and lasts for $\sum RTT^{max}$. $V(i,n)$ packets are transmitted between $t_s(i,n)$ and $t_s(i,n+1)$. The average delay of $V(i,n)$ packets is

$$\overline{W}_V = (3 - \frac{\overline{B}}{A} - \overline{V})RTT^{max}. \quad (27)$$

$Y(i,n)$ are the packets yet to be sent during RBG slot of cycle $(n+1)$. Their transmission starts at $t_s(i,n+1)$, while their arrival on average starts after the $V(i,n)$ packets arrival and lasts for $\sum RTT^{max}$. The average delay of these packets ($\overline{W}_Y$) is

$$\overline{W}_Y = (3 - \frac{\overline{B}}{A} - \overline{V})RTT^{max}. \quad (28)$$

The average delay of PVT is given by

$$\overline{W} = \frac{\overline{B}}{A} \overline{W}_B + \frac{\overline{V}}{A} \overline{W}_V + \frac{\overline{Y}}{A} \overline{W}_Y. \quad (29)$$

(29) is simplified to

$$\overline{W} = \frac{3}{2} - \frac{\overline{B}}{2A} - \frac{\overline{V}}{2A}RTT^{max}. \quad (30)$$

The remaining challenge is how to estimate both $\overline{B}$ and $\overline{V}$. In fact, $\overline{V}$ depends on the used grant sizing approach. Both CCBVF and SCBVVF have different values for $\overline{V}$ and $\overline{B}$. In order to know estimate them, we should know the unfinished work distribution for the polling system under time limited service discipline. In fact, this is still not solved problem yet [10]. For Poisson arrivals, the void size to grant size ratio (VGR) can be computed as,

$$VGR = \frac{1}{\rho}(1 - \rho - \frac{N_t}{RTT^{max}}). \quad (31)$$

where $\rho$ is the offered load. For the offered load below 0.5, VGR is greater than 1, which means that there is “theoretically” available bandwidth for all reported data to be sent during VBG(s).

E. Void Extension (VE)

The analysis of VE is similar to both CCBVF and SCBVVF except for $V(i,n)$ packets. $V(i,n)$ packets are transmitted immediately after $t_s(i,n)$. The average delay of $V(i,n)$ packets is

$$\overline{W}_V = (1 - \frac{\overline{B}}{A} - \overline{V})RTT^{max}. \quad (32)$$

Substitute by (32) in (29), we get

$$\overline{W} = \frac{3}{2} - \frac{\overline{B}}{2A} - \frac{\overline{V}}{2A}RTT^{max}. \quad (33)$$

V. DISCUSSION

In this section, we discuss the relevant delay, fairness, and overhead factors that is associated with void filling schemes.

A. Delay Discussion

The purpose of this section is to compare between the average delay of the proposed void filling schemes. Based on the average
We consider an EPON with single OLT and 32 ONUs with 10 MB buffer size. Otherwise mentioned, the upstream and the downstream transmission rates are symmetric with 1 Gbps. Ethernet frames size ranges between 64 to 1518 bytes, (we used the packet size distribution delay driven in (22), (30), and (33), It is clear that PVT delay is always less than NVF delay. CCBVF and SCBVF are guaranteed to achieve less delay than RBVF and RBPVF if
\[
\frac{B}{A} + \frac{V}{2A} > \frac{3}{2} \left(1 - \frac{RTT}{RTT_{max}}\right).
\]

VE is guaranteed to achieve delay lower than 1.5RTT bound if
\[
\frac{B}{A} + \frac{V}{A} > \frac{3}{2} \left(1 - \frac{RTT}{RTT_{max}}\right).
\]

We will refer to \(\frac{B}{A} + \frac{V}{2A}\) and \(\frac{B}{A} + \frac{V}{A}\) as Batch Void Filling (BVF) bound \((RV_{BV})\) and VE bound \((RV_{E})\) respectively. The derived lower bounds in (34) and (35) are sufficient but not necessary as the derived average delay in (30) and (33) is upper delay bound. For uniformly distributed ONUs between 80-100 Km, CCBVF and SCBVF are required to achieve \(R_{BV,F}\) less than 15\% to ensure average delay lower than the 1.5RTT bound. However, VE can achieve the same figure with \(R_{V,E}\) less than 15\%.

