Editorial Manager(tm) for Photonic Network Communications
Manuscript Draft

Manuscript Number:

Title: Analysis of average burst-assembly delay and applications in proportional service differentiation

Article Type: Manuscript

Keywords: Optical Burst Switching; Size-based burst assembly algorithm;
average assembly delay; proportional delay-based service differentiation.

Corresponding Author: Dr. Jose Alberto Hernandez, Ph.D.

Corresponding Author's Institution: Universidad Autónoma de Madrid

First Author: Pedro Reviriego, Ph.D.

Order of Authors: Pedro Reviriego, Ph.D.; Jose Alberto Hernandez, Ph.D.; Javier Aracil, Ph.D.

Analysis of average burst-assembly delay and applications in proportional service differentiation

Pedro Reviriego, José Alberto Hernández and Javier Aracil

Abstract

In Optical Burst-Switched (aka, OBS) networks, the limitation of optical buffering devices make it impractical to deploy conventional delay-based differentiation algorithms such as Active Queue Management, Weighted Fair Queuing, etc. Furthermore, since only the delay that appears due to the burst-assembly process constitutes a variable quantity (all the other sources of delay are mostly fixed), it is then reasonable to make use of the burst-assembly algorithm to provide class-based delay differentiation.

The aim of the following study is two-fold: first it defines an average assembly delay metric, which represents the assembly delay experienced by a random arrival at the burst assembler of an edge OBS node; and secondly, this metric is used to define and configure a two-class burst-assembly policy which gives preference to high-priority traffic over low-priority packet arrivals.

The results show that, (1) tuning the parameters of the two-class assembly algorithm, the two classes of traffic exhibit different burst-assembly delay; and, (2) such parameters can be adjusted to provide a given differentiation ratio in the light of the proportional QoS differentiation approach proposed in the literature. A detailed analysis of the two-class assembly algorithm is given, along with an exhaustive set of experiments and numerical examples that validate the equations derived.

Index Terms

Optical Burst Switching; size-based burst assembly algorithm; average assembly delay; proportional delay-based service differentiation.

I. Introduction and motivation

Dense Wavelength Division Multiplexing (DWDM) has boosted the amount of available raw bandwidth provided in core networks, since multiple wavelengths, each in the order of Gigabits/sec, can be used for the transmission of data traffic over the same optical fibre [1]. In this light, the research community has proposed the Optical Burst Switching (OBS) paradigm as a cost-effective approach for the maximum utilisation of such raw bandwidth at a moderate computational cost [2], [3], [4], [5].

In OBS networks, packets are assembled into large-size optical bursts at the ingress nodes of the optical network. Once a data burst is completed, its Burst Control Packet (or BCP) is generated and transmitted. The role of the BCP is to advertise each intermediate node of the imminent data burst arrival, and reserve resources for its allocation and switching at each node in the source-destination path, on attempts to reduce burst contention [2]. The time difference between the BCP and its associated data burst is known as offset time, and must be an amount of time enough to allow O/E/O conversion and processing at each intermediate node. After all intermediate nodes are configured properly, the data burst is transmitted all optically, thus suffering only propagation delay.

In this light, packets traversing an OBS network suffer two main types of delay: burst-assembly delay and offset delay, since propagation delay is almost negligible compared to the other two. The former comprises the time that packets spend until the optical burst is completed, and is typically governed by the burst-assembly policy set by the network administrator. On the other hand, the offset delay is typically fixed by the network topology and is rarely modified.

Hence, the variability of packet delay in OBS networks is mainly due to the burst-assembly algorithm employed at the ingress nodes, since all the other sources of delay are constant. The research community has proposed several burst-assembly algorithms, mainly focusing on either limiting the burst-release time (see the timer-based algorithms [6]), or sizing the outgoing burst to a fixed value (see [7]), or a combination of both [8], [9]. The reader is referred to [10] for a detailed analysis of the burst-assembly delay suffered by each packet in a burst under any of the above burst-assembly policies.

As shown in [10], the delay of each packet is typically determined by its relative position within the burst, that is, if a packet arrives when a burst is almost completed, it suffers less delay than if it arrives when the burst is still empty. This characteristic of OBS networks can be exploited to provide Quality of Service to applications and services, a key aspect in the engineering of the Next Generation of Internet.

In the light of this, the Proportional Differentiation approach proposed in [11] brings a simple but powerful mechanism that ensures QoS differentiation (not absolute QoS guarantees) between different types of traffic. Essentially, different applications and services are classified into classes of traffic, which are proportionally benefited or prejudiced against other classes according to some metric, typically loss or delay.

The research community has pointed out that the mechanisms employed for QoS differentiation in OBS must be different to those employed in conventional IP networks, and need to be designed carefully. The reason for this is that most of the existing techniques to implement QoS in IP routers, say Active Queue Management, Weighted Fair Queuing, etc [12], [13], [14] rely on the use of buffering to provide different treatment to different traffic classes or flows. However, in OBS, optical buffering is much more limited and costly than in the electronic domain, and Quality of Service has to be implemented without their help.

Concerning QoS differentiation in OBS networks, most of the mechanisms presented in the literature focus on the blocking probability observed by different classes of traffic as they traverse the network. For example, as shown in [15], [16], the offset time can be used to reduce the blocking probability for high-priority classes, an approach which is further extended and formalised in [17] with the class isolation theorem. A slightly different approach is proposed in [18] at which all incoming bursts within a given time window are grouped up first, then sorted based on priority, and finally scheduled following such arrangement. However, this mechanism requires the use of extra offset, which substantially increases the end-to-end delay suffered by packets in the burst. This may not be acceptable for real-time applications. Alternative mechanisms based on higher-priority bursts overriding previous low-priority reservations either partially (segmentation) or totally (preemption) have also been proposed to reduce the blocking probability of high-priority traffic [19], [20].

However, little attention has been paid to providing delay-based service differentiation which is key for certain applications such as online gaming, telephony over IP, videoconferencing, etc. In fact, some of those applications can tolerate some level of packet loss and, therefore, a given guarantee of blocking probability is less important. In this light, since the majority of delay is due to the burst-assembly process and the offset time, and the latter is typically set by the network topology and cannot be modified, the only possible way to address delay-based class differentiation is to make use of the burst assembly process. Furthermore, the offset delay can be totally or

partially removed just by sending the BCP packet with an estimate of the final burst size and expected release time, even before the data burst is actually completed (see [21], [22]). To the best of the authors' knowledge, only [10] proposes a mechanism to bound the maximum burst assembly delay experienced by packets that belong to different traffic classes. Although the maximum assembly delay is a good metric to define delay differentiation between different classes of traffic, it poses one main problem: This metric says nothing about the actual delay experienced by packets in each class. In this light, it may well happen that low-priority packets experience less delay, on average, that high-priority packets although the latter satisfies a more restrictive maximum delay bound. Thus, in practice, it is more interesting to define QoS mechanims based on "average delays" rather than "maximum delays" experienced by packets, since this gives a more realistic view of the actual delay experienced by the packets of the same class.

The contribution of this paper is thus two-fold: First, it describes the concept of *average assembly delay* observed by packets in a burst; and secondly, it uses this metric to define a two-class burst-assembly algorithm that provides proportional QoS delay differentiation. This algorithm is mathematically analysed in detail, and further validated with simulation experiments and numerical examples.

II. Analysis of average burst-assembly delay

This section addresses the concept of "average burst-assembly delay" and its mathematical formulation and analysis. Clearly, the assembly delay experienced by the packets depends on their relative position within the burst. That is, under a size-based burst assembly policy, the first packet in a maximum of N must wait for $N-1$ subsequent arrivals, which is very likely to be much larger than the assembly delay experienced by the last packets in the burst. However, it is also possible that the first packet in a, say for instance, 5-packet burst suffers less delay than the second packet in another (different) 5-packet burst since it happened that the four subsequent packets in the first burst arrived closer in time, whereas in the second burst, the three subsequent packets did not arrive so close together and took longer. For this reason, it is interesting to obtain an "average burst-assembly delay" metric that takes into account all these situations, and provides a measure of the assembly delay that packets experience *on average* in a sized-based burst-assembly policy.

A. Notation and preliminaries

Let packet arrivals be assumed to follow a Poissonian process at the OBS burst-assembler, as it is the case for highly-multiplexed core Internet traffic [23]. For notation purposes, we shall assume that the first packet arrives at time $t_1 = 0$, the second packet arrives at time $t_2 = x_1$, the third packet arrives at time $t_3 = x_1 + x_2$, and so forth. Clearly, the random variables x_i denote the inter-arrival times between the i-th and the $i+1$-th packets, as shown in figure 1. The x_i values are assumed to be exponentially distributed with rate $\lambda = 1/\mathbb{E}(X)$ (Poisson assumption).

Therefore, the i-th packet in a total of $n+1$ arrivals suffers a burst-assembly delay given by $t_i = \sum_{k=i}^{n} x_i$. The last packet in the burst (packet number $n+1$) suffers no burst-assembly, since no subsequent data packets are expected.

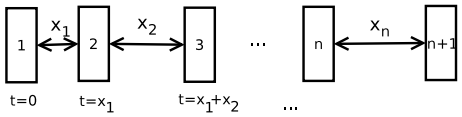

Fig. 1. Notation

Let z_{n+1} denote the average burst-assembly delay suffered by the packets in a burst of $n+1$ packets, that is, the burst-assembly delay that a packet would suffer if it falls randomly in such burst. This value is given by:

$$\begin{aligned} z_{n+1} &= \frac{1}{n+1}[(x_1 + \ldots + x_n) + (x_2 + \ldots + x_n) \\ &+ \ldots + (x_{n-1} + x_n) + x_n] = \frac{1}{n+1}\sum_{j=1}^{n} j x_j \end{aligned} \tag{1}$$

The following studies the probability density function (PDF) of the random variable z_{n+1}, that is, $f_{z_{n+1}}(t)$, $t > 0$.

