

The Effect of Burst Formation Timers on Video Streaming over Optical Burst Switched Networks

F. Espina, D. Morató, M. Izal and E. Magaña

Public University of Navarre, Campus de Arrosadía s/n, E-31006 Pamplona, Spain

e-mail: {felix.espina, daniel.morato, mikel.izal, eduardo.magana}@unavarra.es

Abstract—Optical Burst Switching networks offer the capacity needed by bandwidth hungry services like video distribution to home subscribers. This paper shows the effect that the burst formation mechanism and its parameters has on video traffic flows. The dependence among video frames in MPEG flows shows a noticeable effect on the total number of frames that the video receiver cannot decode and the duration of the video playback interruptions. By tuning the parameters of the ingress node a quality of service objective can be achieved.

I. INTRODUCTION

Optical Burst Switching (OBS) [1] has received considerable research attention as a promising solution for all-optical transmission of data bursts. It allows the creation of high-speed all-optical networks with present technology, compared with Optical Packet Switching (OPS) that still suffers serious technological limitations [2]. In fact, the number of OBS testbeds and implementations has grown considerably in recent years [3][4][5][6][7][8].

In an OBS network the packets from legacy networks are buffered at the ingress nodes and aggregated into bursts based on Forward Equivalence Classes (FECs). Although one FEC is normally created for each destination egress node, priority classes could be implemented with different FEC per class or class differentiation inside the burst [9][10]. These bursts are optically switched by the core nodes in the network and disassembled at the egress node in order to be relayed to the destination (Fig. 1). A Burst Control Packet (BCP) is created and sent by the ingress node an offset time before the burst is sent. This packet is electronically switched and processed at every backbone node. It contains information that depends on the signaling solution used, for example the burst arrival time, burst size, destination, etc. [11]

A WDM (Wavelength Division Multiplexing) solution is used at the fiber links (Coarse WDM, Dense WDM, ...). A wavelength is reserved for the transmission of the BCPs at each link while the remaining ones are used for data transport. Present existing testbeds use transmission rates above 1 Gbps for each wavelength. Larger transmission rates are expected in the near future and using DWDM the result will be an optical switched network with fiber link capacity easily above 1 Tbps. Such high capacity network will be perfectly suited

This work was funded by Spanish MEC (project CAPITAL TEC2004-05622-C04-04 and project STRRONG TEC2007-62192/TCM)

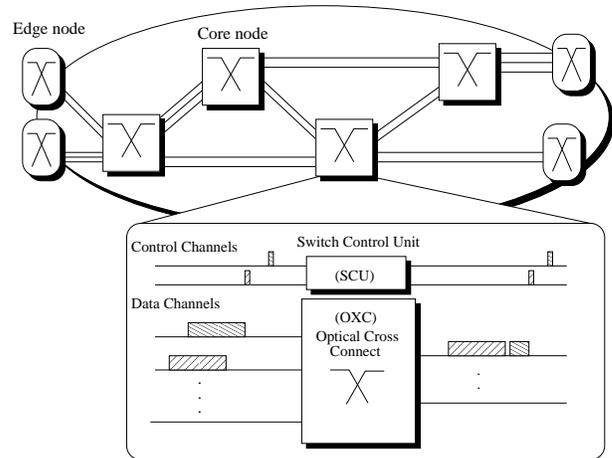


Fig. 1. OBS network architecture

for one of the expected killer services in the near future: the transport of high quality video streams [12]. Nowadays, Internet Service Providers already offer triple-play services containing video streaming flows transported over traditional ATM/SONET/MPLS networks. As the number of subscribers and the video flow requirements grow (High Definition Video) the capacity offered by OBS networks will become essential.

This paper studies the effect of burst creation and transmission on video flows. To the best of the authors' knowledge, no paper has been published that shows the interaction of the burst formation mechanism with the quality requirements for video playback. In [8] an OBS testbed is presented and video flows are transported over it. However, the authors only show that such delay-sensitive services are feasible over an OBS network. This paper offers a study based on simulation of the effect that the burst formation has on the video flow, including the interaction with losses inside the core network. The special characteristics of compressed video traffic have an important impact in video quality as perceived by the user. This paper shows how to decide the best parameters in the burst creation element in order to achieve a target video quality.

The rest of the paper is organized as follows. Section II presents the network scenario and the simulation methodology. Sections III and IV show the effect of network and coded video parameters on the video frame losses that the user suffers. Section V presents the performance from the perspective of

the duration of video playback interruptions. Finally section VI concludes the paper.

II. NETWORK SCENARIO AND METHODOLOGY

Several aspects of the scenario and methodology are presented in this section. First the OBS network scenario, the burst formation mechanism and the loss model for the network are presented. Then special considerations are shown about the effect of losses in modern compressed video. Finally, the simulation tool and video traces used are described.

A. OBS network scenario

The network scenario used in this paper is shown in Fig. 2. The ingress node to the OBS cloud is an IP router with an OBS interface. Inside the OBS interface, the burst formation mechanism (a.k.a. *burstifier*) is implemented. The prevalent *burstifier* types in the literature are timer-based, size-based or a mixture of both timer- and size- based [13][14]. In a timer-based *burstifier* a timer is started on the arrival of a packet to an empty burst formation queue (one for each FEC). When the timer expires, the burst is scheduled for transmission on the output port. In a size-based *burstifier* the burst is sent when the planned minimum size is reached. Mix-timer-size *burstifiers* complete the burst when at least one of the conditions are true: timer expiration or minimum size reached. Other proposals in the literature use predictive or adaptive techniques in order to reduce the delay and losses in the network [15][16][17] or they offer specific mechanisms for QoS support [10][18][19].