B. Control Messages Proliferation

Control message proliferation is considered as a side effect of void filling techniques in both upstream and downstream directions. RBVF slightly increases both report and gate messages load. On the other hand, RBVF increases the report message load more than RBVF as each partial void grant should end with a report message. PVT proposed approaches do not increase the report message load as VBGs do not conclude with a report message. In downstream direction, the situation is different based on what kind of PVT is applied. For VE, there is no increase in gate message load as both RBG and VBG can be sent together in a single gate message. For CCBVF and SCBVF, the amount of increase depends on both \(K_{B}\) and \(V_{max}\). It is also affected by the offered load which controls the average void duration.

C. Bandwidth Assignment Fairness

Both RBVF and RBPVF suffer from unfair bandwidth assignment. This is because their void filling techniques helps near ONUs to flush their data during voids rather than far ones. From (23), it is clear that both RBVF and RBPVF have higher grant delays for far nodes.

Regarding PVT VBG sizing approaches, VE is less fair compared to CCBVF and SCBVF. In VE, the VBG size assigned to each node depends on the ONU RBG size, the next node previous RBG size, and the round-trip-time. In case of non-uniform load, ONUs VBG assignment is not strictly fair. CCBVF is more fair compared to SCBVF as it allocates VBGs based on ONUs relative weights but this comes on the expense of VBG size control. SCBVF is less fair as the last scheduled ONU during a void should be assigned a VBG less than it’s maximum VBG value compared to the other ONU scheduled in the same void. Although voids are not similar in the size. However, on average, both CCBVF and SCBVF assign VBG more fair than VE.

D. Dynamic Bandwidth Allocation

PVT can be integrated with almost all proposed DBA schemes for LR-EPONs. For offline (interleaved-polling-with-stop) DBA schemes, PVT can be used with offline multi-thread polling (MTP-offline) [6], offline single thread (STP-offline) [6], and Double Phase (DP) polling [11]. It also can be used with online (interleaved) polling such as online multi-thread polling (MTP-online) [2], delayed excess scheduling (DES) [12] or online single thread polling with online excess allocation (STP-online-excess) [2]. The results in [2] shows that STP-online-excess outperforms MTP-offline and MTP-online at \(T_{cycle}^{max}\), up to 4 ms. Moreover, STP-online-excess is not as complex as multi-thread polling in terms of reporting process and thread tuning. Although MTP-online has less void size since there is \(\Theta N\) RBGs per cycle for \(\Theta \) threads. This implies that the cycle contains almost the same idle time amount but divided into more slots.

For offline schemes, SCBVF efficiently utilizes more voids compared to VE and CCBVF. Since in offline polling, there is a single large void at the end of cycle, VE will assign this void to the last node in the cycle, while CBBVF will divide it among \(K_{B}\) VBGs. SCBVF seems to be a reasonable option in that case as it will utilize the large void with the maximum possible VBGs. On the other hand, all PVT approaches efficiently work with online polling. For more information about the DBA schemes for LR-PONs, the reader might refer to [13].

E. PVT Computation Complexity

As we mentioned before, PVT consists of two main functions: void detection and VBG sizing. Void detection computation complexity is \(O(1)\), while VBG sizing complexity mainly depends on the grant sizing approach. In general, the complexity of allocating single VBG is \(O(1)\) as PVT is independent from the running DBA thread. For VE, the computation complexity is \(O(1)\) as it allocates single VBG per void. CCBVF has computation complexity of \(O(K_{B})\) as it performs \(K_{B}\) VBG allocations per void. Regarding SCBVF, the computation complexity depends on the number of VBG allocations per void. For the worst case scenario, the computation complexity is \(O\left(\left(\frac{RTT_{max}}{RTT_{min}}\right)\right)\) assuming all ONUs have the same SLA. This case is very rare as it requires that all ONUs have no data to send and the nearest ONU RBG is followed by the most far ONU. This case at most happens once per cycle. In realistic scenario, the average computation complexity is \(O\left(\frac{average\ void\ size}{V_{max}}\right)\).
reported in [14], with minimum inter-frame-gap (IFG) of 12 bytes and preamble of 8 bytes. We use two scenarios for the incoming traffic. The first scenario is single service self-similar traffic with long range dependence (LRD) and Hurst parameter 0.8 and packets are served in first-come-first-served (FCFS) order. The second scenario is differentiated services (DiffServ) traffic with 3 classes of service: Expedited Forward (EF), Assured Forward (AF), and Best Effort (BE). EF ONU offered load share is 20%, while the rest is divided equally between AF and BE. EF is constant-bit-rate (CBR) traffic with Poisson arrivals and fixed packet size of 70 bytes, while AF and BE traffic are self-similar with long range dependence and Hurst parameter 0.8. In order to maintain a fair comparison, we impose a strict priority queuing [15] for Intra-ONU scheduling in the case of DiffServ traffic. $T_{cycle}^{max}$ is set to 4 ms, $t_g$ is set to 1 µs and $G_{i}^{max}$ = 15500 bytes. The offered load is distributed uniformly over ONUs. We chose STP-online-excess [2] as the underlying non-void DBA scheme with excess pool bound of $NG_{i}^{max}$ bytes. STP-online-excess is the main running thread in case of PVT, RBVF, and RBPVF simulation. We will refer to STP-online-excess as NVF. PVT is compared against NVF, RBVF, and RBPVF.