B. Probability density function of z_{n+1}

To obtain the PDF of z_{n+1}, it is first worth noticing that the random variable $(j/(n+1))x_j \sim \exp(\lambda(n+1)/j)$. Thus, the average burst-assembly delay is just the sum of n exponentially distributed random variables, with decreasing parameter $\lambda(n+1)/j$, with $j = 1, \ldots, n$. The easiest way to proceed makes use of the moment generating function.

Recall that the moment generating function of an exponentially distributed random variable x with parameter θ is $M_x(s) = (1 - s/\theta)^{-1}$. Hence, the moment generating function of z_{n+1} is the product of the moment generating function of each component in the sum given by eq. 1, due to the independence of the x_j random variables, i.e.:

$$M_{z_{n+1}}(s) = \prod_{j=1}^{n} \frac{1}{1 - j\frac{s}{(n+1)\lambda}} \tag{2}$$

which can be decomposed into partial fractions:

$$M_{z_{n+1}}(s) = \sum_{j=1}^{n} \frac{A_j}{1 - j\frac{s}{(n+1)\lambda}} \tag{3}$$

whereby the A_j coefficients must be thus computed. By inspection, it can be shown that the A_j coefficients take the following values:

$$A_j = \left(\prod_{k=1, k\neq j}^{n} \left(1 - \frac{k}{j}\right) \right)^{-1} \tag{4}$$

for $j = 1, \ldots, n$. Accordingly, eq. 3 can be tranformed back to:

$$f_{z_{n+1}}(t) = \sum_{j=1}^{n} A_j \frac{\lambda(n+1)}{j} e^{-\frac{\lambda(n+1)}{j}t} \tag{5}$$

for $n = 1, 2, \ldots$.

The mean and variance arise easily from:

$$\mathbb{E}(z_{n+1}) = \mathbb{E}\left(\frac{1}{n+1}\sum_{k=1}^{n} kx_k\right) = \frac{1}{n+1}\sum_{k=1}^{n} k\mathbb{E}(x_k) = \frac{n}{2\lambda} \tag{6}$$

$$\text{Var}(z_{n+1}) = \text{Var}\left(\frac{1}{n+1}\sum_{k=1}^{n} kx_k\right) = \frac{1}{(n+1)^2}\sum_{k=1}^{n} k^2\text{Var}(x_k) = \frac{n(2n+1)}{6(n+1)\lambda^2} \tag{7}$$

since $\mathbb{E}(x_k) = \frac{1}{\lambda}$ and $\text{Var}(x_k) = \frac{1}{\lambda^2}$.

C. Validation

In this first experiment, we have simulated the generation of optical bursts with a maximum of $N_{\max} \in \{2, 4, 6, 8\}$ packets in each burst, assuming the arrival rate of $\lambda = 8$ packets/sec. We have further evaluated the PDFs for z_2, z_4, z_6 and z_8 analitically, following the equations derived in the section above, and plotted them together with the histograms of the assembly delay of randomly chosen packets obtained via simulation (see figure 2). As shown, both the theoretical PDF given by eq. 5 and the simulated average burst-assembly values perfectly match.

Interestingly, as the number of packets in a burst increases, the average burst-assembly delay also increases.

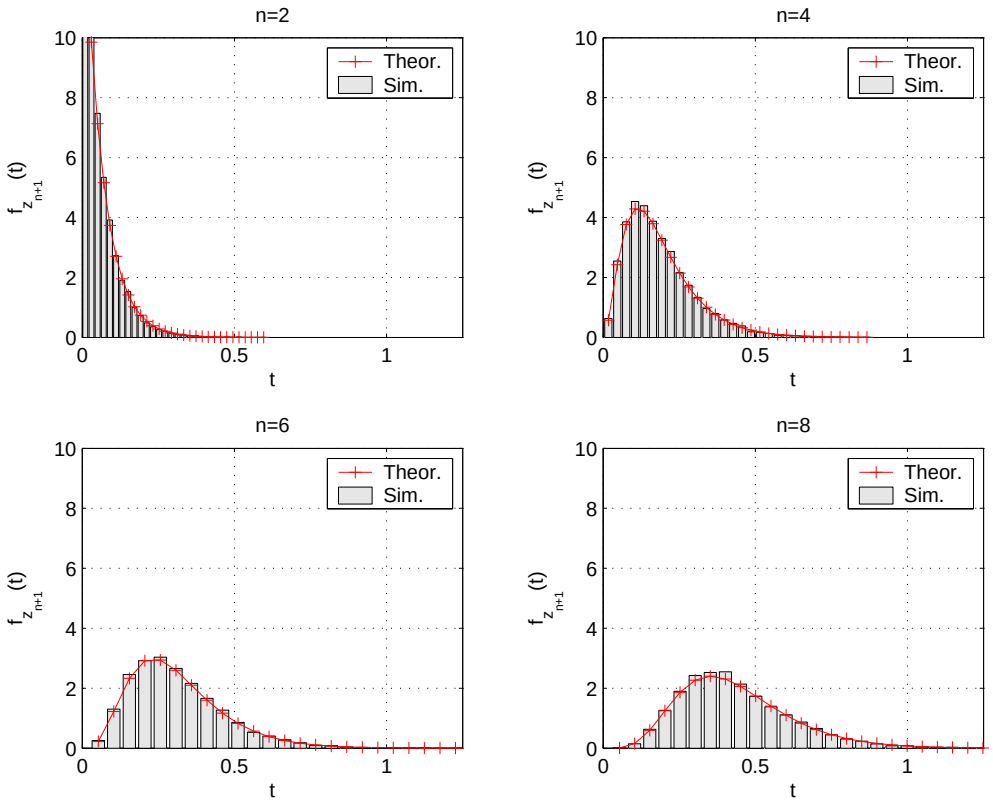

Fig. 2. Probability distribution of z_2 (top-left), z_4 (top-right), z_6 (bottom-left), z_8 (bottom-right).

As shown, for bursts with only two packets in it, the first packet experiences a burst delay of $x_1 \sim exp(\lambda)$, whereas the second (and last) packet experiences zero delay. Thus the average-burst assembly delay is exponentially distributed with rate $\lambda/2$ (fig. 2 top-left). When the maximum number of packets per burst increases, the average burst-assembly delay also increases and becomes more variable, as shown in the remaining plots of fig. 2. Essentially, the average delay mean and variance observed by random packets in a burst increases with the number of packets in each burst (eq. 6 and 7).

The next section shows how to apply these results to define a two-class size-based burst-assembly strategy to provide delay-based service differentiation between different QoS classes of traffic.

III. Analysis of bursts with two service classes

The above has introduced the random variable z_{n+1} which constitutes a measure of the "average assembly delay" experienced by packets arriving randomly at a given edge OBS node that employs a size-based assembly policy with $n+1$ packets per burst. Essentially, z_{n+1} considers the assembly delay contribution of all packets in the burst and computes an average of such values. Clearly, the first arrival suffers much more delay than the last arrival, but since packets arrive randomly at the burst assembler, z_{n+1} gives a measure of the average assembly delay.

As shown in eq. 6, the expectation of such average delay grows linearly with the number of packets in a burst $n+1$. Therefore, the policy of generating large-size data bursts, although it tends to maximise the utilisation of the raw bandwidth available by the DWDM physical layer, it may result harmful to certain applications due to the excessive burst-assembly delay of the first arriving packets. In this light, it is challenging to find a burst-assembly policy that trades off such two aspects: link utilisation and delay. In other words, it is key to define a mechanism that maximises the size of transmitted data bursts, but at the same time is somehow friendly with delay-sensitive applications.

In today's Internet, the majority of applications belongs to the so-called *elastic applications*, which means that they tolerate large delays (but not excessive). Examples include: web-browsing, emailing, file sharing, etc. However, the Internet also carries a small amount of traffic that belongs to *real-time applications*, which are delay-sensitive and whose performance is highly degraded if the end-to-end delay exceeds a certain value. Examples of these are: multimedia streaming, online gaming or Internet telephony. In this light, if the burst assembler is configured to output small data bursts, although the delay constrains of high-priority traffic is met, the amount of processing and O/E/O conversions per unit of data increases, thus leading to small utilisation levels of the OBS network. This problem is particularly harmful if high-priority traffic constitutes only a small portion of the total traffic (which is true most of the times), since it may well happen that a burst assembler generates a small-size data burst on attempts to reduce assembly delay, but none of the actual data packets in it are of high priority.

To solve this problem, a two-class size-based (N_l, N_h) burst-assembly policy can be defined, with two size thresholds proposed: N_l and N_h. The former controls the maximum number of packets in the burst assembled after a low-priority packet has arrived, whereas the latter regulates the maximum number of packets in the burst after a high-priority arrival. For example, let $N_l = 10$ and $N_h = 4$, and let the first packet arrival be of low priority.

Such low-priority arrival sets the maximum number of subsequent arrivals to $N_{\max} = N_l = 10$ packets. Hence, if no high-priority packets arrive within the subsequent 10 packets, the final burst size would be $N_{\text{burst}} = 11$ packets. However, if the second arrival is of high-priority, this sets the maximum burst size to $N_{\max} = 4$, leading to a final burst size of $N_{\text{burst}} = 2 + 4 = 6$ packets. Therefore, the total number of packets in a burst always satisfies $N_{\text{burst}} = \min(N_l + 1, h_1 + N_h)$ where h_1 denotes the position of the first high-priority packet arrival.

With this policy, high-priority packets are benefited against low-priority packets since they have the right to shorten the final burst size on attempts to reduce their expected average assembly delay. Indeed, on average, high-priority packets occupy the latest positions in the burst, which are the ones that exhibit less burst-assembly delay. On the contrary, low-priority packets are stored in the former positions of a burst, thus suffering more assembly delay than high-priority packets. The following analyses the average assembly delay, as defined in the previous section, experienced by high- and low-priority packets following such policy.

Again, packet arrivals shall be assumed to follow a Poissonian basis with rate λ. The value p_h denotes the probability of a packet to be of high-priority (thus $1 - p_h$ is the probability of low-priority packet). According to this, high-priority traffic arrives following a Poisson process with rate λp_h and low-priority traffic follows a Poisson process with rate $\lambda(1 - p_h)$.

