

Article

Efficiency Tests of DBA Algorithms in XG-PON

Pavel Sikora ¹, Tomas Horvath ^{1,2}, Petr Munster ^{1,2} and Vaclav Oujezsky ¹¹ Department of Telecommunication, Brno University of Technology, Technicka 12, 616 00 Brno, Czech Republic² Department of Optical Networks, CESNET a.l.e., Zikova 4, 160 00 Prague, Czech Republic

* Correspondence: xsikor14@stud.feec.vutbr.cz; Tel.: +420-541-146-944

Received: 27 April 2019; Accepted: 4 July 2019; Published: 8 July 2019



Abstract: Sophisticated dynamic bandwidth allocation (DBA) algorithms can dramatically improve or worsen throughput and delay in whole networks. It is very important to choose the right DBA algorithm. Our work tests static assignment and three DBA algorithms, namely GigaPON Access Network DBA, Hybrid Reporting Allocation, and modified Max–Min Fair. All tests were made on our simulator of ten-gigabit passive optical network DBA specially developed for testing DBA algorithms. The tests verify delay of each optical network unit and amount of waste with bandwidth. This paper describes how the used DBA algorithms work and the processes involved in DBA algorithms.

Keywords: DBA; GIANT; HYRA; Max–Min; PON; XG-PON; assignment; simulation

1. Introduction

Passive optical network (PON) provides the advantage of having no active elements in its optical path [1,2]. Moreover, it offers a possibility of using a single optical fiber for all end units, even for bidirectional communication. The network consists of a central unit called optical line terminal (OLT) that is owned by the internet service provider (ISP), and other units connected to it called optical network unit (ONU) that are located with customers. The only other elements used in an optical network, called optical distribution network (ODN), are the so-called splitters, which are passive optical hubs that make it possible to connect several ONUs to a single optical fiber. Such a network is then referred to as point-to-multipoint (P2MP) network; see Figure 1 [3,4].

Passive optical networks (PONs) cannot work without any sophisticated dynamic bandwidth allocation (DBA) algorithm. OLT would implement the undemanding algorithm as a static assignment. OLT thus allocates bandwidth only when an ONU is connected to the ODN. Bandwidth amount would be split among between all connected ONUs [5,6]. Presently, network flows are very unstable and static assignment is thus completely unacceptable in unstable networks.

XG-PON network expects implementation of DBA. XG-PON defines two methods for the detection for the current ONU load. One is called status reporting (SR) and the other is called traffic monitoring (TM) [7].

DBA algorithms have quite accurate information about ONUs load. Introduced DBA algorithms used in our approach are based on this information. With sophisticated DBA algorithms an XG-PON network can quickly change bandwidth in very unstable networks.

The main contribution of this paper is testing DBA algorithms, which are currently available. We have developed our XG-PON simulator specially for testing DBA algorithms performance. In the simulator we implemented the static assignment, and three DBA algorithms, namely hybrid reporting allocation

(HYRA) [8], gigaPON access network (GIANT) [9], and modified Max–Min [10]. Algorithms were tested for transmission delay and amount of bandwidth waste.

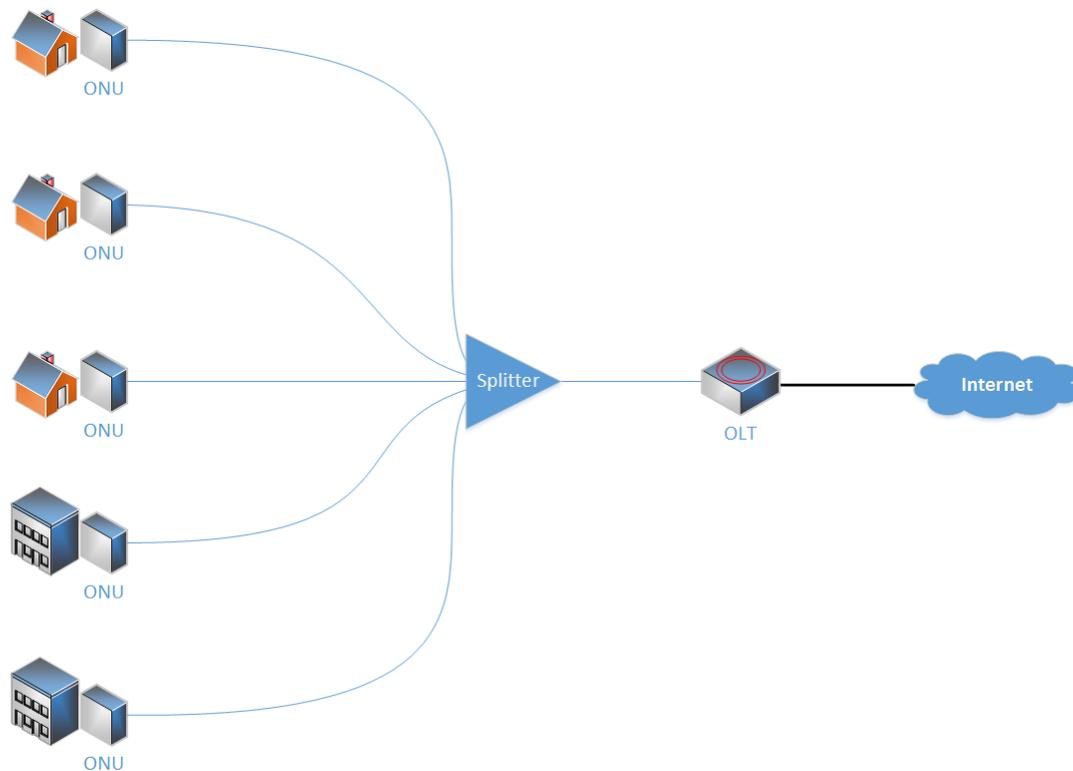


Figure 1. PON topology.

The rest of this paper is structured as follows. Section 2 provides an overview of the related works. Section 3 introduces the selected and implemented algorithms from the theoretical point of the view. Section 4 presents our simulator and simulation results. Finally, Section 5 concludes the paper.

2. Related Work

Assuming a continued annual relative growth ranging from 5.6 to 6.9%, ICT's relative contribution would exceed 14% of the 2016-level worldwide global greenhouse gas emissions (GHGE) by 2040 [11]. A present phenomenon is energy efficiency during sleep cycles and bandwidth allocation for ONUs in Institute of Electrical and Electronics Engineers (IEEE) or International Telecommunication Union (ITU) PONs [12–14].

In [15] the demand forecasting DBA algorithm for reducing packet delay is discussed. The propagation delay between OLT and ONU is about 0.3 ms which contains propagation delay, processing time and whole windows of the OLT unit of one frame. Current DBA algorithms rely on the request-grant cycle. The authors proposed a new DBA algorithm based on demand forecasting with the prediction of ONU future demand by statistical modeling of the demand patterns.

The article [16] deals with the next-generation Ethernet passive optical network (NG-EPON). These networks use optical distribution networks with wavelength sharing by ONUs; however, there are two different design architectures for NG-EPONs—the single scheduling domain (SSD) and multi-scheduling domain (MSD) ethernet passive optical network (EPON). Their algorithm works up to 10 Gbit/s but

the NG-EPON will use more than 10 Gbit/s in final specification such as 2×50 Gbit/s or 4×25 Gbit/s. The proposed algorithm also fits the current 10 Gbit/s PONs.

The future of passive optical networks is in the multi-OLTs platform and it will be necessary to deal with a novel DBA algorithm for the downstream channel. The authors proposed weight factors of bandwidth demand of different users [17]. They also proposed a bandwidth-transfer process between OLTs in ODN and they reduced the delay of the system while massively improving the system throughput.