The timer-based burstifier is the most frequently used in the literature as it can easily guarantee a maximum delay for the traffic. In this paper, the effect of video coding and timer-based burstifier parameter interaction is studied. The results show how to decide the best configurations that minimize losses and the duration of video playback interruptions.

The legacy networks are reduced in the scenario to a video source in the form of a video streamer server and a video sink (the user or video client). The traffic flow is assumed to be uni-directional from the server to the client. One simultaneous video flow and one FEC per video client is assumed, hence, only one burst is being created at each instant. All the bursts are directed to the same egress node that connects the target

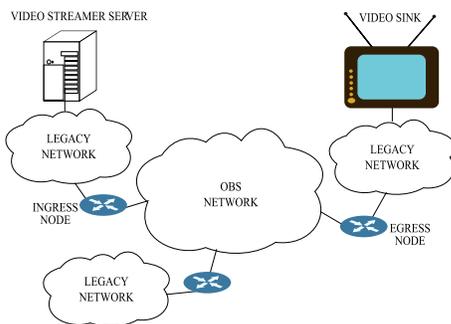


Fig. 2. Simulation network scenario

legacy network. This scenario provides results that are also valid for a multicast network where the bursts are replicated inside the core network to different egress nodes.

As optical buffering in the core nodes will be nonexistent or scarce and implemented using Fiber Delay Lines (FDLs), burst losses will be the result of output port contention that could not be solved by a core node when it received the corresponding BCPs. The optical core network is represented as a black box, modelling the effect of interfering traffic in the video flow as an independent burst loss probability p . This is a common assumption as the losses in the OBS core network are less correlated than in a packet network due to its bufferless nature [20]. The burst loss probability will be a network design parameter, independent on the paths followed by the bursts. Different network scenarios will be evaluated with burst loss probabilities in the range $[10^{-3}, 10^{-1}]$. Other papers have already studied the effect of this range of loss ratios in TCP traffic [4][21][22]. However, for video traffic, a UDP-like transport layer without acknowledgments is more common and is the one selected for the simulations in this paper. It is assumed that there are no losses in the access networks as the objective is to study the effect of the OBS cloud. The video flow rate and the access network capacity are assumed to be much smaller than any optical link's bandwidth. For example, 1 Mbps video flows, 100 Mbps access networks and 10 Gbps wavelengths. Therefore, the number of wavelengths per link will not be relevant and the whole output-port contention problem with the rest of the traffic in the switches will be modeled by the burst loss probability.

B. Video characteristics

Video is nowadays coded using mainly coders from the MPEG family or proprietary but similar in concept ones. MPEG2 and MPEG4 are the preferred solutions for digital compressed video. The MPEG standard defines the data flow that coder and decoder will exchange. However, it does not specify the specific coder/decoder behavior. The objective of the later is to allow for continuous improvements in coder/decoder technology to be compatible with previous ones as they exchange the same data format.

The MPEG standard defines three types of frames [23]: intracoded frames (I-frames), intercoded or predicted frames (P-frames) and bidirectional coded frames (B-frames). I frames can be decoded on their own, however, P-frames depend on the previous I- or P- frame and B-frames depend on the previous I- or P- frame and the following one of either type. This dependence among frames allows for a reduction on frame size keeping the quality, due to similarities among images close in time. The number of frames per second (frame-rate) is constant, however, as the frame size varies, the corresponding bit rate is variable. Thus MPEG video typically leads to variable-bit-rate (VBR) traffic.

I-, P- and B- frames are grouped into GoPs (Groups of Pictures). A GoP is a sequence of frames beginning with an I-frame up to the previous frame to the next I-frame. The

GoP structure is the pattern of I, P and B frames used inside every GoP. A regular GoP structure is usually described as $GxBY$ where x is the number of frames in the GoP and y is the number of contiguous B-frames, for example it could be $G12B2$ or $IBBPBBPBBPBB$. Different GoP sizes and structures are possible, even the change of the GoP structure in a flow. However, most videos are coded using only one GoP structure.

A GoP structure like $IBBPBBPBBPBB$ is shown in *pre-sentation* order, the order in which the frames will be seen by the user. However, as B-frames need the previous and the following I- or P- frame in order to be coded/decoded, the coding/decoding order will be different. Usually, the transmission order is the same as the coding/decoding order. In the previous case, with a $G12B2$ GoP structure, the transmission order will be $IbbPBBPBBPBBiBB$ where the frames in lower-case correspond to frames from the previous or the next GoP. This last example presents the scenario of an open GoP. In an open GoP the last B-frames depend on the I frame from the next GoP. In a closed GoP there is no dependence with frames out of the GoP. A closed GoP ends with a P-frame, like for example in $G9B3$ or $IBBBPBBBP$. The trace files used in the simulations presented in this paper contain open GoPs but the analytical expressions offer results also for closed GoPs.

Although the frames could fit in an IP packet, most of them will be larger than the MTU (Maximum Transmission Unit) in the access network and will be fragmented into several IP packets. The loss in the network of a packet that contains part of a frame produces an incomplete frame at the decoder. The coder behavior in this situation is not defined by the MPEG standard and depends on the specific software implementation. It could drop the whole frame or try to recover the part of the image it contains based on information received in other frames. Due to this dependence on the specific coder a worst case scenario is assumed in this paper, where the receiver drops the whole frame when part of it is lost.