A. Preliminaries

Let $B_{i,n}$, denote a burst generated following the two-class (N_l, N_h) burst-assembly policy, with $N_h + 1 \leq n \leq N_l + 1$ packets (high-priority and low-priority packets), and $0 \leq i \leq N_h + 1$ packets of high priority in it (see Fig. 3). Also, let $h_1 \leq N_l + 1$ denote the position at which the first packet of high-priority occurs, and let $k = n - h_1 \leq N_h$ denote the number of packets (high-priority and low-priority packets) arriving after the first high-priority packet in the burst. Thus, $h_1 - 1 = n - k - 1$ denotes the number of low-priority arrivals before the first high-priority packet. The value of k will be very helpful in the analysis of key metrics of bursts following this assembly strategy, say: burst size and average delay.

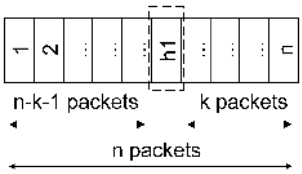

Fig. 3. Notation for a typical burst $B_{i,n}$

To do so, the first quantity required is $P(B_{i,n})$, that is, the probability to have a burst with n packets and i packets of high-priority in it. Fig. 4 shows the region of integration of $B_{i,n}$ which gives all the possible values over which the random variable $B_{i,n}$ is defined, namely $S = \{(n, i) \in \mathbb{Z} \times \mathbb{Z}, \quad N_h + 1 \leq n \leq N_l + 1, \quad 1 \leq i \leq N_h + 1\} \cup (N_l + 1, 0)$ (see figure 4).

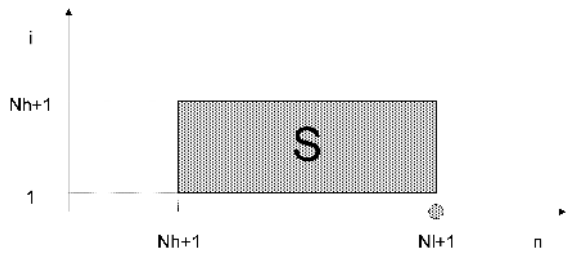

Fig. 4. Region of integration for $B_{i,n}$

Clearly, the main features of the final burst $B_{i,n}$ depend on the position of the first high-priority packet h_1, which determines its size n and has a clear impact on the number of high-priority packets arriving after it, i.e. $P(h_1)$:

$$P(h_1 = m) = \begin{cases} (1-p_h)^{N_l+1}, & \text{if } m = 0 \\ (1-p_h)^{m-1}p_h, & \text{if } m = 1, \ldots, N_l+1 \end{cases} \tag{8}$$

which accounts for the probability to have $h_1 - 1$ low-priority packet arrivals and packet h_1 is of high-priority. The value $P(h_1 = 0)$ refers to the case of bursts with no high-priority packets in it: B_{0,N_l+1}.

In order derive $P(B_{i,n})$, that is, the probability for the burst assembler to output the burst $B_{i,n}$, we consider the conditional probabilities $P(B_{i,n}|h_1)$. To do so, we must take into account the following three cases (see fig. 5) separately:

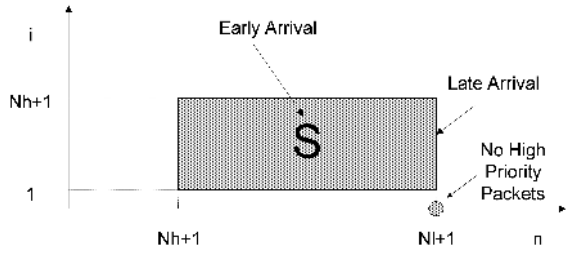

Fig. 5. Region of integration for $B_{i,n}$ (separated by cases)

- **Case 1 ($1 \leq h_1 < N_l + 1 - N_h$):** The burst contains $n < N_l + 1$ packets in it. In this case, the number of packets after the first high-priority arrival is $k = N_h$. In what follows, we shall refer to this case as *early arrival of* h_1, since the first high-priority packet h_1 forces the data burst not to reach the maximum possible size defined by the burst assembly policy, that is $N_l + 1$, packets. In this case,

$$\begin{aligned} P(B_{i,n}|h_1) &= \binom{N_h}{i-1} p_h^{i-1}(1-p_h)^{n-(i-1)}, \\ & \qquad 0 < i-1 \leq N_h, \quad n = h_1 + N_h, \quad 1 \leq h_1 < N_l + 1 - N_h \end{aligned} \tag{9}$$

  and zero otherwise.

- **Case 2 (**$N_l + 1 - N_h \leq h_1 \leq N_l + 1$**):** The burst contains $n = N_l + 1$ packets in it. In this case, $k = N_l + 1 - h_1 < N_h$, that is, the number of maximum possible high-priority packet arrivals depends on the position of the first high-priority h_1. This case shall be denoted as *late arrival of* h_1, since h_1 exceeds or is equal to $N_l + 1 - N_h$, and the burst size is the maximum possible. In this case:

$$\begin{aligned} P(B_{i,n}|h_1) &= \binom{N_l + 1 - h_1}{i-1} p_h^{i-1} (1-p_h)^{N_l+1-h_1-(i-1)}, \\ & 0 < i - 1 \leq N_l + 1 - h_1, \quad n = N_l + 1, \quad N_l + 1 - N_h \leq h_1 \leq N_l + 1 \end{aligned} \tag{10}$$

  and zero otherwise.

- **Case 3 (**$h_1 = 0$**)**: In this case, the burst contains $n = N_l + 1$ packets, and all of them are of low priority. This case shall be denoted as *no arrival of* h_1 and follows:

$$P(B_{i,n}|h_1) = 1 \qquad i = 0, \quad n = N_l + 1, \quad h_1 = 0$$

  and zero otherwise.

Now, we simply consider that $P(B_{i,n}) = \sum_{j=0}^{N_l+1} P(B_{i,n}|h_1 = j)P(h_1 = j)$ and we obtain:

$$P(B_{i,n}) = \begin{cases} (1-p_h)^{N_l+1}, & \text{if } i = 0, \quad n = N_l + 1 \\ \binom{N_h}{i-1} p_h^i (1-p_h)^{n-i}, & \text{if } 0 < i \leq N_h + 1, \quad n < N_l + 1 \\ \sum_{h_1=N_l+1-N_h}^{N_l+1-(i-1)} \binom{N_l+1-h_1}{i-1} p_h^i (1-p_h)^{N_l+1-i}, & \text{if } 0 < i \leq N_h, \quad n = N_l + 1 \end{cases} \tag{11}$$

and zero otherwise.

B. Analysis of average burst size

The average burst size is a key metric of every burst assembly algorithm, since it is highly related to important metrics of the global performance behaviour of the OBS network, as noted in the introduction section.

To this end, let L denote the random variable which represents the length or size (in packets) of the data burst, and let $P(L = n)$ refer to the probability to have a burst with n packets in it. Then,

$$P(L = n) = \sum_{i=0}^{N_h - 1} P(B_{i,n}) \tag{12}$$

and

$$\mathbb{E}(L) = \sum_{n=N_h+1}^{N_l+1} nP(L = n) \tag{13}$$

Also, it is interesting to analyse the number of high- and low-priority packets in a burst. Following this, let L^{lp} and L^{hp} denote the number of high- and low-priority packets respectively in a given burst. The probability to have i packets of high-priority in a burst must consider all possible bursts $B_{i,n}$ regardless of its actual size n. Thus:

$$P(L^{hp}=i)=\begin{cases} P(B_{0,N_l+1}), & i=0 \\ \sum_n P(B_{i,n}), & 0<i\leq N_h+1 \end{cases} \quad (14)$$

Similarly, the probability to have j packets of low-priority in a burst is given by:

$$P(L^{lp}=j)=\begin{cases} P(B_{0,N_l+1}), & j=N_l+1 \\ \sum_n P(B_{n-j,n}), & 0\leq j<N_l+1 \end{cases} \quad (15)$$

Finally, the average number of high- and low-priority packets in a burst are given by:

$$\mathbb{E}(L^{hp}) = \sum_{i=1}^{N_h+1} iP(L^{hp}=i) \quad (16)$$

$$\mathbb{E}(L^{lp}) = \sum_{j=1}^{N_l+1} jP(L^{lp}=j) \quad (17)$$

which just weights the number of high- and low-priority packets in each burst times the probability to have such burst.

C. Analysis of average delay

This section analyses the delay experienced by the high- and low-priority packets in a burst assembled following the two-class burst-assembly strategy. To do so, let $D^{hp}_{B_{i,n}}$ denote the random variable which represents the delay experienced by the high-priority packets in the burst $B_{i,n}$. And, let $D^{lp}_{B_{i,n}}$ denote the same metric, but for low-priority packets.

Additionally, let x_l, $l=1,\ldots,n$ denote the interarrival time between the l-th packet and the $l+1$-th packet, and that all x_l are exponentially distributed with rate $\lambda=\frac{1}{\mathbb{E}(x_l)}$.

The goal is to derive the average delay experienced by high- and low-priority packets in a given burst $B_{i,n}$ with its first high-priority packet located at position h_1. Then, this value must be weighted with the probability of such packet to occur in that particular burst.

First of all, it is straightforward to derive the average delay experienced by the low- and high-priority packets for a burst with no high-priority packets in it, that is, B_{0,N_l+1}:

$$D^{hp}_{B_{0,N_l+1}} = 0 \quad (18)$$

$$D^{lp}_{B_{0,N_l+1}} = \frac{1}{N_l+1}\sum_{l=1}^{N_l} lx_l \quad (19)$$

if $h_1=0$.

with mean values, conditional to h_1:

$$\mathbb{E}(D^{hp}_{B_{0,N_l+1}}|h_1=0) = 0 \quad (20)$$

$$\mathbb{E}(D^{lp}_{B_{0,N_l+1}}|h_1=0) = \frac{N_l}{2\lambda} \quad (21)$$

Obviously, the delay for low-priority packets in this case is the same as the average delay of a burst with $N_l + 1$ packets described in the previous section, z_{N_l+1}.