The article [18] introduces a modular dynamic bandwidth allocation for flexible PON. This concept should be adopted for the virtual OLT hardware abstraction (VOLTHA) [19]. The VOLTHA is an open source project to create a hardware abstraction for broadband access equipment.

We also dealt with DBA algorithms in next-generation passive optical network stage 2 (NG-PON2) networks [20,21]. First, we introduced an implementation of the transmission convergence (TC) layer because the DBA uses the TC layer for allocation structures with grant size, start time, stop time, and other fields. The second article dealt with static bandwidth allocation in comparison with dynamic bandwidth algorithms. Furthermore, we proposed the modification for low load ONU units (for example during midnight or other scenarios).

The passive optical network can support bandwidth-intensive as well as low-latency services for 5th generation mobile networks by using channel bonding and low-latency oriented DBA [22]. Next application of time and wavelength division multiplexed PON (TWDM-PON) is in front hauls streams in the centralized radio access network (C-RAN) architecture of mobile networks [23,24].

3. DBA Performance Requirements

In practice, an OLT DBA algorithm does not know the entire state of the system in any specific moment. The DBA works with bandwidth estimates received from dynamic bandwidth report upstream (DBRu) reports, and data-flow monitoring results [7].

3.1. GIANT (GigaPON Access Network DBA)

The first implementation for gigabit capable passive optical networks (GPON) was published in 2006 [9]. This algorithm uses the parameters service interval (SI), and allocated bytes (AB). To increase the grant allocation period, the algorithm was extended in 2008 by immediate allocation (IA) that uses an additional parameter for the available byte counter, labeled as available byte counter (VB) [9].

In 2013, the GIANT algorithm was further improved by using the efficient bandwidth use (EBU) algorithm enabling IA to have negative values of the VB parameter. With this change it is possible to allocate the excess bandwidth to the same transmission container (T-CONT) type. Furthermore, it comes with a new parameter S_k [9], where:

$$S_k = \sum_{j \in Z} VB(j), \quad (1)$$

$$\text{with } Z = \{i \mid VB(i) > 0, SI(i) = 0, \text{queue}(i) \in TCONT \text{ type } k\}, \quad (2)$$

where the S_k parameter is recalculated at the end of each allocation cycle for each T-CONT in each ONU, the EBU inserts a high overhead for a large number of ONUs supported in the XG-PON. In addition, the EBU grant allocation period was further increased compared to IA. In practice, it was shown that the difference in speed is approximately ten-fold [9].

Description of Algorithm

This name was formed to capture the fact that it is a modification of the GIANT algorithm for XG-PON. X-GIANT maps transmission containers (T-CONTs) to quality of service (QoS) bandwidth distribution as follows [9]:

- **T-CONT 1:** Fixed bandwidth (highest priority).
- **T-CONT 2:** Assured bandwidth.
- **T-CONT 3:** Assured and Non-assured bandwidths.
- **T-CONT 4:** Best-effort bandwidth (lowest priority).

To compute the T-CONT 1 and T-CONT 2 bandwidths, it uses allocated bytes (ABmin) and service interval (SiMax). It also uses these parameters to set-up the polling times for T-CONT 3 and T-CONT 4. To compute allocated bytes for T-CONT 3 and T-CONT 4, AB surplus (ABsur) and SImin are used [9].

In reality, SI is essentially a downward counter. Every T-CONT has its own SI. Its initial value is set to SiMax and it is subtracted by one at the end of each allocation cycle. When it expires, allocated bytes for the corresponding T-CONT are computed. Until it expires, no bytes are allocated for that particular T-CONT. Using simulation, the authors of [9] found setting $SiMax = 1$, i.e., to a single allocation cycle to be the best option.

The calculation algorithm uses these four additional parameters [9]:

- **GIR:** Guaranteed Information Rates.
- **PIR:** Peak Information Rates.
- **GBS:** Guaranteed Burst Size.
- **PBS:** Peak Burst Size.

Where surplus rate (SR) is equal to $PIR - GIR$. The algorithm itself was adapted for the function deployed in the tester (to be described later). The abbreviation surplus rate is used only in the presented calculations. In the text, this abbreviation is used as status reporting, as it was mentioned in Chapter 2 [9].

At first, all allocated grants are reset. Subsequently, the algorithm goes through all the Alloc-IDs to determine whether their SI counters have expired or not. When a counter expires, the algorithm determines the type of T-CONT (1, 2, 3 or 4), and based on the type, it computes the allocated bytes according to Table 1 for the first cycle. It keeps the allocated bytes in a variable and continues with the second allocation cycle. The second cycle adjusts the values of the allocated bytes stored in the variable for T-CONT 3 and 4. This completes the calculation process. Another task is to create a bandwidth map (BWmap) from the allocated bytes and to subtract one from the counters of T-CONTs that did not expire the SI counter. Generating the BWmap is the last step of the algorithm [9].

Table 1. Description of the parameters used to compute AB [9].

| T-CONT | Allocated Bytes Calculation |
|--------------------------------|---|
| 1 | $ABmin = PIR \times SiMax$ |
| 2 | $ABmin = \min \{ \text{required bandwidth}, PIR \} \times SiMax$ |
| First allocation cycle | |
| 3 | $ABmin = \min \{ \min \{ \text{required bandwidth}, GIR \} \times SiMax, GBS \}$ |
| 4 | $ABmin = DBRu \text{ length}$ |
| Second allocation cycle | |
| 3 | $ABsur = \min \{ \min \{ \text{required bandwidth}, SR \} \times SImin, PBS - GBS \}$ |
| 4 | $ABmin = \min \{ \min \{ \text{available bandwidth}, PIR \} \times SiMax, PBS \}$ |

3.2. HYbrid Reporting Allocation

HYRA or hybrid reporting allocation is a DBA algorithm that uses both network load detection methods for ONUs. However, SR is still used as the main method. It favors more demanding ONUs and allocates more bandwidth to them. As its creators demonstrated in their simulations, HYRA can adapt to a variety of changes in the network, and represents a significant improvement in the upstream packet delays. In all situations, this algorithm reduced the resulting delay of the upstream packets compared with an algorithm using the pure SR method [8].

The HYRA algorithm allocates resources according to the principles mentioned in the DBA reference model, i.e., three types of bandwidth are assigned to each ONU:

1. Fixed bandwidth
2. Assured bandwidth.
3. Maximum bandwidth.

With this algorithm, the authors try to solve the problem of too much bandwidth being assigned to unused ONUs resulting in no free bandwidth that the DBA algorithm could assign to other loaded ONUs. This problem is mainly solved by introducing a simple, adaptive learning machine called learning automata (LA) [8].

LA is a finite state machine, based on artificial intelligence, designed to deal with unknown networks and networks with frequently changing parameters as frequently changing network parameters can dramatically affect the performance of the network [8].

Description of the Algorithm

The algorithm works in three phases. The first phase is the status reporting. The algorithm remains in this phase for as long as it receives any data frame. Once the XG-PON encapsulation method (XGEM) idle frame is received, it goes immediately into the traffic monitoring phase and waits for the data frame to be received. Upon this moment, it detects the most likely ONU unit's pause time and recomputes its probability of silencing. Subsequently, the ONU unit is silenced for the specified amount of time. After the pause time expires, it goes back to status reporting [8].

The algorithm itself works with several values. The first is A that represents the ONU's silent time, i.e., time during which the DBA algorithm will not allocate any grants. It can range from zero to 400 indicating the number of algorithm calculations excluding the ONU unit [8].