On the other hand, the drop of one frame has serious consequences on other frames in the same GoP as their decoding procedure could depend on the correct reception of the dropped one. Again a worst case scenario is assumed, where the loss of one frame will trigger the loss of any other frame that depends on it. For example, Fig. 3 shows on the left hand side a frame that has been dropped in the network. In the right hand side of the figure all the frames that cannot be decoded due to that loss are highlighted. The loss of a P-frame triggers the loss of all the following frames in the GoP and the B-frames previous to it and after another I- or P- frame.

C. Methodology

The simulation tool used was OMNeT++ [24]. It is a public-source simulation environment with strong GUI support. Using the INET Framework for OMNeT++ it supports TCP/IP networking protocols. New modules that implement the fundamental functionalities in an OBS network have been

created for this work. Also a feeding video source and a sink module were developed.

For the video flows, traces from [25] were used. Table 1 shows the main statistics from these video traffic traces. The video source module reads from a file information about the frame type, size and the time when it should be sent. The source encapsulates as many bytes as the frame size into UDP datagrams and sends them to the video client (the sink module). The network interface will fragment the IP datagram if needed. The access router to the OBS network creates the data bursts from the IP packets and sends the bursts to the egress node. Depending on the *burstifier* parameters and the traffic process, part of a frame, one frame or more than one frame will travel inside each burst. The network drops each burst with an independent probability p . At the far end of the optical network, the egress router disassembles the burst into the IP packets and sends them to the video client (the sink module).

The sink module records information about every packet received. Using several scripts, the statistics shown in the following sections are computed. Frame losses are the result of dropped bursts inside the core optical network due to blocking and are counted at the video client. A dropped burst will contain packets from one or more video frames. As described before, a video frame will be dropped when at least one of its IP packets gets lost. The packet loss ratio will be equal to the burst loss ratio. However, video frame loss ratio will depend on the amount of frames that get affected and consequently dropped due to a burst loss. One lost burst can contain n whole video frames plus part of two more frames in the form of some packets at the beginning and some at the end of the burst. This border effect will become less noticeable as the number of frames inside a burst increases. We name n_f^* as the number of dropped frames due to part of them getting lost inside the network. N_f is the total number of frames in the video. The *frame loss ratio* (FLR) is measured as the quotient $FLR = n_f^*/N_f$. All the frames that could not be decoded due to losses and inter-frame dependence are counted into n_{tf}^* . The *frame starvation ratio* (FSR) is calculated as $FSR = n_{tf}^*/N_f$. It is evident that $FSR \geq FLR$.

In this paper, not only the FSR is considered as quality measurement. The duration of the video playback interruptions cannot be concluded from its value, hence, these lengths will also be computed and estimated analytically in the following

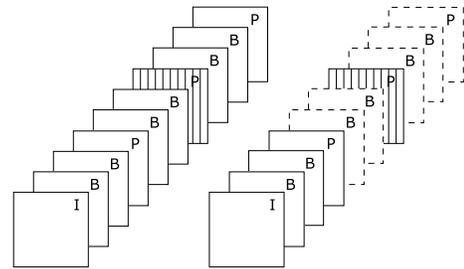


Fig. 3. Example of Inter-frame dependence in a GoP

TABLE 1

Trace file	VIDEO TRACES CHARACTERISTICS						Mean frame size, F_s (KBytes)	Mean bit rate (Kbps)
	GoP	fps	Frames N_f	Size, S (MBytes)	Duration (min)			
Lord of the Rings III (LOTR3)	G12B2	25	289084	984	190	3.5	714	
Matrix (MAT)	G12B2	25	199255	383	133	1.97	403	
Tokyo Olympics (TOK)	G16B7	30	133121	564	74	4.3	1065.8	

sections.

III. FRAME LOSS AND FRAME STARVATION RATIOS

A timer-based burstifier starts a timer when it receives the first packet for a FEC and it schedules the burst for transmission when the timer expires. The video server sends all the IP packets that form a frame at the maximum link speed, hence, the first one starts the timer in the burstifier and all of them get into the same burst (unless the timer has a value smaller than the transmission time of the whole frame). As the timer gets larger more frames get into the same burst. It will be very unlikely that the timer expires while receiving a frame, but it will normally expire in the time between two frames. Hence, an integer number of frames gets into each burst and that number depends on the timer value.

The most frequent frame-rates are around 25-30 frames per second. Therefore the inter-frame time is around 33-40 msec. For a movie file containing 25 frames per second and a GoP structure 12-frames large, the inter-frame time is $t_{if} = 40msec$ and the inter-GoP time (the GoP duration) is $t_{ig} = 12 \times 40 = 480msec$.

Fig. 4 shows the frame loss ratio (FLR) and the frame starvation ratio (FSR) versus the timer value t_{out} in the lower x-axis. The 95% confidence intervals are also presented. The top x-axis shows the number of frames per burst and it is $f_b = \lceil t_{out}/t_{if} \rceil$. The x-axis range is large enough to cover the GoP duration in order to show the effect of its structure. The longest timer duration considered in the figure is below $500msecs$, still reasonable for live multicast and comparable with coding and distribution delays. The number of bursts in the whole video is $N_b = N_f/f_b$. The number of lost bursts will be named as n_b^* and the number of lost frames can be computed as $n_f^* = n_b^* f_b$. Therefore, $FLR = n_f^*/N_f = n_b^* f_b / (N_b f_b) = n_b^*/N_b = p$, and the FLR is equal to the burst loss probability and independent of the timer value, as Fig. 4 shows.