![Fig. 10: (a) Average delay, $D_i=80$-100 Km (b) Average delay reduction, $D_i=80$-100 Km.](image)

![Fig. 11: (a) BVF bound (b) VE bound, $D_i=80$-100 Km](image)

**VII. PERFORMANCE EVALUATION**

**A. Single Service Traffic**

The average void duration of STP-online-excess is shown in Fig 9 for two different distance spans. The results show the void duration decreases with offered load increase. The void duration approximately ranges from 15 to 24 µs at load below 0.5. If we consider the total voids per cycle duration ($\approx 1$ms), the idle time duration represents 50-75% of the cycle length.

Fig. 10 shows the delay reduction comparison among NVF, RBVF, RBPVF, VE, CCBVF, and SCBVF for $D_i = 80$ – 100 Km. At 0.7 load and above, all schemes have approximately similar delay. CCBVF and SCBVF are better than NVF, VE, RBVF, and RBPVF for all load range below 0.7. The delay reduction ratio is calculated based on NVF delay. RBVF achieves delay reduction ratio is up to 5%. VE achieves a delay reduction up to 6%. RBPVF outperforms VE for load range below 0.4. Regarding CCBVF, it achieves delay reduction up to 12% for $K_B = 2$ and 19% for $K_B = 4$. SCBVF achieves the largest delay reduction compared to the other schemes. Its delay reduction is up to 35% and 37% for $v_{max}^{max} = 1538$ bytes and $v_{max}^{max} = 2500$ bytes respectively. It is also worth mentioning that both CCBVF and SCBVF achieves an average of a delay lower than $RTT^{max}$ bound. Both CCBVF and SCBVF delay begins to quickly increase starting from 0.6 offered load. This is because that the time between consecutive grants begins to increase beyond $RTT^{max}$ due
to increased burst arrivals.

In Fig.11, we show the bounds derived in (34) and (35). We observe that although these bounds are for Poisson arrivals and are sufficient but not necessary, they approximately hold for LRD traffic. Fig.11 (a) shows the bound for CCBVF and SCBVF. It can be seen that CCBVF did not achieve the minimum bound for $K_B = 2$ and hence it did not achieve delay lower than $1.5RTT$ (see Fig.10). However, CCBVF achieves $R_{BVVF}$ more than bound up to load $\approx 0.55$. SCBVF also achieves $R_{BVVF}$ more than the bound up to load $\approx 0.65$. VE did not achieve $R_{VE}$ more than the bound in order to achieve average delay lower than $1.5RTT$.

The reason behind VE, CCBVF, and SCBVF excel is shown in Fig.12. It shows four performance measures namely traffic ratio transmitted during VBG $(\frac{W}{G})$, average delay of traffic transmitted during VBG $(\frac{1}{R})$, average delay of traffic transmitted during RBG, and VBG utilization.

Fig.12(a) shows that the traffic transmitted during VBGs decreases with the offered load as the available voids becomes shorter. VE achieves $\frac{W}{G}$ of 6%, CCBVF achieves up to 28% for $K_B = 4$, and SCBVF achieves up to 52% for $V_{max} = 2500$ bytes. The reason behind VE performance lag compared to CCBVF and SCBVF can be explained as follows. Although VE archives the minimum delay in VBG, this is not enough to achieve large overall delay reduction as the traffic percent transmitted during VBG is less than the other competitors. In fact, the void duration is to great extent correlated with $RTT$ and $G(i,n)$. When ONU is lightly loaded, it is more likely to have a larger void succeeding it. In that case, VE will not be of much good as there is no much traffic to be sent during VBG. On the other hand, batch void filling enables more packets to be sent during VBG. This observation is emphasized by VBG utilization shown in Fig.12(d).

Although SCBVF has comparable VBG traffic delay to CCBVF, SCBVF can transmit more packets during VBG and hence reduce the overall traffic. Fig.12(c) highlights that VE, CCBVF, and SCBVF have less RBG delay compared to NVF. SCBVF has the least delay of RBG traffic because it is capable of sending more packets during VBGs (high $\frac{W}{G}$). This would provide available bandwidth for some of recent arriving packets to be sent at the end of the next RBG (more $\frac{G}{R}$). It can be noticed that the traffic ratio sent during VBGs and the RBG traffic delay are correlated. In other words, the scheme that manages to send more packets during VBGs is the one able to achieve the lowest delay during RBGs.