Another one is $P^i(f)$.

$$P^i(f) = p_0^i(f), p_1^i(f), p_2^i(f), \dots, p_{400}^i(f), \quad (3)$$

where i represents the given ONU unit. This probabilistic vector contains 401 elements. Each value shows the probability of one of the 401 situations occurring. For example, the value of $p_{60}^i(f)$ shows the probability of setting the value of A to 60 and thus silencing the ONU unit for 60 calculations. BWmap is calculated every 125 μ s. By simply multiplying the silent value with the BWmap calculation interval $60 \cdot 0.125$, it can be shown that the ONU unit will remain silent for 7.5 ms if the value of 60 is selected [8]. Before setting the probabilities during run-time, it is necessary to determine the value of a_k . This value represents for how long the ONU unit was sending the idle XGEM frames. This value is expressed as follows [8]:

$$a_k = \left\lceil \frac{T2 - T1}{125} \right\rceil \quad 0 \leq k \leq 401, \quad (4)$$

where $T1$ represents the time of receiving the first idle XGEM frame. It retains this value and waits for the time $T2$ when a data frame is received. When it receives this data frame, it computes a_k (an action (A) from zero to 400) [8].

The next step is to determine the current most likely silencing value and to assign this value to A . It can then go into the probability conversion phase. First, the probability of the previous action a_k is computed [8]:

$$p_k^i(f + 1) = p_k^i(f) + \sum_{j=0, j \neq k}^{400} L(p_j^i(f) - a), \tag{5}$$

where i denotes the ONU unit, k corresponds to the action of the feedback loop, and L represents the convergence rate. In addition, a relatively small number is also used to prevent zero-value probabilities. As can be seen, the probability increases by a small portion that is subtracted from the other probabilities with the help of [8]:

$$p_j^i(f + 1) = p_j^i(f) - L(p_j^i(f) - a) \forall j \neq k, 0 \leq j \leq 400. \tag{6}$$

3.3. Modified Max–Min Fair DBA

Modification of the Max–Min Fair DBA is designed for XG–PON networks. The modification originated from the Max–Min Fair DBA algorithm, which was developed for G–PON networks [10]. Implementation of this algorithm without changes specific for the XG–PON network caused long delays. For this reason, the Modified Max–Min Fair DBA was created. Algorithms such as HYRA combine both methods of gaining occupancy of ONU units. One of the advantages of such a combination is the fact that if an ONU unit does not send the cluster with the required queue occupancy information for the SR method, there is an inconsistency of the information on the OLT-side about the occupancy of the queues. By using the non-status reporting method, the inconsistency can be avoided [10].

The algorithm emphasizes three following rules [10]:

1. Bandwidth is allocated with increasing demand.
2. ONU is never assigned a larger *GrantSize* than a queue occupancy report.
3. ONU units that cannot be fully served will receive the same remaining bandwidth.

Description of the Algorithm

In the first phase, sorting of the bandwidth demands per ONUs R_t^n is initialized. It sorts these demands in the increasing order into the list S . *GrantSize* W_t^n is set-up to zero [10].

The first step in the calculation of the allocation of bandwidth is to compute B_L , i.e., the available bandwidth is computed as follows [10]:

$$B_L = \frac{bandwidth - \sum_{k=1}^m W_t^k}{|S|}, \tag{7}$$

where *bandwidth* represents the sum of all clusters in the upstream direction, m is the total number of ONU units, and $|S|$ is equal to the number of elements in S [10].

The next step is to restore the *GrantSize* W_t^n parameter for all ONU units in S as [10]:

$$W_t^k = \min\{R_t^k, W_t^k + B_L.\} \tag{8}$$

The third step removes all fully serviced ONUs from S , and therefore [10]:

$$W_t^k = R_t^k. \tag{9}$$

The fourth and the last step determines whether there are any ONUs in $|S|$. If there is at least a single ONU unit, the entire procedure from the first step is repeated until $|S|$ is finally empty [10].

This procedure takes into account the current occupancy values of all individual ONU units. However, if an ONU does not send the occupancy report of its queue C_t^n , the data gets out of date. In this case, the algorithm introduces a queue occupancy estimate named as virtual demand for capacity R_t^n , which can be computed in two ways. The first way is to compute the virtual demand from previous assigned GrantSize [10]:

$$R_t^n = \begin{cases} C_t^n & \text{if } C_t^n \neq \text{null and } C_t^n > 0, \\ \frac{\sum_{i=1}^{t-1} C_i^n}{t-1} & \text{otherwise.} \end{cases} \quad (10)$$

The second way is to compute the virtual demand R_t^n from the previous ONU [10]:

$$R_t^n = \begin{cases} C_t^n & \text{if } C_t^n \neq \text{null and } C_t^n > 0, \\ \frac{\sum_{i=1}^{t-1} W_i^n}{t-1} & \text{otherwise.} \end{cases} \quad (11)$$

Once the bandwidth allocation for each ONU unit is completed, it is necessary to specify *StartTime* for all clusters in the upstream direction. The algorithm assigns *StartTime* to ONU units in the ascending order according to their distance from the OLT unit. This allocation is omitted in the simulator because the designed simulator neglects these distances meaning that the same value is used in all cases.

4. Simulation Description and Simulation Results Discussion

We use C++ language for development of our XG-PON simulator for testing DBA algorithms. This programming language allows the writing of object-oriented code, so we used this option and whole simulator is thus object-oriented. Every frame flowing through network or every ONU connected to ODN of XG-PON is an object made from one class for frame and one class for ONU. In object-oriented programming language it is much simpler to add additional DBA algorithms, which is also a big plus.

The simulator does not need full implementation for testing DBA algorithms. Functions such as encryption of each frame transmitted through ODN are unnecessary and their implementation would only take up computational resources such as processor time and random access memory (RAM). As some algorithms how to treat all T-CONT are not clearly defined, we decided the following. T-CONT 1 does not need any dynamic allocation, because it has fixed bandwidth. T-CONT 3 and 4 have lower QoS priority. So we decided to focus on T-CONT 2, and generate traffic only on this T-CONT for saving computing resources (central processor unit (CPU) i5-3230M 2.60 GHz, 8 GB random access memory (RAM) double data rate (DDR3), 256 GB solid state drive). This simulator also ignores different distances of ONUs from the OLT. Simulator thus places all ONUs to the same distance [7]. A real passive optical network (in general point of view not only XG-PON or older/newer) uses an activation process of each ONU [25]. During this phase, the ONUs have to be aligned to the same distance from the optical line termination (OLT) point of view. It leads to assigning a unique value of the equalization delay. The ONUs seem at the same distance (virtually) of the OLT, which allows the OLT to control the upstream by a time slot duration for each ONU. The propagation delay between OLT and ONU (up to 20 km) will be around picoseconds, but the OLT has hundreds of microseconds to process all traffic. This phenomenon must be discussed for long-reach passive optical networks.

Nevertheless, all functions can be skipped in the implementation. At first glance on service data unit (SDU) fragmentation, someone might skip this feature. However, that would cause big problem in some situations. When DBA allocates smaller amounts of bandwidth than the size of SDU in buffer, ONU cannot send this SDU and it must wait for higher allocation.

The simulator of XG-PON is controlled through command line. Parameters such as network load from ONU, used DBA, simulation time, and number of connected ONUs to ODN are set through command

line. After the program finishes a simulation, it generates all results and events during simulation to text files. ONUs have one function in addition to the physical ONU. This feature represents a complete client devices simulation. It is unnecessary to simulate connected client devices to ONU, such as computers, smart phones, etc. Traffic is generated directly on ONU device as if it were a client device [7].

Simulation starts from the point when all ONUs are properly connected to ODN, and they fully communicate with the OLT. So, the simulator skips all activation steps and starts in the Operating state (O5) [7].