The number of frames that could not be decoded depends on the inter-frame dependence, therefore on the GoP structure. When a frame gets lost, the number of frames that cannot be decoded depends on the specific frame lost inside the GoP. Fig 4 shows that the FSR is different from the FLR. As the timer value grows larger the FSR goes to the FLR and therefore to the burst loss probability. For small timer values the FSR is much larger than the FLR. The following shows an approximation for the FSR in a network scenario with a burst loss probability p and a video with a known GoP structure.

We define the random variable X_1 as *the number of frames that cannot be decoded due to a frame loss*. We are interested in the average value $E[X_1]$. Let's assume a negligible probability for the events of more than one lost burst carrying frames from the same GoP or from adjacent GoPs. We define random variable Y as *the position inside a GoP of a lost frame*, counting as $Y = 1$ the event of the I-frame being lost and we assume all these events as equally probable.

Let's calculate the probability distribution function of the conditional random variable X_1/Y . $P(X_1/(Y = i) = n)$ is the probability of not being able to decode exactly n frames due to the loss of frame number i in a GoP. For example, if $i = 1$ the I-frame gets lost and the whole GoP cannot be decoded. If the GoP structure is an open one then even the last B-frames from the previous GoP are useless. For a GoP structure $GxBY$ where $x = 12$ and $y = 2$ (G12B2) this event means 14 frames not decoded, therefore $P(X_1/(Y = 1) = n) = \delta_{n,14}$ where $\delta_{i,j}$ is the Kronecker delta that verifies $\delta_{i,j} \equiv 1$ when $i = j$ and 0 otherwise. The values of $P(X_1/(Y = i) = n)$ can be computed based only on the GoP structure. The key value is the number of frames lost due to frame i being dropped. We name this value as $g_1(x, y, i)$. Based on the equal probability of events from Y then $P(X_1 = n)$ can be easily expressed with (1).

$$P(X_1 = n) = \frac{1}{x} \sum_{i=1}^x P(X_1/(Y = i) = n) = \frac{1}{x} \sum_{i=1}^x \delta_{n, g_1(x, y, i)} \quad (1)$$

Then $E[X_1]$ can be easily computed from (2).

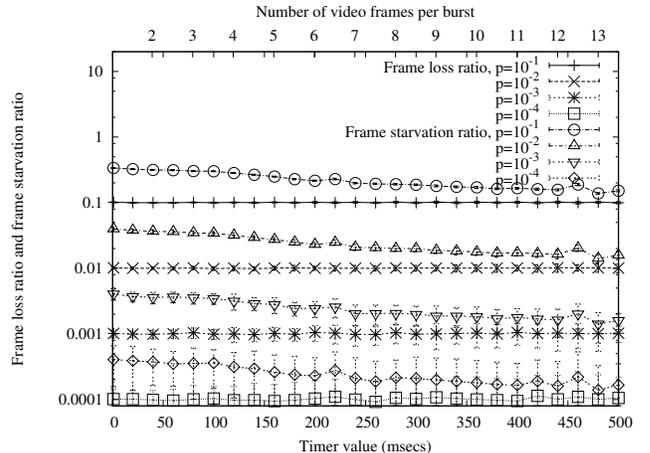


Fig. 4. Losses for LOTR3 trace using a timer-based burstifier

$$E[X_1] = \sum_{n=1}^{\infty} nP(X_1 = n) = \frac{1}{x} \sum_{n=1}^{\infty} \sum_{i=1}^x n\delta_{n,g_1(x,y,i)} \quad (2)$$

This procedure can be extended to the case of a lost burst that contains f_b frames. In this situation f_b consecutive frames get dropped. Random variable X_{f_b} is defined as *the number of frames that cannot be decoded due to the loss of f_b consecutive frames*. Following the previously explained procedure the values of a function $g_{f_b}(x, y, i)$ are computed. Function $g_{f_b}(x, y, i)$ provides the number of frames lost due to a block of f_b frames, starting with frame i in the GoP, getting lost. $g_{f_b}(x, y, i)$ depends only on the GoP structure. $E[X_{f_b}]$ is computed using (3).

$$E[X_{f_b}] = \frac{1}{x} \sum_{n=1}^{\infty} \sum_{i=1}^x n\delta_{n,g_{f_b}(x,y,i)} \quad (3)$$

Table 2 shows the values of $E[X_{f_b}]$ for several GoP structures.

TABLE 2

AVERAGE LOSSES AND CUT LENGTH DUE TO INTER-FRAME DEPENDENCE

f_b	$E[X_{f_b}]$			D_{f_b}		
	G12B2	G16B7	G16B14	G12B2	G16B7	G16B14
1	3.83	3.25	2.81	3.83	3.25	2.81
2	7.33	6.37	4.62	5.5	5.66	4.35
3	10.5	9.37	6.37	6.3	7.5	5.66
4	11.5	12.25	8.06	6.9	8.9	6.78
5	12.5	15	9.68	7.5	10.0	7.75
6	13.5	17.62	11.25	8.1	10.84	8.57
7	14.5	20.12	12.75	8.7	11.5	9.27
8	15.5	22.5	14.18	9.3	12	9.86
9	16.5	23.5	15.56	9.9	12.53	10.37
10	17.5	24.5	16.87	10.5	13.06	10.8
11	18.5	25.5	18.12	11.1	13.6	11.15
12	19.5	26.5	19.31	11.7	14.6	11.44

Finally, let's estimate the number of frames that cannot be decoded as $n_{tf}^* = E[X_{f_b}]n_b^*$. Then the FSR can be computed using (4).

$$FSR = \frac{n_{tf}^*}{N_f} = \frac{E[X_{f_b}]n_b^*}{N_b f_b} = \frac{E[X_{f_b}]}{f_b} p \quad (4)$$

Fig. 5 shows that the estimation follows closely the simulation results. The theoretical result provides an overestimation of the FSR because the situations with more than one burst loss containing frames from the same GoP have been ignored. For example two lost bursts could contain frames from the same GoP. There is a high probability that some frames dropped due to the first loss were the same as some not decoded due to the second one. Therefore the real FSR will be smaller than the computed one.