VBG utilization performance has two phases. The first phase, at low and medium loads, VBG utilization increases with load increase
making use of longer available voids and increased buffered packets. In the second phase (at high load), VBG utilization begins to decrease with load increase as the voids becomes shorter and more large frames can not be fitted within such voids (frame delineation) [16].

investigation is how to adapt $K_B$ or $V_{max}^i$ with void utilization and VBG traffic ratio to achieve a better performance with daily traffic variations through the access network.

Fig. 13: CCBVF delay reduction ratio ($D_i$=80-100 Km).

Fig. 15: Gate message proliferation , $D_i$=80-100 Km.

Gate message proliferation is shown in Fig.15. VE and NVF have identical gate message load. RBVF is much close to both VE and NVF. Both CCBVF and SCBVF have larger gate message load compared to the other schemes. The gate message load increases when more ONUs are assigned VBGs during voids. This explains why the gate message load increases with larger $K_B$ while decreases with larger $V_{max}^i$. In order to achieve a fair comparison, we compare both CCBVF and SCBVF with NVF at short range ($D_i$ = 20 Km) EPONs. It can be noticed that both CCBVF and SCBVF gate message load is far less than NVF in short range EPONs. Based on this note, we observe that the control message proliferation caused by CCBVF and SCBVF is considered acceptable price for the high average delay reduction ratio they achieve.

Fig. 16: Delay reduction ratio, $D_i$=95-100 Km.

Fig.16 shows the average delay reduction compared to NVF for $D_i$=95-100 Km. It shows that both RBVF and RBPVF are much affected by the reduced distance span compared to 20 km distance span presented in Fig.10 (b). However all PVT VBG sizing schemes did not face any performance degradation in terms of delay reduction ratio. Both VE and CCBVF achieves better delay reduction than the 20 km distance span. SCBVF is less sensitive to distance span as it
Fig. 17: 10 Gbps (a) Average delay, $D_i=80-100$ Km (b) Average delay reduction, $D_i=80-100$ Km.

Fig. 18: Average time between two consecutive grants.

Fig. 19: EF delay performance ($D_i=80-100$ Km) (a) Average delay (b) Average delay reduction.

is almost achieves the same reduction ratio. It can be inferred that PVT is strong competitor for both RBVF and RBPVF in both cases of long and short differential span distance.

Fig. 17 shows average delay comparison for symmetric 10 Gbps transmission capacity. Both RBVF and RBPVF outperform VE at loads below 0.4, however their performance quickly degrades with load increase. Both CCBVF and SCBVF continue their excel in terms of delay reduction. It is worth to note that in 10G-EPON, a mandatory Forward Error correction (FEC) parity codewords are inserted within the stream transmission [17]. This means that in order to transmit the maximum frame size (1518 bytes plus 20 bytes IFG and preamble), the ONU needs a transmission window of 1984 bytes. SCBVF achieves delay reduction up to 62%, while CCBVF achieves delay reduction up to 20%. Compared to 1 Gbps, both CCBVF and SCBVF achieve better performance. This was expected due to the expected increase in average void size.

B. DiffServ Traffic

Fig. 18 shows the time between consecutive bandwidth grants for NVF, RBVF, RBPVF, VE, CCBVF, and SCBVF. Both CCBVF and SCBVF have significant lower time period between consecutive bandwidth grants (between $0.2-0.5\text{ms}$), while it is between $(0.9-1\text{ms})$ for the other schemes. This means that both CCBVF and SCBVF are
able to utilize the voids more efficiently. The short time between grants helps ONU to flush the higher priority packets and delay sensitive traffic more frequently and thus reduces their average delay. It also helps the lower priority packets (BE) to be transmitted mainly in RBG and small portion of them during VBG.

Fig.19 shows EF traffic average delay performance. Both CCBVF and SCBVF achieves delay reduction up to 63% and 81% respectively. VE performance quickly degrades with load increase and shows negative delay reduction. This is because the uncontrolled VBG is more beneficial for lower priority traffic classes. It is also shown that reducing VBG size is more beneficial to EF traffic. While SCBVF EF performance improves with decreasing $V_{max}$, CCBVF performance improves with increasing $K_B$. It is also notable that SCBVF is less sensitive to HOLB as EF traffic packets are much smaller than $V_{max}$.