Simulation was run for every DBA algorithm separately. All common simulation parameters were set equally. Traffic flow generated in the simulations was 1 kB/s per ONU. So, every ONU generates every second 1024 B SDU. SDUs go directly to buffers and wait. This is not the correct value which is transmitted through ODN. To this we must add the overhead from XGEM and XG-PON transmission convergence layer (XGTC) frame, to which the SDU is encapsulated. In simulation we observe a lot of fragmentation, so this process adds some overhead data too.

Not all algorithms used in our simulation clearly describe how to allocate non-assured bandwidth and best-effort bandwidth. So, we decided not to simulate this bandwidth. We focus on allocation-identifier (Alloc-ID) 2, because this Alloc-ID uses only Assured bandwidth which all algorithms describe clearly.

4.1. Static Assignment

DBA algorithm in this simulation was set to static assignment. This pseudo DBA algorithm only assigns grants at the begin of the simulation. After that the algorithm does not make any changes in allocated bandwidth. For 10 ONUs the algorithm allocates 242 grants for all Alloc-IDs on every connected ODN.

When we look at Figure 2 at first we see a peak of 435.442 μ s. This is caused by waiting for the first allocation. After the OLT allocates bandwidth, the delay fluctuates between 250 μ s and 350 μ s. Fluctuations are caused by waiting time of SDUs in the buffer. ONU must wait for its allocated time slot. To this delay we must add the delay in the OLT buffer. When OLT processes an arrived SDU, OLT puts it to the downstream buffer. In the downstream buffer the SDU must wait until the next downstream XGTC frame is created. This process could take maximally 125 μ s.

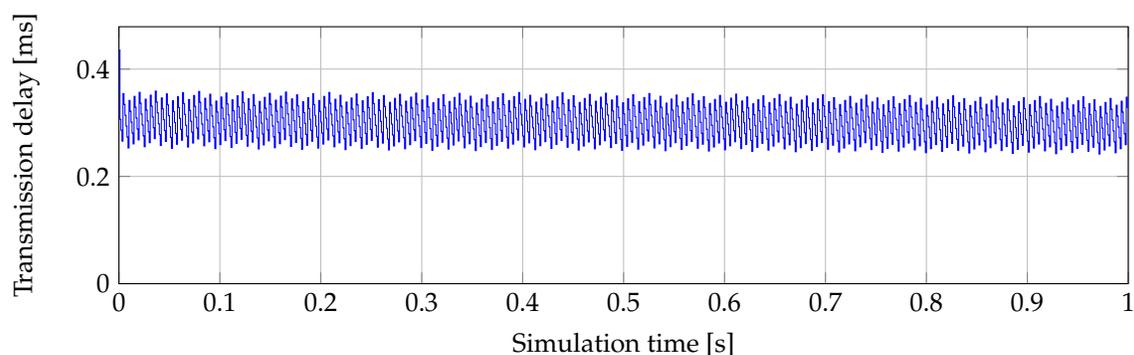


Figure 2. Transmission delay for second ONU with static assignment.

Figure 3 shows time and size of idle XGEM frames generated by the second ONU. It is clear for static assignment that the value fluctuates between 894 B and 962 B. Value 962 B represents the situation when the ONUs upstream buffer is empty and the ONU does not have data to send. ONU cannot transmit whole SDU with size 1024 B if its bandwidth allocation is smaller than 1024 B, so it must use fragmentation. The ONU first sends upstream XGTC frame filled with XGEM frame, which contains SDU of maximal size that was allocated to the ONU. After that, the ONU processes the second fragment of SDU and fills remaining

bandwidth with an idle frame with length of 894 B. Static assignment lead to generation of many idle frames. When we counted it, it was 69 idle frames generated in 1 ms of simulation. In whole 1 s simulation all ONUs generate 6.67 MB—see Table 2. Assignment of DBA was not changed during simulation, so all ONUs generate the same value. These values are the worst, respectively the highest of all algorithms tested in our simulations. When we look at Table 3, we find out that with static assignment transmission the delay through ODN is the smallest, when we compare the value with real DBA algorithms, because ONUs do not wait for assignment sufficient bandwidth and ONUs can send SDUs immediately to OLT. This works only in case that traffic load of that ONU is not higher than the allocated bandwidth.

Table 2. Sum of whole size of idle frames generated in 1 s of simulation for every ONU.

| ONU | HYRA [MB] | Static Assignment [MB] | Max–Min [MB] | X-GIANT [MB] |
|-----|-----------|------------------------|--------------|--------------|
| 1 | 0.286282 | 6.665038 | 0.450252 | 3.104062 |
| 2 | 0.395916 | 6.665038 | 0.773320 | 3.095722 |
| 3 | 0.396510 | 6.665038 | 0.802048 | 3.073150 |
| 4 | 0.397104 | 6.665038 | 0.807792 | 3.063060 |
| 5 | 0.400668 | 6.665038 | 0.724032 | 3.096292 |
| 6 | 0.403044 | 6.665038 | 1.125 | 3.117054 |
| 7 | 0.404232 | 6.665038 | 0.855466 | 3.102806 |
| 8 | 0.405420 | 6.665038 | 0.779574 | 3.094514 |
| 9 | 0.414330 | 6.665038 | 0.725078 | 3.124804 |
| 10 | 0.425022 | 6.665038 | 0.799454 | 3.120646 |

Table 3. Average delays.

| ONU | HYRA [ms] | Static Assignment [ms] | Max–Min [ms] | X-GIANT [ms] |
|-----|-----------|------------------------|--------------|--------------|
| 1 | 0.907451 | 0.382307 | 109.244 | 0.434343 |
| 2 | 0.677102 | 0.30088 | 87.8914 | 0.386335 |
| 3 | 0.679279 | 0.313181 | 33.3388 | 0.345512 |
| 4 | 0.681565 | 0.3257 | 76.3823 | 0.347145 |
| 5 | 0.684722 | 0.338001 | 28.3254 | 0.351173 |
| 6 | 0.687008 | 0.350302 | 30.4899 | 0.354983 |
| 7 | 0.689403 | 0.36293 | 69.4605 | 0.362604 |
| 8 | 0.69256 | 0.375231 | 76.8177 | 0.372292 |
| 9 | 0.693649 | 0.387859 | 64.6643 | 0.379042 |
| 10 | 0.694193 | 0.400487 | 74.8914 | 0.387097 |

This pseudo DBA algorithm is useful in real deployment only in special cases, such as when ONUs have same traffic flow at all time while traffic load of ODN is not higher than maximum capacity of XG-PON. In other cases, this algorithm is not useful, because it wastes a lot of bandwidth. This wastes of bandwidth can be eliminated through real DBA algorithms. There is also a case when load of an ONU is higher than allocated bandwidth, and other ONUs are not fully loaded. In that case the loaded ONU delayed SDUs. When a network uses a real DBA algorithm, it can allocate higher bandwidth for certain time by reduction of bandwidth allocated to no loaded ONUs. Loaded ONU thus gains an opportunity to faster emptying the buffer and not to delay SDUs. Then the DBA algorithm shrinks bandwidth and other ONUs have an opportunity to get more bandwidth to allocate.

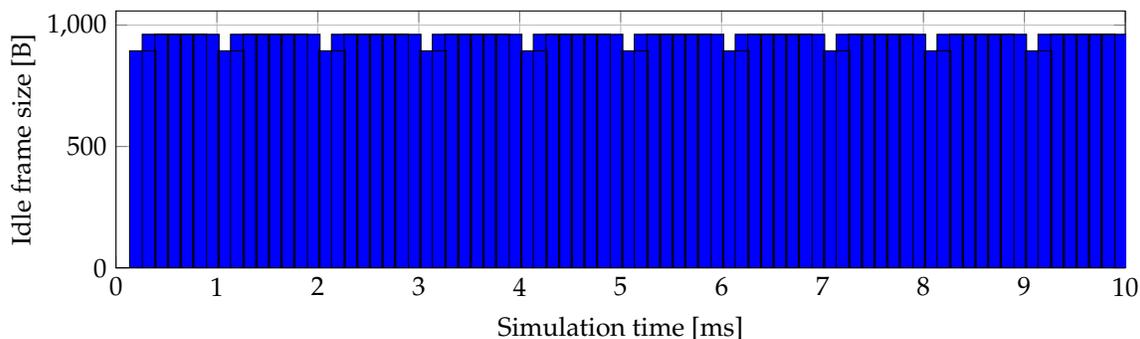


Figure 3. Idle frames generated by second ONU with static assignment.