Let us consider a burst $B_{i,n}$ generated by the two-class burst-assembly algorithm whose first high-priority packet is located at position h_1, and let us first concentrate on the analysis of the average delay experienced by the high-priority packets. Clearly, $D^{hp}_{B_{i,n}}$ must take into account two components (see fig. 6):

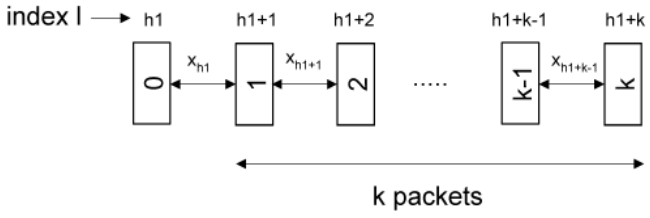

Fig. 6. Analysis of delay for high-priority packets

- The first packet (high-priority), which arrives at position h_1, suffers a total delay of $k = n - h_1$ exponentially distributed interarrivals:

$$\sum_{l=h_1}^{h_1+k-1} x_l$$

- The subsequent $i-1$ high-priority packets are located randomly in the following k positions and observe an average delay of:

$$z_k = \frac{1}{k} \sum_{l=h_1+1}^{h_1+k-1} (l - h_1) x_l$$

each of them. This makes use of the definition of average delay of a packet in burst of k packets, which is exactly what the high-priority packets observe in the latest k positions of the burst.

Thus, the random variable $D^{hp}_{B_{i,n}}$, which represents the average delay observed by the i high-priority packets, is the weighted sum of the two components:

$$D^{hp}_{B_{i,n}} = \frac{1}{i} \left(\sum_{l=h_1}^{h_1+k-1} x_l + (i-1) \frac{1}{k} \sum_{l=h_1+1}^{h_1+k-1} (l - h_1) x_l \right) \tag{22}$$

where the term $\frac{1}{i}$ weights the result over the total number of high-priority packets i and $0 < h_1 \leq N_l + 1$.

It is easy to compute its mean value given the linear properties of the expectation operator:

$$\begin{aligned} \mathbb{E}(D^{hp}_{B_{i,n}} | h_1) &= \frac{1}{i} \left(\sum_{l=h_1}^{h_1+k-1} \mathbb{E}(x_l) + (i-1) \frac{1}{k} \sum_{l=h_1+1}^{h_1+k-1} (l - h_1) \mathbb{E}(x_l) \right) \\ &= \frac{1}{i} \left(\frac{k}{\lambda} + (i-1) \frac{k-1}{2\lambda} \right) \end{aligned} \tag{23}$$

since $\mathbb{E}(x_l) = 1/\lambda$. It is also worth remarking that: $\left(\sum_{i=1}^{k-1} i\right) = \frac{(k-1)k}{2}$.

The above holds for both the cases of early arrival and late arrival of h_1, with $0 < h_1 \leq N_l + 1$. In the case of early arrival, the value of $k = N_h$, whereas in the case of late arrival, $k = N_l + 1 - h_1$.

For the analysis of $D^{lp}_{B_{i,n}}$ with $h_1 > 0$, again, the total contribution to delay by the low-priority packets is analysed separately with the following two cases (see fig. 7):

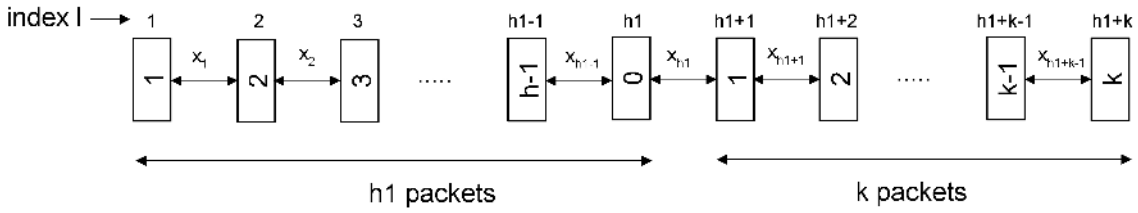

Fig. 7. Analysis of delay for low-priority packets

- The first packets until h_1 are of low priority and contribute with delay:

$$\sum_{l=1}^{h_1-1} l x_l + (h_1 - 1) \sum_{l=h_1}^{h_1+k-1} x_l$$

- The subsequent $j = k - (i-1)$ low-priority packets arriving after h_1 contribute, on average, with:

$$\frac{1}{k} \sum_{l=h_1+1}^{h_1+k-1} (l - h_1) x_l$$

each of them, as explained in the section above.

The sum of the two cases above yields:

$$\begin{aligned} D^{lp}_{B_{i,n}} &= \frac{1}{(h_1 - 1) + (k - (i-1))} \left(\sum_{l=1}^{h_1-1} l x_l + (h_1 - 1) \sum_{l=h_1}^{h_1+k-1} x_l + \right. \\ &+ \left. (k - (i-1)) \frac{1}{k} \sum_{l=h_1+1}^{h_1+k-1} (l - h_1) x_l \right) \end{aligned} \tag{24}$$

Clearly, the value $(h_1 - 1) + (k - (i-1)) = n - i$ since $k = n - h_1$. Again, using the properties of the expectation operator:

$$\mathbb{E}(D^{lp}_{B_{i,n}} | h_1) = \frac{1}{n-i} \left((h_1 - 1) \left(\frac{h_1}{2\lambda} + \frac{k}{\lambda} \right) + (k - (i-1)) \frac{k-1}{2\lambda} \right) \tag{25}$$

if $0 < h_1 \leq N_l + 1$.

Again, the above holds for both the cases of early arrival and late arrival of h_1. In the case of early arrival, the value of $k = N_h$, whereas in the case of late arrival, $k = N_l + 1 - h_1$.

Now, to obtain a measure of the average delay of high- and low-priority packets, the next step is to weight the delay results obtained for every possible burst times the probability of a packet to to actually appear in such burst. For high-priority packets, this is:

$$\begin{aligned} D^{hp}_{(N_l,N_h)} &= \frac{\sum_{i=1}^{N_h+1}\sum_{n=N_h+1}^{N_l} D^{hp}_{B_{i,n}|h_1=n-N_h} iP(B_{i,n}) + \sum_{h_1=N_l-N_h}^{N_l+1}\sum_{i=1}^{N_l+1-h_1} D^{hp}_{B_{i,N_l+1}|h_1} iP(B_{i,N_l+1},h_1)}{\sum_{i=1}^{N_h+1}\sum_{n=N_h+1}^{N_l} iP(B_{i,n}) + \sum_{h_1=N_l-N_h}^{N_l+1}\sum_{i=1}^{N_l+1-h_1} iP(B_{i,N_l+1},h_1)} \\ &= \frac{1}{\mathbb{E}(L^{hp})}\left(\sum_{i=1}^{N_h+1}\sum_{n=N_h+1}^{N_l} D^{hp}_{B_{i,n}|h_1=n-N_h} iP(B_{i,n}) + \sum_{h_1=N_l-N_h}^{N_l+1}\sum_{i=1}^{N_l+1-h_1} D^{hp}_{B_{i,N_l+1}|h_1} iP(B_{i,N_l+1},h_1)\right) \end{aligned} \tag{26}$$

since the value in the denominator is equal to the average number of high-priority packets $\mathbb{E}(L^{hp})$. Following the same reasoning for the low-priority packets:

$$\begin{aligned} D^{lp}_{(N_l,N_h)} &= \frac{1}{\mathbb{E}(L^{lp})}\left(\sum_{i=1}^{N_h+1}\sum_{n=N_h+1}^{N_l} D^{lp}_{B_{i,n}|h_1=n-N_h}(n-i)P(B_{i,n}) + \right. \\ &+ \left. \sum_{h_1=N_l-N_h}^{N_l+1}\sum_{i=1}^{N_l+1-h_1} D^{lp}_{B_{i,N_l+1}|h_1}(N_l+1-i)P(B_{i,N_l+1},h_1) + D^{lp}_{B_{0,N_l+1}}(N_l+1)P(B_{0,N_l+1})\right) \end{aligned} \tag{27}$$

These give a measure of the average delay per packet of high- and low-priority packets in a burst.

The equations derived above for the average delay of high- and low-priority packets take into account all possible cases of bursts and are difficult to handle in practice. The following sections provide a number of approximations to such equations that are very close to real values in certain scenarios. Their validity and accuracy shall be analysed in the experiments section.

D. Approximation for $\mathbb{E}(D^{hp}_{(N_l,N_h)})$

First of all, it is worth noticing that, when $N_l >> N_h$, it is very unlikely to have bursts such that their first high-priority packet arrive $h_1 > N_l + 1 - N_h$. The case of late arrival of h_1 is rare and can be removed in the analysis of the average delay for high-priority packets. Accordingly, the average delay can be approximated to:

$$\mathbb{E}(D^{hp}_{(N_l,N_h)}) \approx \frac{1}{N_h p_h + 1}\left(\frac{N_h}{\lambda} + N_h p_h \frac{N_h}{2\lambda}\right) \tag{28}$$

that is, the first packet suffers a delay of $\frac{N_h}{\lambda}$ and an average of $\mathbb{E}(i) = N_h p_h$ subsequent high-priority packets experience a delay of $\frac{N_h}{2\lambda}$. The result is the weight sum of these two values.

In addition to this, if $N_h p_h >> 1$, then the effect of the contribution to the average delay of the first high-priority packet is small compared to the other $N_h p_h$ subsequent high-priority packets. In this case, we can use the following approximation:

$$\mathbb{E}(D^{hp}_{(N_l,N_h)}) \approx \frac{N_h}{2\lambda} \tag{29}$$

where the first packet has been assumed to contribute with $\frac{N_h}{2\lambda}$ delay, just like the other $i-1$ high-priority packets. However, is is worth emphasising that this only holds if $N_h p_h >> 1$.

Finally, for cases of N_l and N_h comparable, the above equation does not apply and the case of late arrival of h_1 cannot just be removed. In such case, if $p_h << 1$, then we can assume that most bursts have none or one high-priority packet only, and h_1 is uniformly distributed accross the burst, such that the following equation approximates better:

$$\mathbb{E}(D^{hp}_{(N_l,N_h)}) \approx \frac{N_h}{\lambda}\frac{N_l - N_h}{N_l} + \frac{N_h}{2\lambda}\frac{N_h}{N_l} \tag{30}$$

where the first term considers that the only high-priority packet lays in the first $N_l - N_h$ positions (early arrival), and the second term regards to the case in which it lays in the N_h latest positions (late arrival).

