NOC/OC&I 2009 Proceedings

Universidad de Valladolid

Valladolid (Spain), 10 - 12 June 2009

14th European Conference on Networks and Optical Communications

4th Conference on Optical Cabling and Infrastructure

Edited by:
I. de Miguel, R.M. Lorenzo, D.W. Faulkner and R. Neat

Organized by:

UVa
Universidad de Valladolid

NOC/OC&I 2009 Proceedings - 14th European Conference on Networks and Optical Communications and 4th Conference on Optical Cabling and Infrastructure
Valladolid, Spain, 10-12 June 2009

Published by:

Universidad de Valladolid
Plaza del Colegio de Santa Cruz 8
47002 Valladolid
Spain

ISBN: 978-84-692-2943-9

The effect of multiplexing video flows in the performance of OBS edge node buffers

Felix Espina, Mikel Izal, Daniel Morató and Eduardo Magaña
Public University of Navarre, Campus de Arrosadía, E-31006 Pamplona, Spain
e-mail: {felix.espina, mikel.izal, daniel.morato, eduardo.magana}@unavarra.es *

Traffic injected into an Optical Burst Switched (OBS) network is a burst arrival process whose characteristics depend not only on input traffic parameters but also on design parameters of the OBS network such as the burst formation timeout value. The properties of this traffic will dictate how it is affected by transmision over the OBS core. If OBS is used as a transport network inside a triple-play provider infraestructure the user-perceived quality of the video streams that are delivered at the edge of the network depends not only on video stream parameters but also on some design parameters of the architecture. The main effect that degrades this quality is data loss because jitter can be easily compensated using buffers at the receiver. The impact of some parameters of the architecture like the burstifier timer and the buffer at the ingress node have already been studied, but the convenience of grouping several streams in the same burstifier remains an open question. An OBS network for the transport of video flows will have a number of ingress video streams that will have to be delivered to different egress nodes. Usually, several of the video streams will have to be delivered to the same edge node. This poses the question of whether bursts should be built separately for each stream, gathered all in one burstifier per destination or something in-between. This paper addresses the study of the optimal number of streams per burstifier in a video over OBS scenario. Results show a large degree of multiplexing per burstifier allows reducing losses at the edge node. However If a high multiplexing degree is not achievable, the opposite extreme with buffering and no multiplexing is advised to minimize losses. In order to fill the minimum burst length imposed by the OBS technology it is better to multiplex more streams than to increase the timer value to get the same average burst length.

1. Introduction

Optical Burst Switched (OBS) networks provide an efficient backbone to high bandwidth demands. Video-on-demand providers are clear candidates to use an OBS infrastructure to transport data flows. The OBS architecture allows for building a backbone of passive optical switches governed by electronic control units that switch data on the optical plane in large packets called bursts. The handling of large bursts instead of IP packets relaxes the requirements for control electronic, thus permitting to achieve higher data rates. The counterpart is that the gathering of bursts from incoming traffic introduces one way latency and makes losses to occur also in bursts.

---

* This work was funded by Spanish MEC (project STRRONG TEC2007-62192/TCM). The authors want to thank Spanish thematic network IPoTN (TEC2008-02552-E/TEC)

A typical OBS transport scenario consists of a backbone network and the edge nodes. The backbone network is composed of interconnected optical switches that can schedule bursts in the optical domain. The edge nodes perform packet gathering for ingress traffic and disassemble incoming bursts into IP packets to be injected into the legacy access networks. A key component of ingress nodes is the burstifier which accumulates incoming IP packets in a queue to form a burst. Several types of burstifiers have been proposed depending on the conditions that trigger the sending of bursts: size-based, timer-based or a mixture of both [1]. In the case of a video provider using an OBS infrastructure, video servers located at an IP access network will send video streams to clients located in other access networks (see figure 1). The video flows will be burstified and sent as bursts through the OBS backbone. In this scenario, one way latency of video transmission can be limited by using timer-based burstifiers with properly selected timer values. But selecting the timer value or other parameters at the edge node will have an effect on the OBS input traffic process and thus on the performance of the OBS transport infrastructure through burst losses. The knowledge of this dependency will allow the dimensioning of OBS networks for video transport purposes.