4.2. Max–Min DBA

In our implementation of this algorithm, we decided to set the fixed bandwidth to three grants. Fixed bandwidth is allocated to Alloc-ID 1 of all ONUs regardless the current network load to Alloc-ID one [10]. We implement the first type, which computes the value from previously allocated grants. The authors of this algorithm claim the Max–Min DBA algorithm is hybrid, so it uses SR and TM methods for detection of the current buffer occupancy of every ONU. However, they do not describe how. So our implementation uses only the TM method.

Our implementation of the Max–Min DBA algorithm provides the worst average transmission delay compared with all tested DBAs. Table 3 clearly shows that the value of 109 ms is the maximum average delay. This value was measured on first ONU. The lowest average delay of 28 ms was made by the fifth ONU. This delay is also the worst if we compare this highest value with all highest delays measured in all tested DBA algorithms. Figure 4 shows the transmission delay for the second ONU, where peaks of 1.85 ms are in 150 ms of simulation. This issue occurs in the whole 1 s simulation. Delays can reach 360 ms in maximal values. These values then make the worst average delay.

However, this algorithm is not the worst at all. It is the second-best DBA algorithm to save bandwidth as Table 2 shows. Sum of all idle XGEM frames generated by the sixth ONU is 1 MB, which is the highest value for that DBA algorithm. The lowest value generates the first ONU, where the sum is 0.45 MB. Second ONU generates 26 idle XGEM frames in 10 ms of simulation. Size of these frames is described in Figure 5. All ONUs generate similar number of IDLE frames of similar size in 10 ms of simulation. Average size of IDLE frames in whole simulation was between 0.45 and 1.125 MB, we decided to present the results of the second ONU which is in the middle of these Figures.

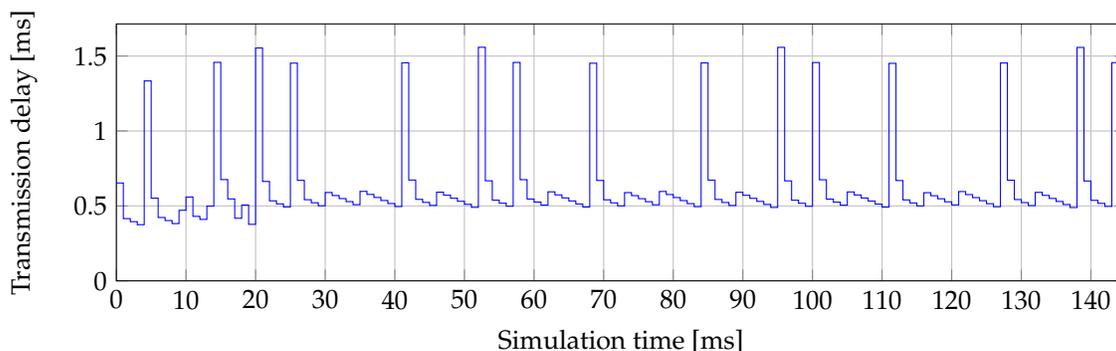


Figure 4. Transmission delay for second ONU with Max–Min DBA.

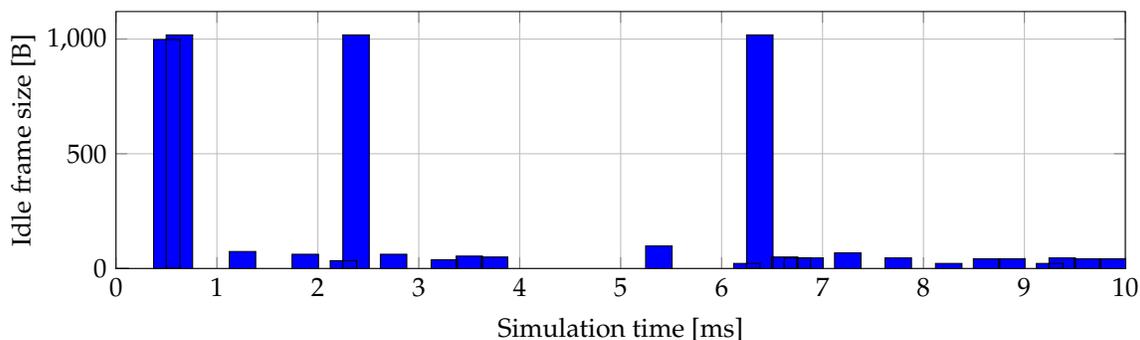


Figure 5. Idle frames generated by the second ONU with Max-Min DBA.

4.3. Hybrid Reporting Allocation

In our implementation of this algorithm, we decided to set the fixed bandwidth to six grants. Assured bandwidth was set as authors in the paper [8] did to 125 words. Max bandwidth was set to 150 words. Values L and a were same as the authors proposed in their paper, so L was set to 0.1 and a to 10^{-5} .

Figure 6 shows the SDU delay of transmission from the second ONU to OLT and back to ONU. The learning period of the algorithm, which takes less than 200 ms is clearly visible. After the algorithm learned the probabilities, the delay fluctuates between 600 μ s and 800 μ s with a tendency to decline. This algorithm makes the lowest jitter in tested real DBA algorithms, after the algorithm learns the probabilities, but a static assignment can generate the lowest jitter.

Grants allocated to the second ONU were very varied with the maximum value of 150 words, as can be seen in Figure 7. 150 words is the correct value because we set the maximum allocated bandwidth to 150 words. There are no values between 150 and one grant. This is also correct, because we generate 1024 B SDUs, so it is 250 words without the overhead. So, when an SDU arrived at the buffer, ONU sent its buffer occupancy to the OLT and HYRA has enough bandwidth to allocate the maximum bandwidth. When the ONUs buffer is empty, ONU starts sending idle XGEM bursts with the value of buffer occupancy equal to zero. HYRA decided to allocate the minimal bandwidth which is one word. One word is minimal because ONU needs to report periodically its buffer occupancy. If we set this value to zero, ONUs will lose opportunities to send any upstream XGTC bursts, so the OLT will decide that this ONU does not need any bandwidth, and thus the ONU will be forever without bandwidth.

Figure 7 shows the size and when the second ONU sends idle XGEM in 150 ms of simulation. It is clearly visible that firstly there are not sent any XGEM idle frames, because the whole XGTC frame was filled with an XGEM frame that the contains SDU with maximum value of allocated bandwidth. After that, the ONU sends a fragment of SDU and adds to the XGTC frame an idle XGEM frame with the size of 150 B. An idle XGEM frame is also sent when the ONU has empty buffer and the ONU has still allocated bandwidth from previous buffer occupancy. Total number of sent idle frames in 10 ms of simulation was 15.