Both CCBVF and SCBVF switch their positions for the AF traffic performance as shown in Fig.20. CCBVF outperforms SCBVF because its relatively higher VBG size allows more AF traffic packets to be transmitted during VBG compared to SCBVF. CCBVF achieves delay reduction up to 52% compared to 45% achieved by SCBVF.

Regarding BE traffic (Fig.21), VE shows moderate improvement with delay reduction up to 10%. Both SCBVF and CCBVF continue their excel with delay reduction up to 52% and 56% respectively. It is important to emphasise that choosing large $K_B$ value might be beneficial to EF traffic but the performance quickly degrades in case of AF and BE classes.

It is also important to highlight that NVF, RBVF, RBPVF, and VE suffer from light load penalty phenomena [15], [16] for all traffic classes. The light load penalty phenomena is that the average delay decreases with the load increase at lower load range. As the load becomes moderate, this phenomena disappears. Both CCBVF and SCBVF are less affected by this phenomena for relatively lower VBG size.

**VIII. Conclusion**

This paper presents a novel void filling approach, called Parallel Void Thread (PVT) for Long-Reach Ethernet Passive Optical Networks (LR-EPONs). PVT is a polling thread independent from the running DBA thread(s). When the OLT detects a void succeeding a bandwidth grants, PVT is invoked to allocate void based grant(s) to fill that void. PVT delay reduction mechanism is to enable ONUs to early transmit part of their upstream traffic during VBGs. This technique reduces the grant delays and enables bandwidth grants more frequently than both RBVF and RBPVF. There are three proposed VBG sizing schemes for PVT namely VE, CCBVF, and SCBVF.
For single class LRD traffic, numerical results show that all of VE, CCBVF, and SCBVF outperform NVF, RBVF, and RBPVF and also achieve average delay below the \( 1.57V_{\max} \) bound. Regarding PVT, SCBVF is more efficient than CCBVF and VE but this comes with the expense of higher gate message load proliferation. Compared to SCBVF, CCBVF achieves less delay reduction but has very low gate message proliferation. VE has the least gate message proliferation as well as the least delay reduction. Regarding upstream report message proliferation, PVT does not add any additional control messages to the underlying DBA scheme. PVT performance mainly depends on how much traffic is transmitted during voids, the average delay of such traffic, and VBG utilization. Numerical results show that SCBVF has the highest VBG traffic ratio, while VE has the least one. For future research direction, \( K_B \) and \( V_{\max} \) can be adapted to those parameters to achieve optimum performance with continuous load variations.

For DiffServ traffic, both CCBVF and SCBVF were able reduce the time between consecutive grants to \( \approx 0.2 \) ms compared to \( \approx 0.2 \) for the other schemes. This has a great impact on delay reduction for EF, AF, and BE traffic classes respectively. The numerical results show that the smaller the VBG size is, the lower EF delay SCBVF and CCBVF achieve. On the other hand, reducing VBG size increases both AF and BE traffic delay. Numerical results show that SCBVF achieves delay reduction up to 80%, 45%, and 52% for EF, AF, and BE traffic classes. CCBVF achieves delay reduction up to 62%, 52%, and 57% for EF, AF, and BE traffic classes. VE achieves delay reduction up to 3%, 5%, and 11% for for EF, AF, and BE traffic classes respectively.

PVT can be applied without any modification to hybrid time/wavelength division multiplexing PONs (TWDM-PONs) for static wavelength allocation. Applying PVT for TWDM-PONs with dynamic wavelength allocation must be tied with the wavelength allocation algorithm and is left as a future research direction.

**References**


Amr Elrasad received his B.Sc. in Electrical Engineering and M.Sc. in Engineering Mathematics from Alexandria university, Egypt in 2004 and 2010, respectively. Currently, he is pursuing his Ph.D. at King Abdullah University of Science and Technology (KAUST), Saudi Arabia. Prior to KAUST, he was assistant lecturer at Faculty of Engineering, Alexandria University, Egypt. His research interests include optical networks, wireless networks and queuing systems.

Basem Shihada is an Assistant Professor of Computer Science and by courtesy Electrical Engineering at KAUST. He received his PhD from University of Waterloo in 2007. He joined Stanford University as a visiting faculty in Stanford Computer Science in 2008. He has been actively working on networking systems. Particularly interested in identifying the performance patterns for existing wireless network infrastructures and rebuild them to enhance their energy consumption, resource allocation, and reduce the associated performance delays. His current research covers a wide range of topics in broadband wired and wireless communication networks, including multi-hop, sensor, and cognitive networks. He is also interested in fiber-wireless network integration, optical networks, and green communication protocols.