Design of DBA Algorithms for EPONs

Swapnil Bhatia
Department of Computer Science
Univ. of New Hampshire
Email: sbhatia@cs.unh.edu

***Abstract*— The recently adopted IEEE 802.3ah standard for Ethernet-based Passive Optical Networks (EPONs) leaves the algorithm for dynamic bandwidth allocation (DBA) open to vendor implementation. In response, a number of DBA algorithms have been proposed. We outline simple modifications to existing algorithms to optimize their average delay performance. Our modifications are generally applicable to a broad class of DBA algorithms. In the process, we find compelling reasons to study DBA as a scheduling problem. Constructing such a formulation may be challenging and further work along this direction is in progress.**

I. Introduction

Currently, the access portion of the Internet is the main bottleneck in providing next-generation services such as triple play (voice, video and data) to subscribers. *Ethernet Passive Optical Networks (EPONs)* are the emerging contenders as a solution to this "last mile" problem. An EPON is a point-to-multipoint, bidirectional, high rate optical network for data communication. The EPON link is shared by multiple users. Each user connects to the EPON link through a device known as an *Optical Network Unit (ONU)*. Since the link is shared, link use must be centrally arbitrated. This function is performed by a single special device called the *Optical Line Terminator (OLT)*. The direction of communication from the ONUs to the OLT is known as *upstream* direction whereas the direction from the OLT to the ONUs is known as the *downstream* direction. The data rate in each direction is set to 1 Gbps by the IEEE EPON standard [1]. Overall, the link exhibits a star topology with the OLT at the root of the star and the ONUs at the leaves. The EPON link is shared by all users in the upstream direction. The OLT decides which ONU is allowed to transmit data and for how many bytes. The OLT uses a special control message called a *Gate* to grant transmission opportunities to ONUs. Appended to the data traffic, the ONU also transmits a control message containing a *Report* of the number of bytes buffered in its queue, waiting for a subsequent transmission opportunity. An algorithm implemented in the OLT, which uses these reports and gate messages to construct a transmission schedule and conveys it to the ONUs is known as a DBA algorithm.

II. Existing DBA Algorithms: Global vs. Local

Existing solutions to the DBA problem can be broadly classified into two categories: global and local schemes. The defining characteristic of global schemes is the amount of information required by them to make a DBA decision. Global schemes usually collect *all* available information, i.e., queue reports from all the ONUs connected to the EPON before making any bandwidth allocation decisions. Thus, their DBA decisions are "global" and consider the demands of all ONUs. Many existing schemes fall into this category [2][3][4]. Local schemes do not require information from all ONUs connected to the EPON. Their DBA decisions regarding the bandwidth to be allocated to any ONU j can be impromptu and "local", i.e., based on perhaps a report from only that particular ONU j. IPACT, one of the earliest schemes proposed as a solution to the DBA problem, as well as schemes based on IPACT fall into this category [5][6]. More formally, let $g(i,j)$ and $q(i,j)$ denote the size of the i-th grant and reported queue (respectively) for ONU j. Global schemes compute the grant size as $g(i,j) = f(q(i-1,1),\ldots,q(i-1,N))$, whereas local schemes compute the grant size as $g(i,j) = f(q(i-1,j))$ for some function f defining their DBA policy. Local schemes are usually much simpler to implement than global schemes. Global schemes can usually incorporate fairness policies into their decisions and can produce better DBA assignments owing to the extra information. However, waiting for information from all ONUs usually means that the global algorithm may not be able to overlap the delay due to downstream messaging required to assign grants with other upstream transmissions, thus affecting the upstream utilization of the EPON link. However, some schemes side-step this problem by making certain assumptions about the traffic pattern [3][4].

III. Global Schemes and Sorted Granting

EPONs are designed to carry all types of user traffic including high quality video and voice. Such traffic requires delay guarantees and hence the delay or waiting time is the most important measure for determining the quality of any DBA algorithm. All existing algorithms report delay performance to prove the superiority of their proposed scheme. Define the average delay per ONU in cycle i as

$$\hat{D}(i) = \frac{1}{N} \cdot \sum_{j=1}^{N} D(i,j) \tag{1}$$

where $D(i,j)$ is the delay faced by ONU j in the ith cycle. Consider global schemes which possess information about the demands from all N ONUs. Measure $D(i,j)$ from the beginning of cycle i. Now, at the beginning of a new cycle i, the algorithm knows the size of each grant $g(i,j)$ to be allocated to ONU j. It is easy to see that Eqn. (1) will be *minimized when grants* $g(i,j)$ *are scheduled in increasing* *order of their size.* To see this consider the following simple argument from scheduling theory [7]. Suppose S is a sequence of grants $g(i,j)$ such that grants for ONUs j and m are scheduled consecutively in S. Thus, $S = \{G_1, g(i,j), g(i,m), G_2\}$ where G_1 and G_2 represent the (possibly empty) prefix and suffix sequences of S. Assume that $g(i,j) > g(i,m)$. Suppose the last grant in G_1 ends at time $t \geq 0$. Then $D_S(i,j) = t+g(i,j)$ and $D_S(i,m) = t+g(i,j)+g(i,m)$. Now, consider a schedule $\bar{S}$ where the order of $g(i,j)$ and $g(i,m)$ is switched, i.e., $\bar{S} = \{G_1, g(i,m), g(i,j), G_2\}$. Clearly, $D_{\bar{S}}(i,m) = t + g(i,m)$ and $D_{\bar{S}}(i,j) = t + g(i,m) + g(i,j)$. Thus,

$$N\hat{D}_S(i) - N\hat{D}_{\bar{S}}(i) = g(i,j) - g(i,m) > 0,$$

proving that it is always beneficial to order grants according to their size. Thus, by simply sorting the grants by size, any global scheme can improve the average waiting time of ONUs. Note that while existing schemes have not considered our grant ordering heuristic, Ma et al. [6] have instead proposed sorting in *descending order* of grant size. *Prima facie*, this strategy seems far from optimal.