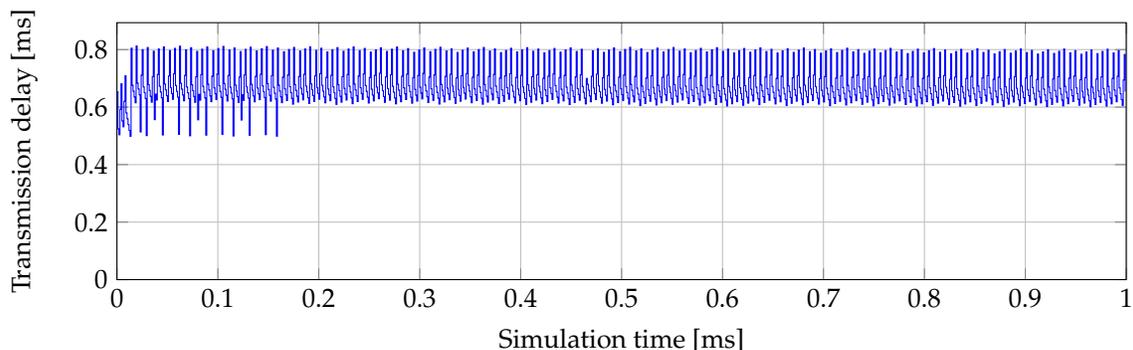


Figure 6. Transmission delay for second ONU with HYRA DBA.

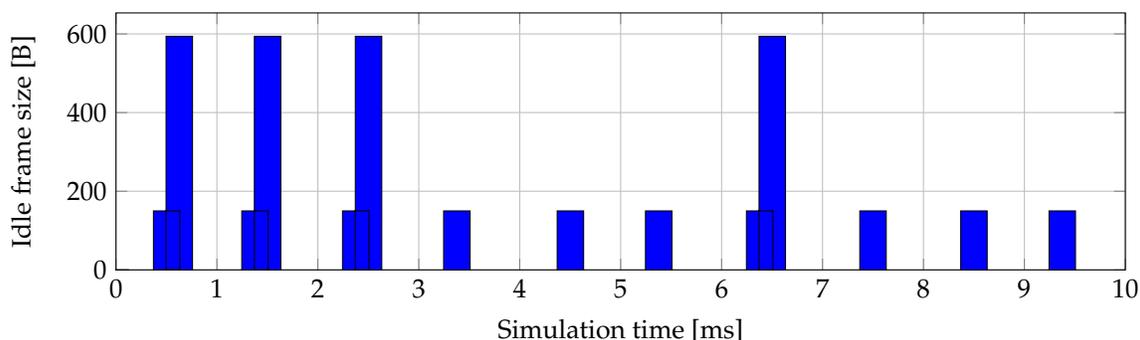


Figure 7. Idle frames generated by second ONU with HYRA DBA.

HYRA is the best choice for minimal waste of bandwidth in the selection of real DBAs. As described Table 2, the average size of generated ONUs is between 0.29 and 0.43 MB. It is not the best choice for minimal delay. Delay with the HYRA algorithm is between 0.68 and 0.9 ms, as described in Table 3.

4.4. X-GIANT DBA

The X-GIANT DBA algorithm works with 6 parameters. We used same values as the authors of the paper [9]. However, authors in [9] experimented with SiMax and SiMin. So, we decided to set the parameters as described in Table 4.

When we compared this algorithm with Max–Min we found that the delay of every ONU is much smaller. Comparing two maximal average transmission delays, we found out that both are by first ONU and values differ by 108.81 ms. Maximal average transmission delay of all algorithms was made by the Max–Min DBA algorithm. When we compare average transmission delay from second ONU with static assignment and again transmission delay of first ONU with Max–Min, then we found extremes. These values differ by 108.94 ms. All average values of delays with X-GIANT algorithm are summarized in Table 3.

Table 4. Parameters set for X-GIANT algorithm.

| Parameter | Value |
|-----------|----------------------|
| SiMax | 1 |
| SiMin | $2 \times SiMax = 2$ |
| PIR | 150 |
| GIR | 120 |
| PBS | 150 |
| GBS | 120 |

Average delay of the second ONU with algorithm X-GIANT in first 150 ms shows Table 3. Values are comparable to static assignment, except peaks. Peaks can reach 0.89 ms in the whole 1 s simulation. In first 150 ms peaks reach maximum value of 0.65 ms. Considering the average delay, the algorithm took the first place in real DBAs. When comparing it with static assignment, static assignment is still better. Maximal average value was measured on the first ONU, with the value of 0.43 ms. Minimal average delay was on the third ONU, with 0.35 ms. First 150 ms of simulation are shown in Figure 8.

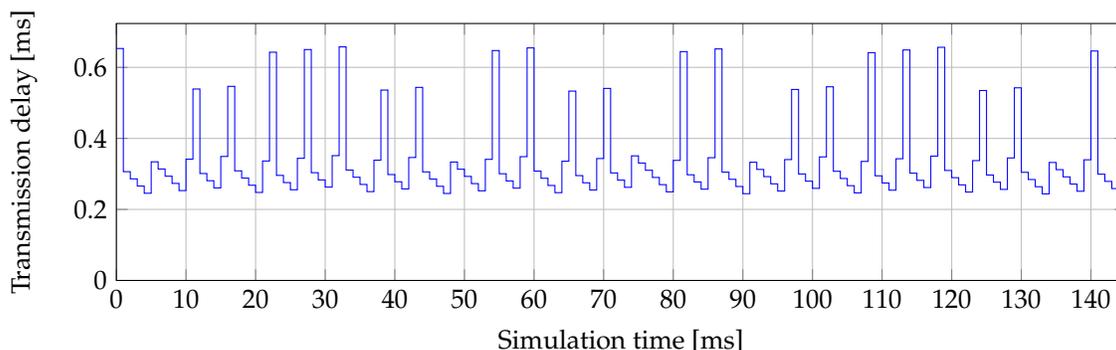


Figure 8. Transmission delay for second ONU with X-GIANT DBA.

Considering use of bandwidth, the algorithm is the worst of real DBAs. However, it is still better than static assignment, which took twice the value. The lowest sum of generated idle frames in whole simulations, was measured on the fourth ONU with the value 3.06 ms. The highest value was 3.12 ms, measured on the ninth ONU. All values are shown in Table 2. In 10 ms of simulation the second ONU transmits 66 idle XGEM frames, which is not the highest. The highest value generates static assignment. First 150 ms of simulation are shown in Figure 9.

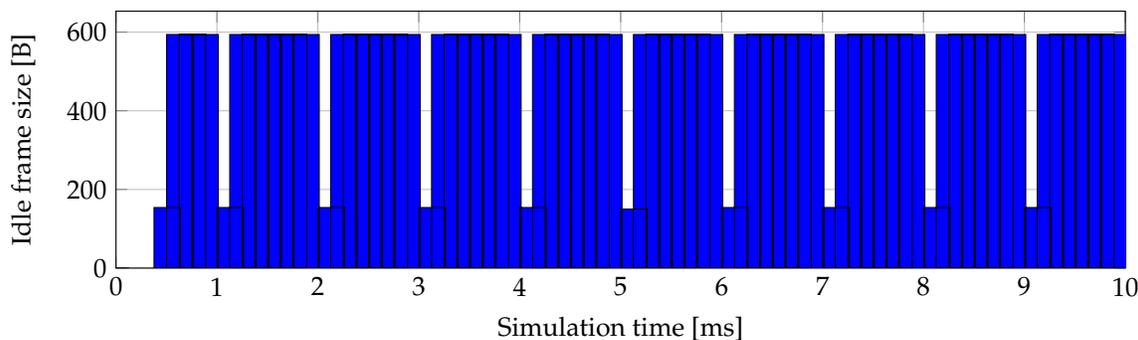


Figure 9. Idle frames generated by second ONU with X-GIANT DBA.