The accuracy of the three approximations is shown in the validation section.

E. Approximation for $\mathbb{E}(D^{lp}_{(N_l,N_h)})$

Again, the exact value for the average delay of low-priority packets can approximated to a more simple equation under certain assumptions.

For instance, if h_1, that is, a truncated geometric distribution which determines the position of the first high-priority arrival, is approximated by the mean of a geometric distribution with parameter p_h: $\mathbb{E}(h_1) \approx \frac{1}{p_h}$, the final burst size is given by:

$$\min(N_l + 1, N_h + \mathbb{E}(h_1)) = \min(N_l + 1, N_h + 1/p_h)$$

Thus, if p_h small, then most of the packets in the burst are of low priority, and the average delay of them is given by:

$$\mathbb{E}(D^{lp}_{(N_l,N_h)}) \approx \frac{\min(N_l, N_h + 1/p_h)}{2\lambda} \tag{31}$$

In spite of its simplicity, this approximation is only valid if the number of high-priority packets is much smaller than the total size of the burst ($p_h << 1$). A more accurate approximation can be derived as follows:

$$\begin{aligned} \mathbb{E}(D^{lp}_{(N_l,N_h)}) \quad &\approx \quad \frac{\mathbb{E}(h_1) - 1}{\mathbb{E}(h_1) - 1 + N_h(1-p_h)}\left(\frac{\mathbb{E}(h_1) - 1}{2\lambda} + \frac{N_h}{\lambda}\right) + \\ &+ \quad \frac{N_h(1-p_h)}{\mathbb{E}(h_1) - 1 + N_h(1-p_h)}\frac{N_h}{2\lambda} \end{aligned} \tag{32}$$

The justification for this is the following: the first arrivals until h_1 (that is, $h_1 - 1$ packets) are of low-priority always, and after the first high-priority packet has arrived, a total of $N_h(1-p_h)$ low-priority packets arrive. The

former packets see an average delay until h_1 (that is, $\frac{h_1-1}{2\lambda}$), and a fixed value of $\frac{N_h}{\lambda}$ for waiting the subsequent arrivals. Then, the arrivals after h_1 experience a delay of $\frac{N_h}{2\lambda}$ on average. The contributions of the two sets of low-priority packets (before h_1 and after h_1) are then weighted over the total number of low-priority packets in the burst.

However, the two approximations highly depend on h_1, and the approximation of the geometric distribution to its mean: $\mathbb{E}(h_1) \approx 1/p_h$ may lead to inaccurate results. This is analysed in the next section.

F. Validation

This section shows the validity of the equations derived for the burst size and average delay of both high- and low-priority packets under the two-class burst-assembly strategy.

The simulation scenario considers a border OBS node at which incoming packets arrive with rate $\lambda = 5$ packets/sec. A two-class burst-assembly policy with $N_l = 24$ fixed and N_h variable in the range $N_h = 0, \ldots, N_l$ has been considered. With this configuration, figs. 8 and 9 show the mean average assembly delay and average burst size for several values of p_h. The circles, squares and diamonds represent the theoretical values following the equations above, whereas the stars represent the experimental values obtained according to the simulation parameters. In this light, he circles should read as "low-priority", either for average delay or number of packets, whereas the squares are related to "high-priority" of the same metric. The diamonds denote average total burst size. As shown, in all cases, the simulated values are contained within the theoretical shapes (circles, squares and diamonds), which concludes that the theoretical equations derived above perfectly match the experimental results.

Those cases with high values of p_h ($p_h > 0.25$ for instance) are not practical, since in most networking scenarios, the volume of high-priority traffic comprises only a small portion of the total traffic. However, the authors have considered necessary to include simulations of all possible cases for validation purposes, on attempts to show the applicability of the equations derived above over a large range of scenarios.

Fig. 10 shows the accuracy of the approximations of mean average assembly delay for high-priority traffic, assuming several values of p_h. As shown, each equation is valid in a range of cases, as explained. The approximation:

$$\mathbb{E}(D^{hp}_{(N_l,N_h)}) \approx \frac{N_h}{2\lambda}$$

plotted with symbol "+", is very inaccurate and can only be applied to cases where $N_h p_h >> 1$ where the contribution of the delay of the first high-priority packet is small compared to the contribution of the other high-priority packets in the burst. However, such condition typically meets at scenarios with large values of p_h, which are rare in practice. A rather better approximation is:

$$\mathbb{E}(D^{hp}_{(N_l,N_h)}) \approx \frac{1}{N_h p_h + 1}\left(\frac{N_h}{\lambda} + N_h p_h \frac{N_h}{2\lambda}\right)$$

depicted with symbol "x", as explained above. This approximation performs better, but degrades when N_l and N_h are comparable, since it does not take into account the case of late arrival of h_1.

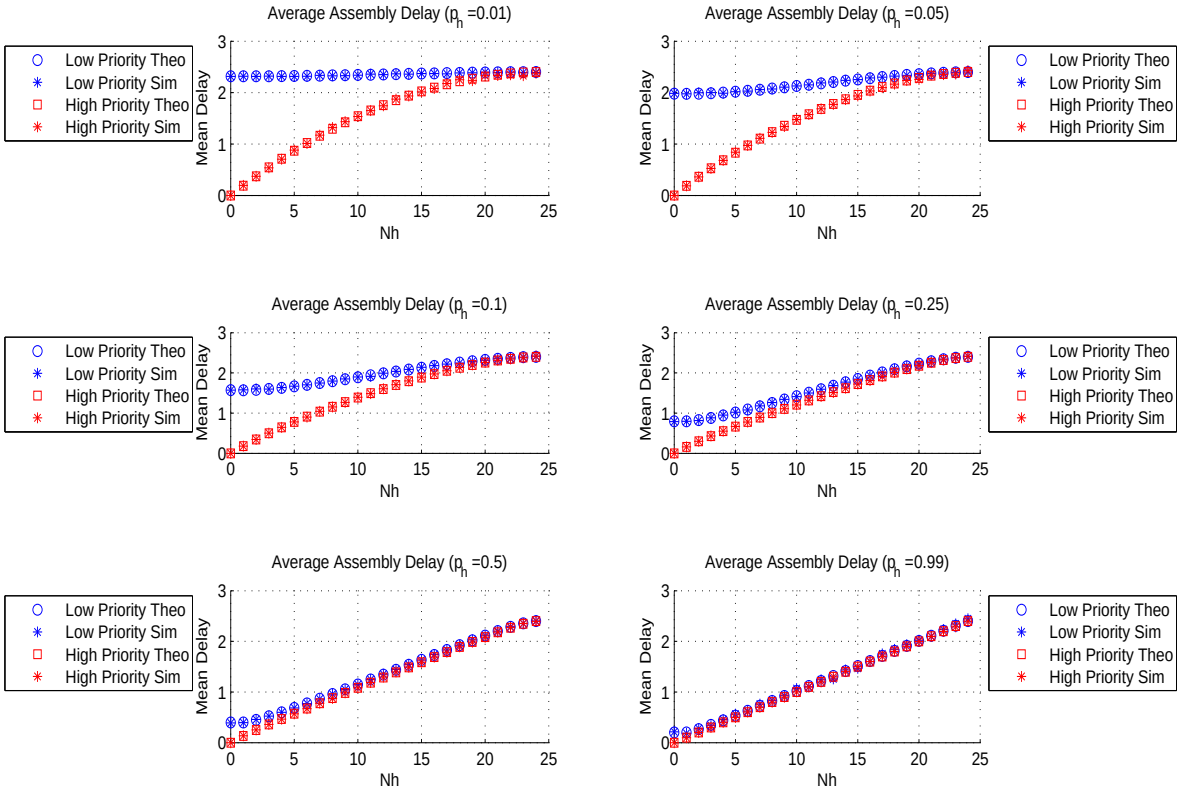

Fig. 8. Mean average delay experienced by the high-priority and low-priority packets:$p_h = 0.01$ (top-left), $p_h = 0.05$ (top-right), $p_h = 0.1$ (middle-left), $p_h = 0.25$ (middle-right), $p_h = 0.5$ (bottom-left), $p_h = 0.99$ (bottom-right)

Finally, the approximation:

$$\mathbb{E}(D^{hp}_{(N_l,N_h)}) \approx \frac{N_h}{\lambda}\frac{N_l - N_h}{N_l} + \frac{N_h}{2\lambda}\frac{N_h}{N_l}$$

depicted with dots (symbol ".") is accurate when $p_h << 1$, such that most bursts constain zero or one high-priority packet only, as explained above.