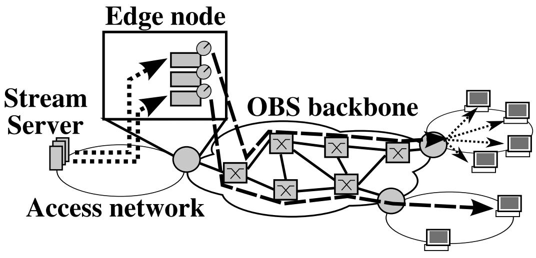


Figure 1: Scenario

Previous works on generic data traffic over an OBS network have grouped packets into different burstifiers according mainly to the destination address. The ingress node to the OBS backbone classifies input traffic, redirecting the packets to the burstifiers that prepare bursts to be sent to the right egress node. In [2], video transmission is simulated over an OBS network, describing the effect of timer- and size-triggered burstifiers when all video streams are aggregated into a common burstifier. In [3], the authors study the effect that burst losses in the OBS network have on the decoding of video frames, but for the case of a different burstifier used for each video stream.

Regarding the performance of the edge nodes, queueing models are presented in [4, 5], offering results on delay and throughput but not on losses. In [6] the burst blocking probability of an edge node is presented but using a scenario of OBS with acknowledgments (OBS/A) which assumes a more circuit-like approach, thus its burstifier parameters cannot be applied to an unacknowledged scenario. Finally, on [7] losses at the edge node are modelled using an Erlang-B formula but only the one wavelength output port scenario is studied.

This paper addresses the dimensioning of ingress nodes for a streaming service over an OBS backbone. It takes into account the burstifier timer value, the number of wavelengths at the access link, the buffer space used and also the applied multiplexing scheme. The stream aggregation and spreading over a larger or smaller number of burstifiers emerges in this paper as a new important dimensioning parameter.

The rest of the paper is organized as follows. Section 2. describes the specific scenario studied and the simulation methodology. Section 3. presents and discusses results on a bufferless edge node case. Section 4. discusses the results obtained when buffering can be used to accommodate bursts in the edge node. Finally, section 5. concludes the paper

2. Scenario and methodology

Figure 1 presents the scenario studied in this paper. It shows the transport and access networks used by a video on demand service provider. The video streaming servers are located in a non-OBS legacy network, connected to the OBS backbone through an edge node. The edge node gathers bursts with incoming video traffic and sends them to the egress nodes where the clients are located.

The ingress edge node classifies incoming video traffic into several different burstifiers. To study the effect of the level of stream multiplexing on edge node performance, a homogeneous traffic is assumed, composed of N total video streams entering the network (see figure 2). Each of the video streams has an average bit rate of r. These streams are evenly distributed into k burstifiers, in order to have every burstifier gathering packets from n video traffic processes (thus $N = kn$). The link from the ingress node to the OBS core network contains c wavelengths with link capacity v. Therefore input load of N video flows generates an utilization factor of $\rho = \frac{Nr}{cv}$.

The ingress node burstifiers are timer triggered burstifiers with timer value set to T_{out}. Once a burst if formed by the burstifier, it is injected into an available output wavelength. If there is not an available wavelength two approaches are considered. The simplest case is the one where the burst is dropped due to the bufferless nature of the OBS ingress node. But given that edge nodes are processing bursts in the electronic domain, where memory is feasible, a buffer to store formed bursts is a reasonable assumption. Thus, the case of edge nodes with a buffer of size b is also studied.

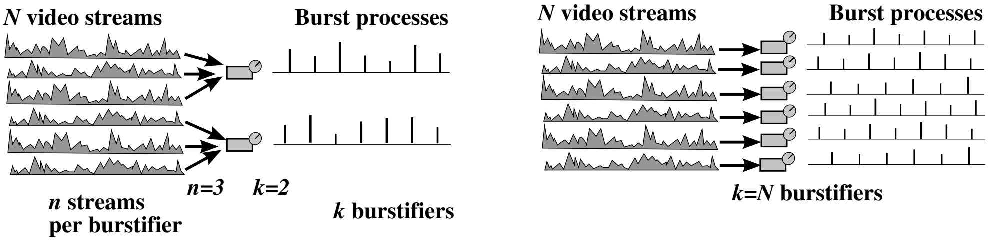


Figure 2: Multiplexing streams in burstifiers

Simulations are performed to measure the burst loss probability of the system described. The input traffic is generated from traces of MPEG4 video made available from [8] to use in research. The traces provide the frame size and time sequence of compressed movies with different bit rates and coding characteristics. They are used to generate the burst process coming out of burstifiers receiving n simultaneous video streams. The output processes from k burstifiers are fed together to a custom-made event-driven simulator that computes burst loss in the output link.