As the burst loss probability gets smaller the probability of more than one loss per GoP gets negligible and the FSR estimation gets more accurate.

The results were checked with movie files that use different GoPs and frame-rates. Fig. 6 shows the FSR for the movie trace file from *Lord of the Rings III* that used G12B2 and

for *Tokyo Olympics* that uses G16B7. The 95% confidence intervals are also presented and the analytical results match quite well the simulations in both situations.

Fig 6 shows that using the same timer value better results (lower losses) can be obtained by coding the movie with the proper GoP structure. For example, a target FSR quality value of 0.025 (2.5% of frames not decoded) in an OBS network that drops 1% of the bursts cannot be achieved with a timer value of 250msec and a GoP structure G16B7 but it is obtained with a G12B2 (see again Fig. 6). It must be noted that the GoP structure can be changed keeping the same flow rate by adjusting the coding quality. The analytical results on FSR shown in Fig. 6 are not always monotonically decreasing with the timer value. It must be understood that this behaviour is due to the relationship between the timer value and the inter-frame interval. This relationship results in a slightly greater timer producing bursts with a greater number of packets inside when it includes a new video frame. On the following section, a comparison of different GoP structures and their effect on the frame starvation ratio is provided.

IV. THE EFFECT OF THE GOP STRUCTURE

Using the procedure detailed in (3) and (4), for a GoP structure $GxBY$ the FSR can be obtained. For any network scenario there will exist GoP structures that will provide or not the target FSR. $E[X_{f_b}]$ is the component of FSR that depends on the GoP structure. Fig. 7 shows $E[X_{f_b}]$ for the different possible structured GoPs with size 12 frames. Some (x, y) GoP parameter combinations are eliminated as they result in a nonregular GoP. For example, G12B6 would be *IBBBBBBPBBBB* where the second block of B-frames cannot be 6 due to the GoP size limit. When $x = 12$ only the values $y \in \{1, 2, 3, 5, 10, 11\}$ are used. Ignoring the special case G12B10 that will be commented later, the FSR for rest of the GoPs follows two slopes, one for values of f_b lower than the number of consecutive B-frames y and the other for greater values. In general, the GoPs with a lesser number of

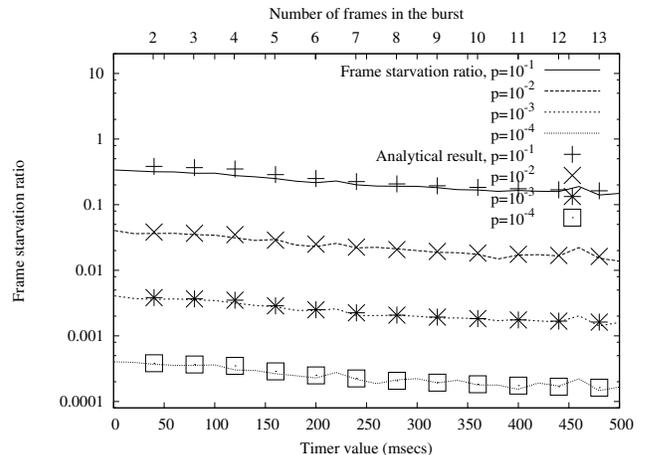


Fig. 5. Analytical estimation of the FSR for LOTR3 trace file

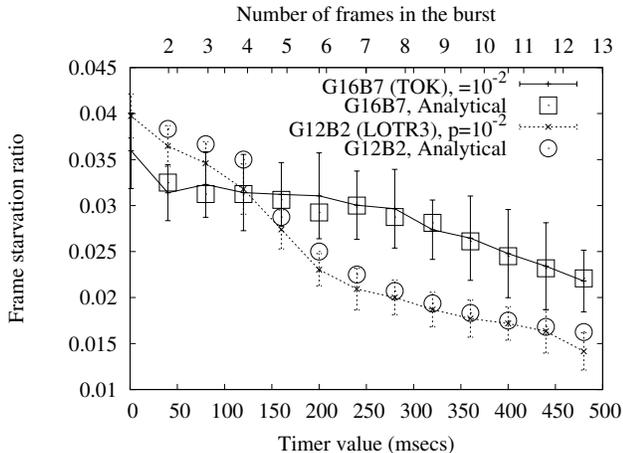


Fig. 6. Analytical estimation of the FSR for different GoP structures and $p = 10^{-2}$