Concerning the approximation of average delay for low-priority packets, fig. 11 shows the results obtained for the cases $p_h = 0.01$, $p_h = 0.05$, $p_h = 0.1$ and $p_h = 0.5$. The first approximation, i.e.:

$$\mathbb{E}(D^{lp}_{(N_l,N_h)}) \approx \frac{\min(N_l, N_h + 1/p_h)}{2\lambda}$$

is depicted with dots (symbol ".") and the second approximation provided:

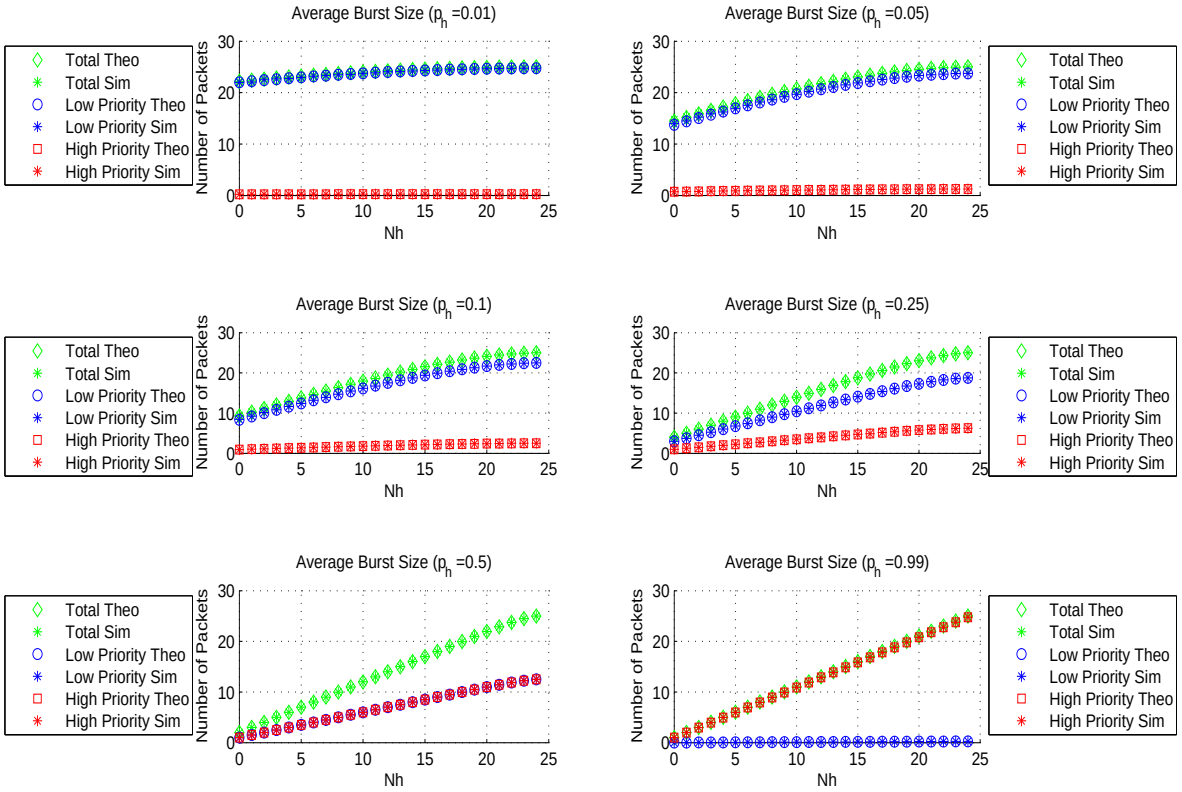

Fig. 9. Average burst size and average number of high- and low-priority packets per burst:$p_h = 0.01$ (top-left), $p_h = 0.05$ (top-right), $p_h = 0.1$ (middle-left), $p_h = 0.25$ (middle-right), $p_h = 0.5$ (bottom-left), $p_h = 0.99$ (bottom-right)

$$\mathbb{E}(D^{lp}_{(N_l,N_h)}) \approx \frac{\mathbb{E}(h_1)-1}{\mathbb{E}(h_1)-1+N_h(1-p_h)}\left(\frac{\mathbb{E}(h_1)-1}{2\lambda}+\frac{N_h}{\lambda}\right)+ + \frac{N_h(1-p_h)}{\mathbb{E}(h_1)-1+N_h(1-p_h)}\frac{N_h}{2\lambda}$$

is shown with crosses (symbol "x"). Clearly, both approximations are valid when either p_h is small (fig. 11 top-left) or large (fig. 11 bottom-right). However, for values of p_h in the range $[0.05-0.1]$, the results obtained with both approximations are far from accurate, as shown. The reason for this is that h_1, which is a truncated geometric distribution, has been approximated by the value $1/p_h$ (the mean of a geometric distribution). The conclusion is that the average delay observed by the low-priority packets highly depends on the arrival of the first high-priority packet, which is variable and cannot be approximated by a fixed value. For this reason, the following numerical example uses the exact value of $\mathbb{E}(D^{lp}_{(N_l,N_h)})$ given by eq. 27 instead of any of the above approximations.

Finally, a few more interesting conclusions can be derived from figure 12. In this figure, the values of $(N_l = 24, N_h = 5)$ are fixed, but the parameter p_h varies in the range $[0,1]$. Obviously, the average assembly delay

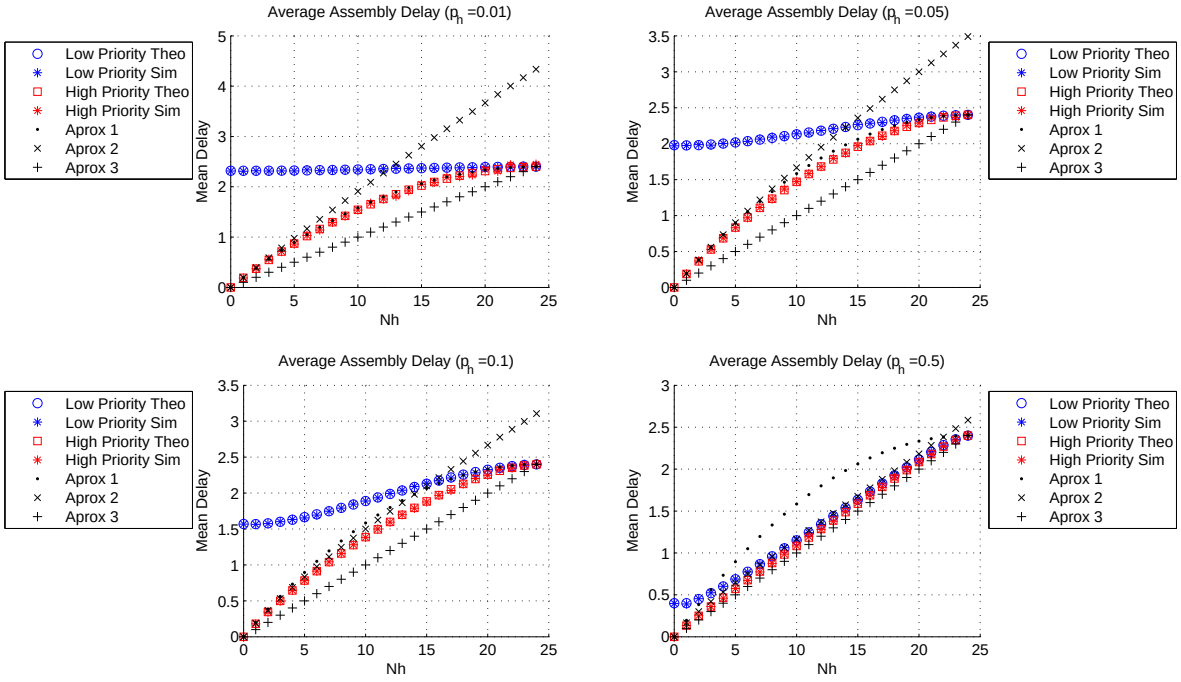

Fig. 10. Accuracy of the assembly delay approximations for high-priority packets with simulation values: $p_h = 0.01$ (top-left), $p_h = 0.05$ (top-right), $p_h = 0.1$ (bottom-left) and $p_h = 0.5$ (bottom-right)

experienced by high-priority packets is smaller than the delay observed by low-priority packets, and the separation between the average delay for each classes is larger the smaller the parameter p_h is. However, when p_h approaches one, there is little differentiation between classes since the first-high priority packet is expected to arrive early in the burst, thus forcing the assembly policy switch to a maximum of N_h packets after the first high-priority packet arrival, which occurred very early. Thus, a small number of low-priority packets are fitted to the less priviledged positions in the burst. Indeed, this can also be seen in fig. 12 bottom since most of the packets in a burst under $p_h \approx 1$ condition are of high-priority.

IV. Numerical example

As proposed in [11], the proportional QoS model assumes that a high-priority class receives a better service than a lower priority class by means of a ratio that can be quantitatively adjusted. In our case, the QoS metric under analysis is the burst-assembly delay and, in order to provide proportional differentiation, the relationship that must be satisfied is the following:

$$\frac{\mathbb{E}(D^{lp}_{(N_l,N_h)})}{\mathbb{E}(D^{hp}_{(N_l,N_h)})} = K \tag{33}$$

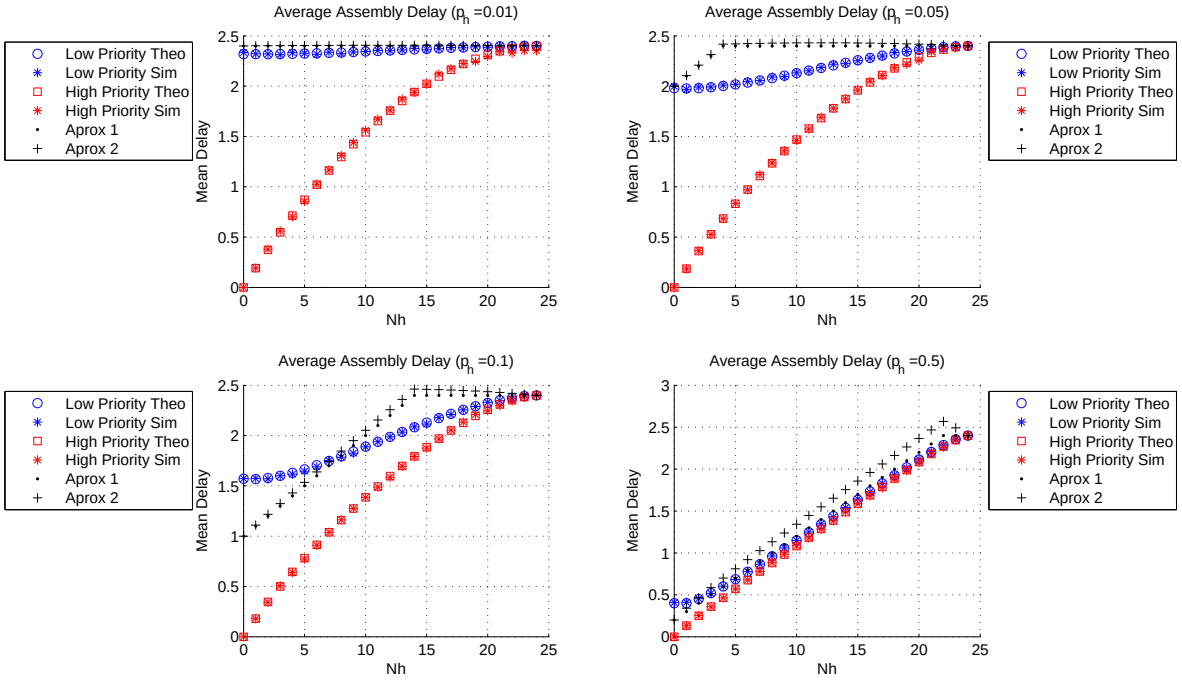

Fig. 11. Accuracy of the assembly delay approximations for low-priority packets with simulation values: $p_h = 0.01$ (top-left), $p_h = 0.05$ (top-right), $p_h = 0.1$ (bottom-left) and $p_h = 0.5$ (bottom-right)

That is, the burst-assembly policy must adjust N_l and N_h such that the average assembly delay observed by the low-priority packets is K times larger than the same value for high-priority packets.