IV. Value of Queue Reports for Local Schemes

What do local schemes lose by not waiting for more information? We consider local schemes which use a "gated" allocation policy [5], i.e., $g(i,j) = q(i-1,j)$. For the i-th cycle, this reduces to the problem of single-machine online scheduling without preemption with jobs arriving over time. Each report message $q(i,j)$ is a new job and the total number of jobs is unknown in advance since a local scheme acts on a per-report basis oblivious of the total number of ONUs yet to be allocated. For this model, an elegant argument by Hoogeveen and Vestjens [8] shows that a local scheme cannot hope to produce a schedule with a delay within a factor of less than 2 of the optimal schedule (i.e., one produced by the same scheme but with full information of all demands). Thus, waiting to recieve more information is advantageous for any local scheme. Different waiting strategies for a local scheme are possible. If a local scheme waits till it receives information from all N ONUs, then it turns into a global scheme. Hoogeveen and Vestjens also provide a waiting strategy that achieves the 2-competitive schedule. We can adapt their algorithm to local DBA schemes, i.e., IPACT by using the following rule [8]: *Suppose the new DBA cycle begins at some absolute time T. All times below are measured starting from T. If at time t into the cycle, the EPON link is available to be assigned to any ONU, and if queue reports $q(i-1,j)$ have been received, then determine the ONU j with the smallest demand (with earliest report reception as a tie-breaker). Now, if $q(i-1,j) \leq t$ then schedule ONU j with grant size $q(i-1,j)$. Otherwise, wait until time $t = q(i-1,j)$ into the DBA cycle to schedule ONU j or until a new queue report is received (whichever occurs first).* Inclusion of the above rule in the existing local algorithms (i.e. IPACT-based schemes [5][6] under a gated policy) can provide them with a waiting strategy that optimizes the delay as much as possible with nonpreemptive, deterministic, local schemes. To our knowledge, no such heuristic has been considered for local schemes yet.

V. DBA as a Scheduling Problem

The above observations offer compelling reasons to approach the DBA problem as one of scheduling. We believe that many fundamental bounds on the performance possible from any EPON DBA algorithm could be proved if the DBA problem were formulated as a scheduling problem. Over the last several decades, many variants of the standard scheduling model have been studied with many interesting results [9]. Consider two variants. In the *online-list model*, jobs are presented as a list. A scheduler must assign a job to one of m machines before being presented the next job. In this case, job arrival and completion times are irrelevant and meaningless; this model is best suited for studying load balancing. In the *online-time* model, jobs arrive over time. Each job j is released at a certain time r_j at which its processing requirement p_j becomes known to the scheduler. The scheduler must design a schedule which respects the release times of the jobs and minimizes some function of the completion times of the jobs. While the DBA problem fits neither, it can perhaps be viewed as a combination of online-time model with N online-lists. The DBA algorithm can wait for N jobs but will not see any new jobs unless it schedules one of them. Also note that in the DBA problem, the release time and the time at which the processing requirement of a job becomes known may be different. Formulation of DBA as a scheduling problem is a promising direction and work in this direction is still in progress.

Acknowledgments

The author thanks Prof. Radim Bartoš for his useful comments and guidance, and Cisco Systems for supporting this work under the Cisco University Research Program.

References

[1] *IEEE Draft P802.3ah (tm)*, IEEE Standard, 2004.
[2] C. M. Assi, Y. Ye, S. Dixit, and M. A. Ali, "Dynamic Bandwidth Allocation for Quality-of-Service Over Ethernet PONs," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 9, pp. 1467–1477, Nov. 2003.
[3] X. Bai, A. Shami, N. Ghani, and C. Assi, "A Hybrid Granting Algorithm for QoS Support in Ethernet Passive Optical Networks," in *International Conference on Communications (ICC)*, 2005, pp. 1869–1873.
[4] A. Shami, X. Bai, C. Assi, and N. Ghani, "Jitter Performance in Ethernet Passive Optical Networks," *J. Lightwave Technol.*, vol. 23, no. 4, pp. 1745–1753, Apr. 2005.
[5] G. Kramer, B. Mukherjee, and G. Pesavento, "Interleaved Polling with Adaptive Cycle time (IPACT): A Dynamic Bandwidth Distribution Scheme in an Optical Access Network," *Photonic Network Communications*, vol. 4, no. 1, pp. 89–107, Jan. 2002.
[6] M. Ma, L. Liu, and T. H. Cheng, "Adaptive Scheduling for Differentiated Services in an Ethernet Passive Optical Network," *Journal of Optical Networking*, vol. 4, no. 10, pp. 661–670, Oct. 2005.
[7] R. G. Parker, *Deterministic Scheduling Theory*. Chapman and Hall, 1995.
[8] J. A. Hoogeveen and A. P. A. Vestjens, "Optimal On-line Algorithms for Single-Machine Scheduling," in *Proc. of the 5th Conference on Integer and Combinatorial Optimization*, 1996.
[9] K. Pruhs, E. Torng, and J. Sgall, "Online Scheduling," in *Handbook of Scheduling: Algorithms, Models and Performance Analysis*, J. Y. T. Leung, Ed. CRC Press, 2004, ch. 15.