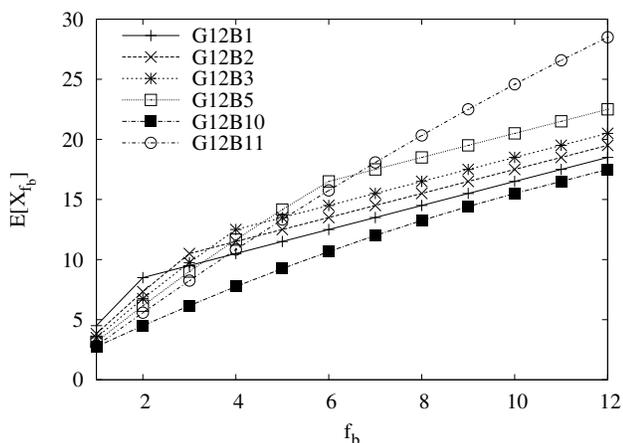


Fig. 7. $E[X_{f_b}]$ for a $G12Bf_b$ structure

B-frames have a lower FSR when f_b grows. This is due to the smaller number of B-frames that get lost due to a P-frame loss. The exception is GoP $G12B10$ that is the only closed GoP with $x = 12$. Losses in a closed GoP do not affect other GoPs resulting in the lowest $E[X_{f_b}]$, therefore the lowest FSR. Similar results are obtained with different GoP sizes.

For a large burst loss ratio, closed GoPs offer the lowest FSR. Fig. 8 compares the closed GoPs for sizes $x \in \{10, 12, 16\}$. For a fixed x and as f_b increases, $E[X_{f_b}]$ goes to the same value independent of the number y of B-frames. Also, the lowest value of $E[X_{f_b}]$ for a fixed x is always obtained with the largest number of consecutive B-frames. Comparing different GoP sizes, for large f_b the shorter the GoP the better. For small values of f_b there is not a clear winner. However, using a closed GoP structure instead of a similar open one results in larger bit rates for the same video quality as there is no possibility to use the temporal correlation in order to reduce the size of the frames in an adjacent GoP.

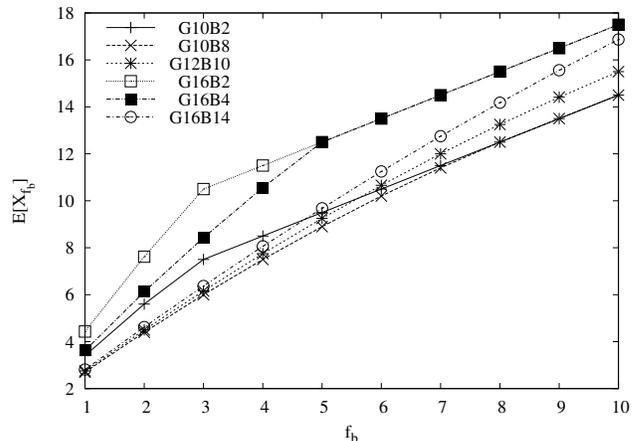


Fig. 8. $E[X_{f_b}]$ for the closed GoPs with $x \in \{12, 16\}$

V. VIDEO CUTS

We name a *video cut* as a sequence of consecutive frames in presentation order that could not be decoded. We define function $h_{f_b}(x, y, i, j)$ as the characteristic cut length function of the GoP structure. $h_{f_b}(x, y, i, j) = 1$ iff for a GoP $GxBy$ when there is a loss of f_b consecutive frames starting with frame i in the GoP then there will be a video cut of length j frames. For example, in a GoP structure $G12B2$, or $IbbPBBPBBPBBiBB$ in transmission order, if $f_b = 2$ and $i = 9$ the bold frames get lost: $IbbPBBPBBPBBiBB$. As a P-frame gets lost, all the following frames in the GoP cannot be decoded. In presentation order, not decoded frames are: $IBBPBBPBBPBB$. There are two independent cuts, the first one is 1 frame-long and the other one contains 5 consecutive frames, therefore, $h_2(12, 2, 9, 1) = 1$, $h_2(12, 2, 9, 5) = 1$ and $h_2(12, 2, 9, j) = 0, \forall j \in N \setminus \{1, 5\}$. We define D_{f_b} as the average cut length when f_b consecutive frames get lost. Let's consider all the possible loss situations inside the GoP as equally probable, hence, D_{f_b} can be expressed using (5). Values of D_{f_b} for some GoP structures are shown in Table 2.

$$D_{f_b} = \frac{\sum_{j=1}^{\infty} j \sum_{i=1}^x h_{f_b}(x, y, i, j)}{\sum_{j=1}^{\infty} \sum_{i=1}^x h_{f_b}(x, y, i, j)} \quad (5)$$

Fig. 9 shows the average cut lengths from the simulations with LOTR3 movie file and computed using (5) for a $G12B2$. The approximation matches the simulation results better as the burst loss probability gets smaller. For large p there is a higher probability that two losses are in the same GoP or adjacent GoPs with the result of a larger video cut. This scenario has not been considered in the estimation of D_{f_b} and is not very interesting as the network will be designed for a low loss ratio. For $p = 10^{-4}$ the results match quite well the simulations. The average duration of the interruption measured in time units is directly obtained multiplying the value from (5) by the inter-frame time.

Fig. 10 compares D_{f_b} as obtained from (5) for different closed GoP structures. Closed GoPs are selected because the

video cut duration is limited to the GoPs that contain frames dropped in the network. For any GoP structure, as the timer value grows, more frames get into each burst, hence, when a burst is lost the video playback interruption lasts longer. For larger GoP structures (larger x value) video cuts can achieve larger durations, reaching the GoP length. Therefore, the shorter the GoP the better, however, again this means larger bit rates as the number of I-frames will increase.