In this light, this numerical example considers a simulation scenario with variable values of λ and p_h as shown in fig. 13.

Essentially, this example considers a typical traffic profile at which the incoming rate of packets λ is relatively small (around 10 packets/unit of time) during the night hours (5 p.m. until 7 a.m.), and substantially grows (until 30 packets/unit of time) within the day (8 a.m. until 7 p.m.). Also, the amount of high-priority packets increases from 1% to 5% over the same period of time.

In this scenario, the border node makes an estimation of the values of λ and p_h, using the well-known Exponentially Weighted Moving Average (EWMA), giving the values of $\hat{\lambda}$ and $\hat{p}_h$, which are used to determine the appropriate parameters N_l and N_h of the two-class burst-assembly policy that meets eq. 33.

The algorithm for adjusting N_l and N_h proceeds as follows. First of all, with the estimates of $\hat{\lambda}$ and $\hat{p}_h$, the algorithm computes $\mathbb{E}(D^{hp})$ and $\mathbb{E}(D^{lp})$ using the approximation given by eq. 30 and the exact expression given by eq. 27 respectively (this is because the approximation for $\mathbb{E}(D^{lp})$ is not accurate). Given a target $\mathbb{E}(D^{lp}_{\text{target}})$ the algorithm tries sevaral values of N_l and N_h such that:

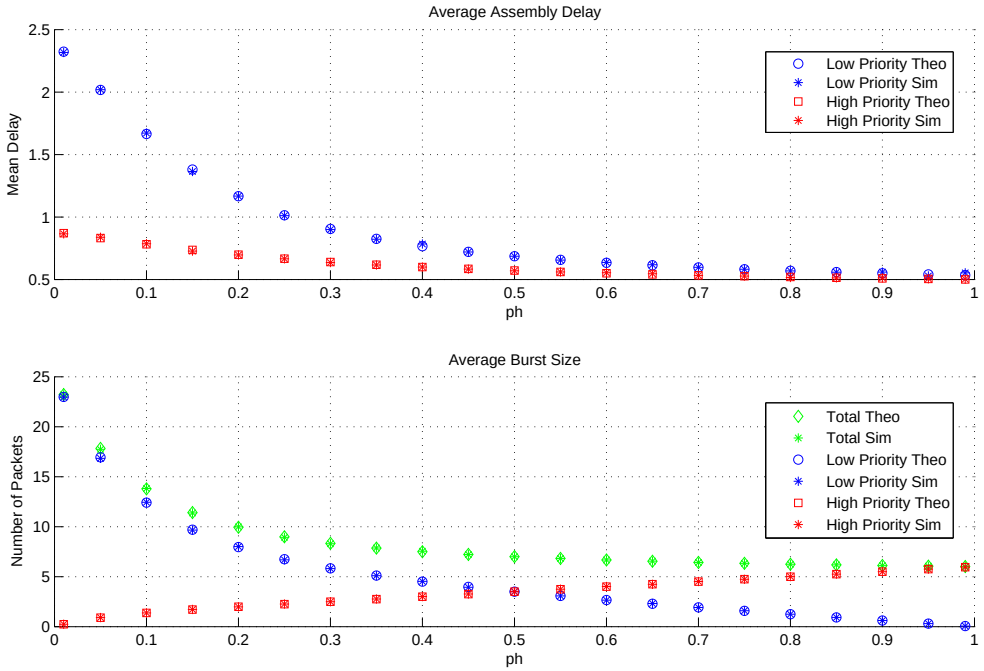

Fig. 12. Average assembly delay (top) and burst size (bottom) under a two-class (N_l, N_h) assembly policy with variable p_h

$$\mathbb{E}(D^{lp}) \approx \mathbb{E}(D^{lp}_{\text{target}})$$

and $\mathbb{E}(D^{hp})$ is adjusted such that:

$$\frac{\mathbb{E}(D^{lp})}{\mathbb{E}(D^{hp})} \approx K$$

After this, the values of $\hat{\lambda}$ and $\hat{p}_h$ are estimated continuously in order to update $\mathbb{E}(D^{lp})$ and $\mathbb{E}(D^{hp})$ in real-time. Then, the sign of:

$$\mathbb{E}(D^{lp}) - \mathbb{E}(D^{lp}_{\text{target}})$$

is used to gradually adjust N_l such that the above equation approaches zero. Then, the sign of

$$K - \frac{\mathbb{E}(D^{lp})}{\mathbb{E}(D^{hp})}$$

is used to gradually adjust N_h such that the above equation also approaches zero as much as possible. With these two mechanisms, the burstifier adjusts N_l and N_h, on attempts to maintain the proportional QoS specified regardless of changes in the network conditions (λ and p_h). This algorithm is summarised in fig. 14.

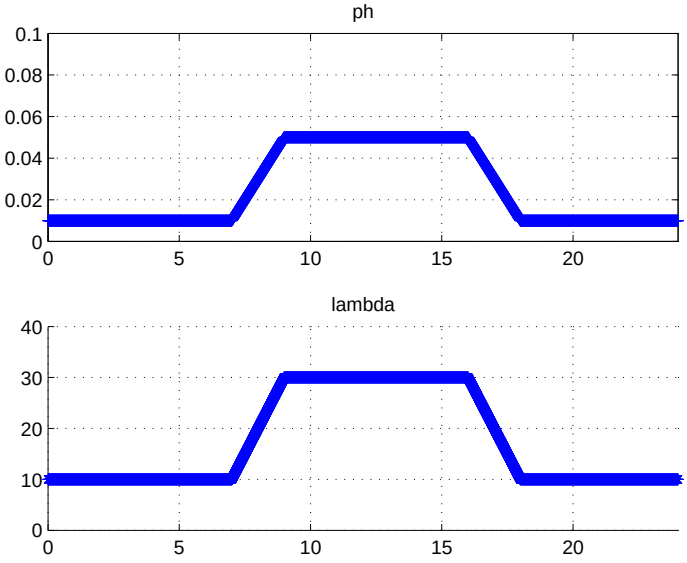

Fig. 13. Variation of environment parameters over time of day: p_h (top), and λ (bottom)

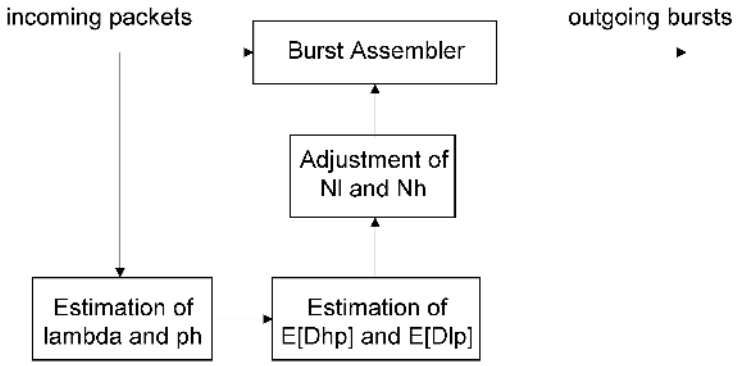

Fig. 14. Diagram of the Algorithm for adjusting N_l and N_h

Fig. 15 shows the values of N_h and N_l estimated by the algorithm in order to guarantee the QoS ratio given by eq. 33. Obviously, during the day, with the increase of λ, both N_h and N_l substantially increase. However, given the fact that the ratio of high-priority increases, in order to maintain the difference between the delay observed by high- and low-priority packets, the value of N_l goes very large ($N_l \approx 30$).

Fig. 16 shows the average delay observed by the high-priority and low-priority packets over time, together with the ratio between them. As shown, such values remain relatively constant over time (around $K = 2$), proving the robustness of the burst assembly algorithm in terms of QoS guarantees regadless of changes in the environment conditions of traffic.

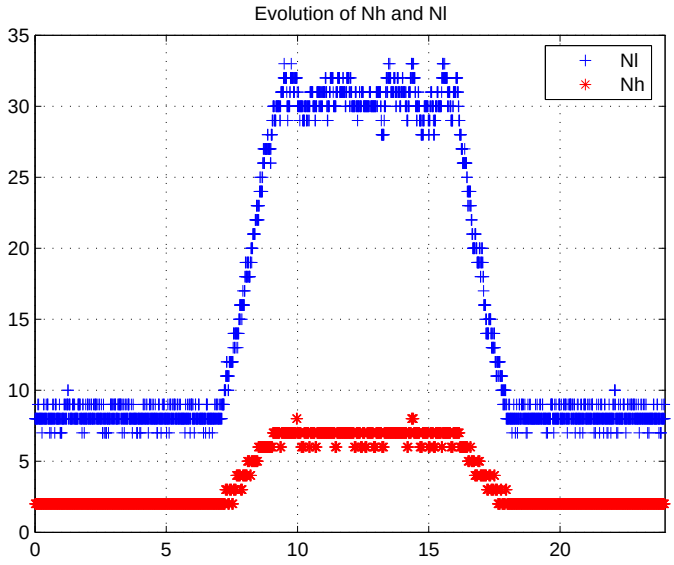

Fig. 15. Variation of N_l and N_h as the algorithm adapts to the changing environment

V. Summary and conclusions

This work presents a novel performance metric to measure the average delay experienced by packets during the burst-assembly process in Optical Burst-Switched networks. Such metric takes into account both the assembly delay of the first packet arrivals (which is relatively high) and that of the later packet arrivals (which is smaller) and performs an average on it. Since packets arrive randomly at the burst assembler, this metric provides a measure of the average burst assembly delay observed by packets arriving at an OBS network.