Figures 10 and 11 show transmission delay from second ONU in 1 s simulation. Max–Min DBA algorithm was separated, because algorithm generates a more significant transmission delay in comparison with other algorithms. It is clearly visible from Figures that X-GIANT generates the lowest average transmission delay, if we compare real DBA algorithms. The lowest values reach values same as static assignment. HYRA generated the second highest delay. This delay still does not make any problems, because the values were larger by about 400 μ s when compared with static assignment. The advantage of the HYRA DBA algorithm shows Figure 12, which shows generated idle frames and their size by second ONU in 10 ms of simulation. The advantage is that this algorithm generates small amount of unnecessary traffic flow.

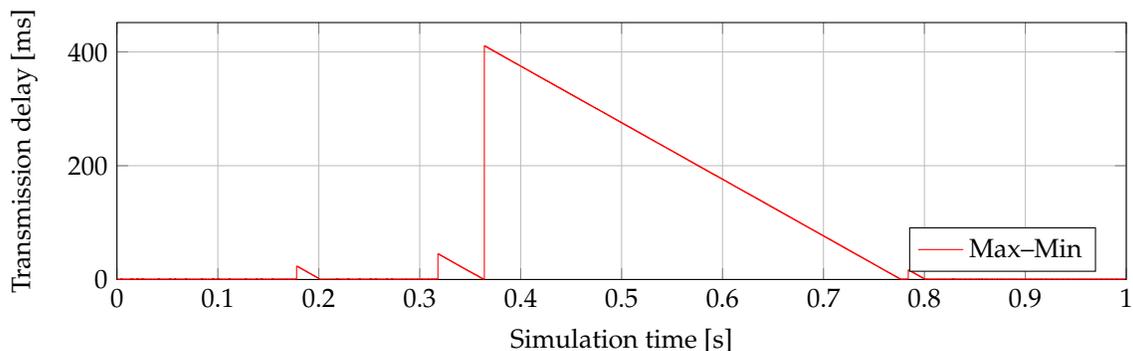


Figure 10. Transmission delay for second ONU with Max–Min DBA for 1 s simulation.

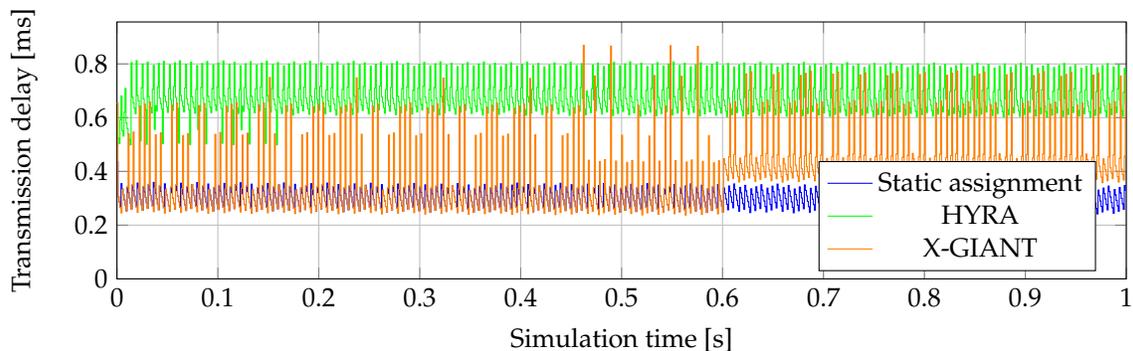


Figure 11. Transmission delay for second ONU in 1 s simulation.

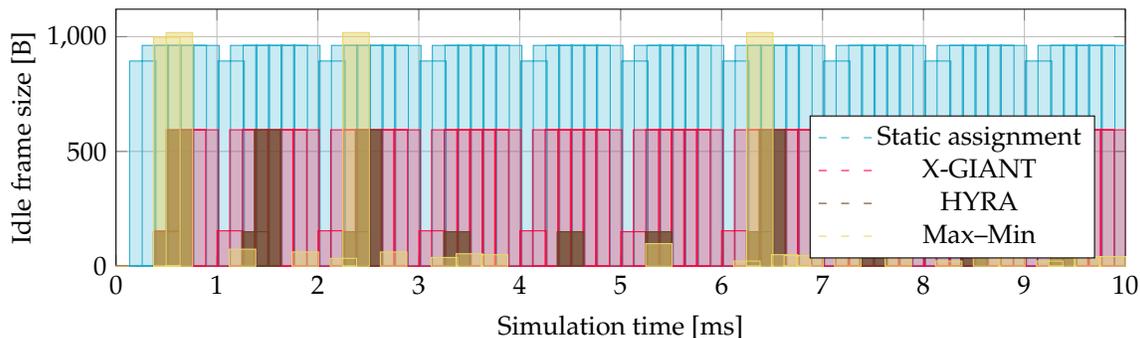


Figure 12. Idle frames generated by second ONU for 10 ms simulation.

5. Conclusions

Today's networks have very variable network flow. Without sophisticated DBA algorithms, XG-PON networks cannot achieve fast adjustment to current conditions. Static assignment has some advantages, as that ONUs have the lowest transmission delay, but only in some cases. When network load of at least one ONU is higher than the allocated bandwidth, for example in our case with 10 ONUs, 969 B for one allocation is sufficient to gradually overflow the buffer. XG-PON networks need a DBA algorithm which is fair, generates lowest delay on ODN and minimally wastes the bandwidth. In our simulation approach with tested DBAs, we found two options. Both two options are a compromise in higher delay or higher waste of bandwidth. The Max–Min DBA algorithm has lower waste of bandwidth, but transmission delay was very high. For it we decided not to choose this algorithm at all. The HYRA DBA algorithm is a proper choice when the major network requirement is the low bandwidth waste. The X-GIANT DBA algorithm is a proper choice when the major network requirement is the low delay.

The future work will be an implementation of our modification of the algorithm and improvement the current implementation with an activation process with tuning process for NG-PON2 networks.

Author Contributions: Conceptualization P.S. and T.H.; investigation P.S.; P.M.; V.O.; funding acquisition T.H.; P.M. and V.O.; software P.S.; simulations P.S.; methodology P.S.; T.H.; P.M., and V.O.; data curation T.H.; P.M. and V.O.; supervision T.H.; V.O.; validation P.M.; T.H.; V.O.; visualization P.S.; T.H.; P.M. and V.O.; writing—original draft P.S. and T.H.

Funding: The presented research has been supported by the projects of the Ministry of Interior under grant no. VI20172020110 “Reduction of security threats at optical networks” and “E-infrastructure CESNET—modernization”, registration no. CZ.02.1.01/0.0/0.0/16_013/0001797.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

| | |
|----------|---|
| AB | Allocated bytes |
| ABsur | AB surplus |
| Alloc-ID | Allocation-identifier |
| BWmap | Bandwidth map |
| C-RAN | Centralized radio access network |
| DBA | Dynamic bandwidth allocation |
| DBRu | Dynamic bandwidth report upstream |
| EBU | Efficient bandwidth use |
| EPON | Ethernet passive optical network |
| GBS | Guaranteed burst size |
| GIANT | GigaPON access network |
| GIR | Guaranteed information rates |
| GPON | Gigabit capable Passive Optical Networks |
| HYRA | Hybrid reporting allocation |
| IA | Immediate allocation |
| ICT | Information and communication technologies |
| IEEE | Institute of Electrical and Electronics Engineers |
| ISP | Internet service provider |
| ITU | International Telecommunication Union |
| LA | Learning automata |
| MSD | Multi-scheduling domain |

| | |
|---------|--|
| NG-EPON | Next-generation Ethernet passive optical network |
| NG-PON2 | Next-generation passive optical network stage 2 |
| ODN | Optical distribution network |
| OLT | Optical line terminal |
| ONU | Optical network unit |
| P2MP | Point-to-multipoint |
| PBS | Peak burst size |
| PIR | Peak information rates |
| PON | Passive optical network |
| QoS | Quality of service |
| RAM | Random access memory |
| SDU | Service data unit |
| SI | Service interval |
| SR | Status reporting |
| SR | Surplus rate |
| SSD | Single scheduling domain |
| TC | Transmission convergence |
| T-CONT | Transmission container |
| TM | Traffic monitoring |
| TWDM | Time and wavelength division multiplex |
| VB | Available byte counter |
| VOLTHA | Virtual OLT hardware abstraction |
| XGEM | XG-PON encapsulation method |
| XG-PON | Next-generation passive optical network |
| XGTC | XG-PON transmission convergence layer |