Finally, let's estimate the total amount of not decoded video as $n_b^* D_{fb}$. The proportion of Movie Not Shown, or MNS , is approximated with (6). Fig. 11 shows the MNS from simulation compared with the approximation.

$$MNS = \frac{n_b^* D_{fb}}{N_f} = \frac{n_b^* D_{fb}}{f_b N_b} = p \frac{D_{fb}}{f_b} \quad (6)$$

As the timer value grows the amount of movie lost gets reduced. With large timers there are more frames per burst, hence the video interruptions last longer, however also the losses (the FSR) gets reduced. The reduction in the FSR compensates the increase in video cut duration, with a net decrement in the amount of movie that could not be decoded.

VI. CONCLUSIONS

This paper has shown that fine tuning of the parameters in a timer-based burstifier for an OBS network must take into account the special characteristics of compressed video traffic. There is an important dependence between the video GoP structure, the timer value in the burstifier and the performance results. The results obtained for this dependence were intuitive, however, we have also offered an analytical methodology to compute the exact expected frame loss ratio and the duration of video interruptions. This methodology allows the network designer to select the video GoP structure and timer burstifier that provides an objective quality result measured based on those two parameters.

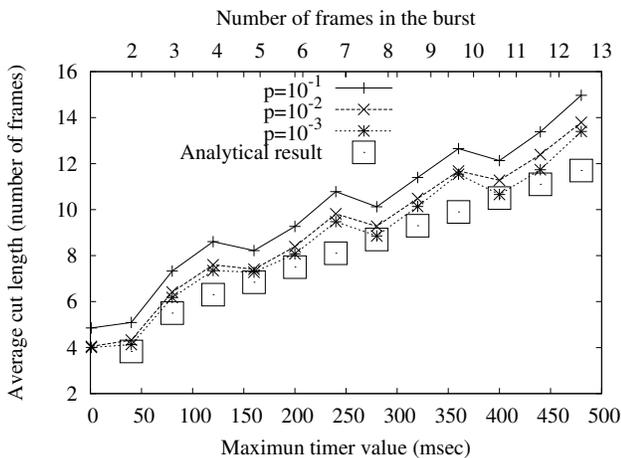


Fig. 9. D_{fb} for LOTR3 trace file ($G12B2$ GoP structure)

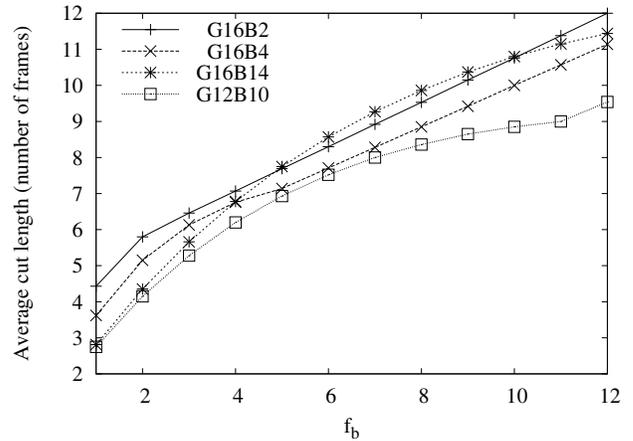


Fig. 10. Analytical D_{fb} for different closed GoPs

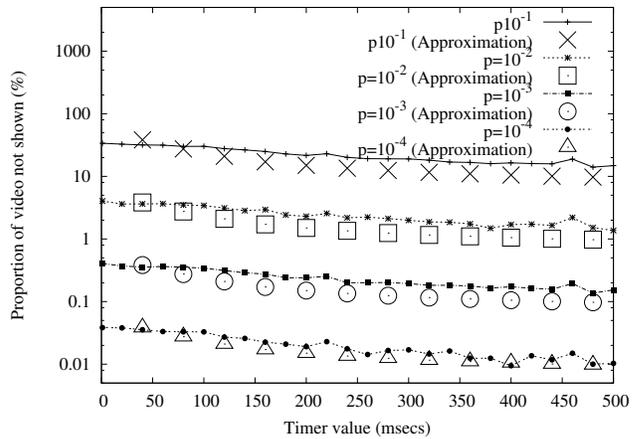


Fig. 11. Percentage of video not received (LOTR3 trace file)

ACKNOWLEDGMENTS

The authors would like to thank Augusto Mauch for his contribution to the development of the modules for the simulation of OBS networks in OMNeT++. They also want to thank the fruitful discussions during the meetings of COST action 291 "Towards Digital Optical Networks".

REFERENCES

- [1] C. Qiao and M. Yoo, "Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet," *Journal of High-Speed Networks*, vol. 8, no. 1, 1999.
- [2] S. Yao, B. Mukherjee, and S. Dixit, "Advances in Photonic Packet Switching: An Overview," *IEEE Communications Magazine*, vol. 38, no. 2, pp. 84-94, February 2000.
- [3] A. Campi, W. Cerroni, F. Callegati, G. Zervas, R. Nejabati, and D. Simeonidou, "SIP Based OBS networks for Grid Computing," in *LNCs 4534, Proceedings of ONDM*, Athens, Greece, May 2007.
- [4] W. Zhang, J. Wu, J. Li, W. Minxue, and S. Jindan, "TCP Performance Experiment on LOBS Network Testbed," in *LNCs 4534, Proceedings of the 11th International IFIP TC6 Conference, ONDM*, I. Tomkos, F. Neri, J. Sole, X. Masip, and S. Sanchez, Eds., Athens, Greece, May 2007, pp. 186-193.
- [5] J. Kim, J. Cho, M. Jain, D. Gutierrez, L.G. Kazocsky, C.F. Su, R. Rabbat, and T. Hamada, "Demonstration of a 2.5 Gbps Optical Burst Switched WDM Ring Network," in *Proceedings of OFC/NFOEC*, March 2006.