This work also proposes a new mechanism to improve the burst-assembly delay experienced by high-priority packets, with respect to the assembly delay perceived by low-priority packets, for a typical scenario of two-class traffic. Essentially, such two-class burst-assembly algorithm defines two burst-size limits: N_l and N_h which represent the maximum number of packets that are accepted to arrive after a low- or high-priority arrival respectively. Thus, the algorithm must adjust such limits on attempts to provide *proportional delay differentiation* between the two classes of traffic.

The metrics defined, i.e. average assembly delay as perceived by the high- and low-priority packets are analysed in detail for the two-class assembly algorithm, and exhaustively checked with simulations.

Finally, a numerical example is proposed to show the applicability of such two-class algorithm in a changing environment with variable traffic conditions (incoming traffic rate and percentage of high-priority packets with respect to total). The results show that, if correctly adjusted the burst size limits N_l and N_h, then the two-class burst-assembly algorithm proposed outputs bursts which satisfy, on average, the quality-of-service requirements set a priory, which proportionally benefits high-priority over low-priority packets.

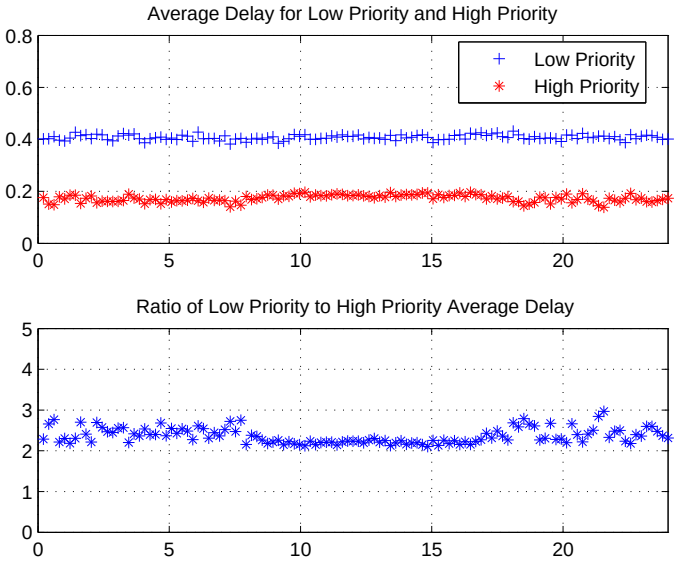

Fig. 16. Variation of average delay for low-priority and high-priority packets and ratio between them

REFERENCES

[1] J. Turner, "Terabit burst switching," *J. High Speed Networks*, vol. 8, pp. 3–16, 1999.

[2] C. Qiao and M. Yoo, "Optical Burst Switching (OBS) – A new paradigm for an optical Internet," *J. High Speed Networks*, vol. 8, pp. 69–84, 1999.

[3] S. Verma, H. Chaskar, and R. Ravikanth, "Optical Burst Switching: A viable solution for terabit IP backbone," *IEEE Network*, vol. 14, no. 6, pp. 48–53, Nov/Dec 2000.

[4] L. Xu, H. G. Perros, and G. Rouskas, "Techniques for optical packet switching and optical burst switching," *IEEE Comms. Magazine*, vol. 39, pp. 136–142, Jan 2001.

[5] Y. Chen, C. Qiao, and X. Yu, "Optical Burst Switching: A new area in optical networking research," *IEEE Network*, vol. 18, no. 3, pp. 16–23, May/June 2004.

[6] V. Vokkarane, K. Haridoss, and J. P. Jue, "Threshold-based burst assembly policies for QoS support in optical burst-switched networks," in *Proceedings of SPIE/IEEE OPTICOMM*, Boston, Massachusetts, July 2002, pp. 125–136.

[7] A. Ge, F. Callegati, and L. S. Tamil, "On Optical Burst Switching and self-similar traffic," *IEEE Comm. Letters*, vol. 4, no. 3, pp. 98–100, March 2000.

[8] X. Cao, J. Li, Y. Chen, and C. Qiao, "Assembling TCP/IP packets in Optical Burst Switched networks," in *Proceedings of IEEE GLOBECOM*, Taipei, Taiwan, Nov. 2002, pp. 2808– 2812.

[9] X. Yu, Y. Chen, and C. Qiao, "Study of traffic statistics of assembled burst traffic in optical burst switched networks," in *Proceedings of SPIE/IEEE OPTICOM*, Boston, Massachusetts, July 2002, pp. 149–159.

[10] J. A. Hernández, J. Aracil, V. López, and J. López de Vergara, "On the analysis of burst-assembly delay in OBS networks and applications in delay-based service differentiation," *Photonic Network Communications (to appear)*, 2007.

[11] Y. Chen, C. Qiao, M. Hamdi, and D. H. K. Tsang, "Proportional differentiation: A scalable QoS approach," *IEEE Comms. Magazine*, vol. 41, no. 6, pp. 52–58, June 2003.

[12] H. Zhang, "Service disciplines for guaranteed performance service in packet-switching networks," *Proc. IEEE*, vol. 83, no. 10, pp. 1374–1396, 1995.

[13] A. Varma and D. Stiliadis, "Hardware implementation of fair queuing algorithms for Asynchronous Transfer Mode networks," *IEEE Comms. Magazine*, vol. 35, pp. 74–80, Dec. 1997.

[14] D. Stiliadis and A. Varma, "Efficient fair queueing algorithms for packet-switched networks," *IEEE/ACM Trans. Networking*, vol. 6, no. 2, pp. 175–185, 1998.

[15] M. Yoo, C. Qiao, and S. Dixit, "QoS performance of Optical Burst Switching in IP over WDM networks," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 2062–2071, 2000.

[16] M. Yoo, C. Qiao, and S. Dixit, "Optical Burst switching for service differentiation in the Next-Generation Optical Internet," *IEEE Comms. Magazine*, vol. 39, no. 2, pp. 98–104, Feb. 2001.

[17] N. Barakat and E. H. Sargent, "Analytical modeling of offset-induced priority in multiclass OBS networks," *IEEE Trans. Communications*, vol. 53, no. 8, pp. 1343–1352, Aug. 2005.

[18] F. Farahmand and J. P. Jue, "Analysis and implementation of look-ahead window contention resolution with QoS support in Optical Burst-Switched networks," *IEEE J. Select. Areas in Communications*, vol. 24, no. 12, pp. 81–93, Dec. 2006.

[19] C. W. Tan, G. Mohan, and J. C.-S. Lui, "Achieving multi-class service differentiation in WDM Optical Burst Switching Networks: A probabilistic preemptive burst segmentation scheme," *IEEE. J. Select. Areas in Communications*, vol. 24, no. 12, pp. 106–119, Dec. 2006.

[20] W. Liao and Loi. C.-H., "Providing service differentiation for optical-burst-switched networks," *IEEE/OSA J. Lightwave Technology*, vol. 22, no. 7, pp. 1651–1660, July 2004.

[21] J. Liu, N. Ansari, and T. J. Ott, "FRR for latency reduction and QoS provisioning in OBS networks," *IEEE J. Selected Areas in Communications*, vol. 21, no. 7, pp. 1210–1219, Sept 2003.

[22] J. A. Hernández and J. Aracil, "On the early release of Burst-Control Packets in Optical Burst Switched networks," in *Proc. Int. Conf. Information Networking*, Estoril, Portugal, Jan. 2007.

[23] T. Karagiannis, M. Molle, M. Faloutsos, and A. Broido, "A nonstationary Poisson view of Internet traffic," in *IEEE INFOCOM*, Honk Kong, PRC, March 2004.

*** Author Biography (250 words max.)**

Pedro Reviriego received the M.Sc. and Ph.D. degrees (Honours) from Universidad Politécnica de Madrid in 1994 and 1997 respectively, both in Telecommunications Engineering. From 1997 to 2000 he was a R&D engineer at Teldat, working on router implementation. In 2000, he joined Massana where he worked on the development of 1000BaseT transceivers, until 2003 when he became a Visiting professor at University Carlos III. From 2004 onwards, he is Distinguished Member of Technical Staff at LSI Corporation. His research interests focus on the performance evaluation of communication networks and the design of physical layer communications devices. He has authored a large number of papers in international conferences and journals, and has also participated in the IEEE 802.3 standardization for 10GBaseT.

*** Author Biography (250 words max.)**

José Alberto Hernández completed the five-year degree in Telecommunications Engineering at Universidad Carlos III de Madrid (Spain) in 2002, and the Ph.D. degree in Computer Science at Loughborough University (United Kingdom) in 2005. After this, he joined the Networking Research Group at Universidad Autónoma de Madrid (Spain), where he actively participates in a number of both national and european research projects concerning the modeling and performance evaluation of communication networks, and particularly the optical burst switching technology. His research interests include the areas at which mathematical modeling and computer networks overlap.

*** Author Biography (250 words max.)**

Javier Aracil received the M.Sc. and Ph.D. degrees (Honors) from Technical University of Madrid in 1993 and 1995, both in Telecommunications Engineering. In 1995 he was awarded with a Fulbright scholarship and was appointed as a Postdoctoral Researcher of the Department of Electrical Engineering and Computer Sciences, University of California, Berkeley. In 1998 he was a research scholar at the Center for Advanced Telecommunications, Systems and Services of The University of Texas at Dallas. He has been an associate professor for University of Cantabria and Public University of Navarra and he is currently a full professor at Universidad Autónoma de Madrid, Madrid, Spain. His research interest are in optical networks and performance evaluation of communication networks. He has authored more than 50 papers in international conferences and journals.

*** Author's Photo (black-and-white preferred)**
Click here to download high resolution image

*** Author's Photo (black-and-white preferred)**
Click here to download high resolution image

*** Author's Photo (black-and-white preferred)**
Click here to download high resolution image