References

1. Harstead, E.; Van Veen, D.; Houtsma, V.; Dom, P. Technology Roadmap for Time Division Multiplexed Passive Optical Networks (TDM PONs). *J. Lightwave Technol.* **2019**, *37*, 657–664. [[CrossRef](#)]
2. Horvath, T.; Munster, P.; Vojtech, J. Deployment of PON in Europe and Deep Data Analysis of GPON. In *Telecommunication Systems [Working Title]*; InTech: London, UK, 2019; pp. 1–20.
3. Jay, S.; Neumann, K.-H.; Plückerbaum, T. Comparing FTTH access networks based on P2P and PMP fibre topologies. *Telecommun. Policy* **2014**, *38*, 415–425. [[CrossRef](#)]
4. Hernandez, J.A.; Sanchez, R.; Martin, I.; Larrabeiti, D. Meeting the Traffic Requirements of Residential Users in the Next Decade with Current FTTH Standards: How Much? How Long? *IEEE Commun. Mag.* **2019**, *57*, 120–125. [[CrossRef](#)]
5. Ford, G.S. Is faster better? Quantifying the relationship between broadband speed and economic growth. *Telecommun. Policy* **2018**, *42*, 766–777. [[CrossRef](#)]
6. Stocker, V.; Whalley, J. Speed isn't everything: A multi-criteria analysis of the broadband consumer experience in the UK. *Telecommun. Policy* **2018**, *42*, 1–14. [[CrossRef](#)]
7. G.987.3: 10-Gigabit-Capable Passive Optical Networks (XG-PON): Transmission Convergence (TC) Layer Specification. Available online: <https://www.itu.int/rec/T-REC-G.987.3/en> (accessed on 8 April 2019).
8. Sarigiannidis, P.; Papadimitriou, G.; Nicolaitidis, P.; Varvarigos, E.; Yiannopoulos, K. HYRA: An efficient hybrid reporting method for XG-PON upstream resource allocation. In Proceedings of the 2014 5th International Conference on Optical Communication Systems (OPTICS), Vienna, Austria, 28–30 August 2014; pp. 1–10.
9. Arokiam, J.A.; Brown, K.N.; Sreenan, C.J. Refining the GIANT dynamic bandwidth allocation mechanism for XG-PON. In Proceedings of the 2015 IEEE International Conference on Communications (ICC), London, UK, 8–12 June 2015; pp. 1006–1011.

10. Gravalos, I.; Yiannopoulos, K.; Papadimitriou, G.; Varvarigos, E. A. A modified max-min fair dynamic bandwidth allocation algorithm for XG-PONs. In Proceedings of the 2014 19th European Conference on Networks and Optical Communications—(NOC), Milano, Italy, 4–6 June 2014; pp. 57–62.
11. Belkhir, L.; Elmeligi, A. Assessing ICT global emissions footprint. *J. Clean. Prod.* **2018**, *177*, 448–463. [CrossRef]
12. Helmy, A.; Nayak, A. Towards More Dynamic Energy-Efficient Bandwidth Allocation in EPONs. In Proceedings of the 2018 IEEE Global Communications Conference (GLOBECOM), Abu Dhabi, UAE, 9–13 December 2018; pp. 1–6.
13. Zin, A.M.; Idrus, S.M.; Ismail, N.A.; Ramli, A.; Butt, R.A. Energy Efficient Performance Evaluation of XG-PON for Sustainable Green Communication Infrastructure. In Proceedings of the 2018 Progress in Electromagnetics Research Symposium (PIERS-Toyama), Toyama, Japan, 1–4 August 2018; pp. 950–955.
14. Garg, S.; Dixit, A. Models for Evaluating Power Saving Techniques in Flexible Optical Access Networks. In Proceedings of the 2018 20th International Conference on Transparent Optical Networks (ICTON), Bucharest, Romania, 1–5 July 2018; pp. 1–4.
15. Memon, K.; Mohammadani, K.; Ain, N.; Shaikh, A.; Ullah, S.; Zhang, Q.; Das, B.; Ullah, R.; Tian, F.; Xin, X. Demand Forecasting DBA Algorithm for Reducing Packet Delay with Efficient Bandwidth Allocation in XG-PON. *Electronics* **2019**, *8*, 147. [CrossRef]
16. Rafiq, A.; Hayat, M. QoS-Based DWBA Algorithm for NG-EPON. *Electronics* **2019**, *8*, 230. [CrossRef]
17. Liu, Z.; Gan, C.; Xie, W.; Yan, Y.; Qiao, H. Algorithm of both release and allocation bandwidth for downstream channel in multi-OLT PON. *IET Commun.* **2018**, *12*, 824–831. [CrossRef]
18. Yasunaga, R.; Walid, A.; Ananth, P. Modular Dynamic Bandwidth Allocation for a Flexible PON: Concept and Evaluation. In Proceedings of the 2017 European Conference on Optical Communication (ECOC), Gothenburg, Sweden, 17–21 September 2017; pp. 1–3.
19. VOLTHA—Open Networking Foundation. Available online: <https://www.opennetworking.org/voltha/> (accessed on 9 April 2019).
20. Horvath, T.; Munster, P.; Cymorek, P.; Oujezsky, V.; Vojtech, J. Implementation of NG-PON2 transmission convergence layer into OPNET modeler. In Proceedings of the 2017 International Workshop on Fiber Optics in Access Network (FOAN), Munich, Germany, 6–8 November 2017; pp. 1–5.
21. Horvath, T.; Cymorek, P.; Munster, P.; Oujezsky, V.; Vojtech, J. Simulations of Grant Allocation in NG-PON2 Networks Using OPNET Modeler. *J. Commun. Softw. Syst.* **2018**, *14*, 281–289. [CrossRef]
22. Kim, K.O.; Doo, K.H.; Lee, H.H.; Kim, S.H.; Park, H.; Oh, J.Y.; Chung, H.S. High Speed and Low Latency Passive Optical Network for 5G Wireless Systems. *J. Lightwave Technol.* **2019**, *1*, 2873–2882. [CrossRef]
23. Nakayama, Y.; Uzawa, H.; Hisano, D.; Ujikawa, H.; Nakamura, H.; Terada, J.; Otaka, A. Efficient DWBA Algorithm for TWDM-PON with Mobile Fronthaul in 5G Networks. In Proceedings of the GLOBECOM 2017—2017 IEEE Global Communications Conference, Singapore, 4–8 December 2017; pp. 1–6.
24. Mesodiakaki, A.; Maniotis, P.; Vagionas, C.; Gatzianas, M.; Datsika, E.; Kartsakli, E.; Vardakas, J.; Kalfas, G. Medium-transparent Dynamic Bandwidth Allocation for 5G Fiber Wireless Dense Fronthaul Networks. In Proceedings of the 2018 IEEE 23rd International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD), Barcelona, Spain, 17–19 September 2018; pp. 1–6.
25. Horvath, T.; Munster, P.; Oujezsky, V.; Vojtech, J. Activation Process of ONU in EPON/GPON/XG-PON/NG-PON2 Networks. *Appl. Sci.* **2018**, *8*, 1934. [CrossRef]