- [6] Y. Sun, T. Hashiguchi, V.Q. Minh, Xi. Wang, H. Morikawa, and T. Aoyama, "A Burst-Switched Photonic Network Testbed: Its Architecture, Protocols and Experiments," *IEICE Transactions on Communications*, vol. E88-B, no. 10, pp. 3864–3873, October 2005.
- [7] Z. Gao, Y. Qiao, Y. Ji, and T. Saito, "The Optical Burst Switching Ring Network Using AOTF to Drop Data Burst," *Journal of Optical Communications*, vol. 26, no. 6, pp. 255–259, 2005.
- [8] Y. Sun, T. Hashiguchi, V.Q. Minh, X. Wang, H. Morikawa, and T. Aoyama, "Design and Implementation of an Optical Burst-Switched Network Testbed," *IEEE Communications Magazine*, vol. 43, no. 11, pp. S48–S55, November 2005.
- [9] M.d.V. Rodrigo and M.A. Remiche, "Planning OBS Networks with QoS Constraints," *Photonic Network Communications*, vol. 14, no. 2, pp. 229–239, October 2007.
- [10] Vinod M. Vokkarane and Jason P. Jue, "Prioritized Burst Segmentation and Composite Burst-Assembly Techniques for QoS Support in Optical Burst-Switched Networks," *IEEE Journal on Selected Areas in Communications*, vol. 21, no. 7, pp. 1198–1209, September 2003.
- [11] Jing Teng and George N. Rouskas, "A Comparison of the JIT, JET and Horizon Wavelength Reservation Schemes on a Single OBS Node," in *Proceedings of the First International Workshop on Optical Burst Switching (WOBS)*, Dallas, Texas, October 2003.
- [12] Y. Xiao, X. Du, J. Zhang, F. Hu, and S. Guizani, "Internet Protocol Television (IPTV): The Killer Application for the Next-Generation Internet," *IEEE Communications Magazine*, vol. 45, no. 11, pp. 126–134, November 2007.
- [13] J.Y. Choi, J.S. Choi, and M. Kang, "Dimensioning Burst Assembly Process in Optical Burst Switching Networks," *IEICE Transactions on Communications*, vol. E88-B, no. 10, pp. 3855–3863, October 2005.
- [14] Xiang Yu, Yang Chen, and Chunming Qiao, "A Study of Traffic Statistics of Assembled Burst Traffic in Optical Burst Switched Networks," in *Proceedings of SPIE Opticomm*, Boston, July 2002, SPIE, pp. 149–159.
- [15] Xiaojun Cao, Jikai Li, Yang Chen, and Chunming Qiao, "Assembling TCP/IP Packets in Optical Burst Switched Networks," in *Proceedings of GLOBECOM'02*, November 2002, pp. 2808–2812, IEEE.
- [16] D. Morato, J. Aracil, L.A. Diez, M. Izal, and E. Magana, "On Linear Prediction of Internet Traffic for Packet and Burst Switching Networks," in *Proceedings of the International Conference on Computer Communications and Networks (ICCCN 2001)*, Scottsdale, Arizona, October 2001.
- [17] A. Sideri and E.A. Varvarigos, "New Assembly Techniques for Optical Burst Switched Networks Based on Traffic Prediction," in *LNCS 4534, Proceeding of ONDM*, Athens, Greece, May 2007, pp. 358–367.
- [18] P. Reviriego, J.A. Hernandez, and J. Aracil, "Analysis of Average Burst-Assembly Delay and Applications in Proportional Service Differentiation," *Photonic Network Communications*, vol. 14, no. 2, October 2007.
- [19] M. Izal, J. Aracil, D. Morato, and E. Magana, "Delay-throughput curves for timer-based OBS burstifiers with light load," *IEEE Journal of Lightwave Technology*, vol. 24, no. 1, pp. 277–285, January 2006.
- [20] Xiang Yu, Jikai Li, Xiaojun Cao, Yang Chen, and Chunming Qiao, "Traffic Statistics and Performance Evaluation in Optical Burst Switched Networks," *IEEE Journal of Lightwave Technology*, vol. 22, no. 12, pp. 2722–2738, December 2004.
- [21] G. Gurel and E. Karasan, "Effect of Number of Burst Assemblers on TCP Performance in Optical Burst Switching Networks," in *Proceedings of BROADNETS*, San Jose, CA, October 2006, pp. 1–7.
- [22] S. Gowda, R.K. Shenai, K.M. Sivalingam, and H.C. Cankaya, "Performance evaluation of TCP over optical burst-switched (OBS) WDM networks," in *Proceedings of ICC*, May 2003, vol. 2, pp. 1433–1437.
- [23] D.P. Heyman and T.V. Lakshman, "Source Models for VBR Broadcast-Video Traffic," *IEEE/ACM Transactions on Networking*, vol. 4, no. 1, pp. 40–48, February 1996.
- [24] A. Varga, "The OMNeT++ Discrete Event Simulation System," in *Proceedings of the European Simulation Multiconference*, June 2001.
- [25] G. Van der Auwera, P. T. David, and M. Reisslein, "Traffic Characteristics of H.264/AVC Variable Bit Rate Video," Tech. Rep., Arizona State University, March 2007.