An interactive scenario is assumed, where users can select a movie and start a new video flow at any time. This is modelled by multiplexing a traffic trace selecting different random starting times. The results are homogeneous input traffic processes with different rate based on the number of flows multiplexed. Different video traffic traces have been used, obtaining the same results. The results presented and discussed at the following sections were generated from the "Tokyo Olympics" video trace located at `http://trace.eas.asu.edu/mpeg4`. The trace is extracted from a 74 minutes long movie sequence encoded using the GoP (Group of Pictures) structure G16B7 at 30 frames per second. It contains a base video stream of 6.6Mbps with an average frame size of 27.5kB.

3. Bufferless edge node

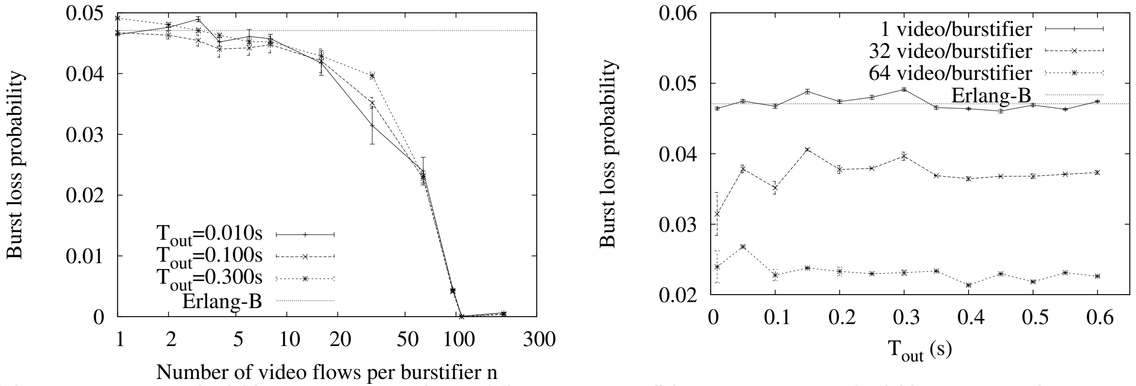
In this section the performance results for the bufferless case are examined. The ingress node output link utilization factor is set to $\rho = 0.713$. Figure 3(a) shows the Burst Loss Probability versus the number of video flows per burstifier n. It can be noted that for low values of n the loss probability is equal to the value obtained using the Erlang-B formula (plotted as a reference). As the multiplexing level n grows, the loss probability falls to 0.This can be explained by examining the burst process offered to the OBS access link. In the case of one or just a few video flows per burstifier, the burst traffic process is the result of multiplexing around k burst arrival processes coming from k independent burstifiers. Each of the burstifiers generates a close to periodic arrival process due to the periodic nature of timer-triggered burstifier traffic. The multiplexing of this large number of independent periodic arrival processes is close to a Poisson process, at least in an interval of arrivals as large as the number k of multiplexed flows. The variable bit rate of the video streams translates into the distribution of burst sizes but not into the arrival times. In an interval of k arrivals, their sizes are independent as they are from independent flows, hence the system behaves close to a M/G/c/c system and the burst loss probability can be approximated by the Erlang-B formula $B(c, c\rho)$.

Being t_{if} the inter frame time of each video stream, the number of frames per video flow in a burst is $n_f = \lfloor \frac{T_{out}}{t_{if}} \rfloor$. Being v_i the sum of the n_f frame sizes from flow i in a burst, the whole burst size is $s_b = \sum_{i=1}^{n} v_i$, where the values of v_i in a burst are independent. As the multiplexing level per burstifier n increases, the burst arrival process gets reduced to a periodic process with a large number of video flows being gathered in a single burstifier. The Central Limit Theorem states that $s_b \sim N(E[v_i], \frac{\sqrt{Var[v_i]}}{\sqrt{n}})$. As n increases the traffic process becomes a set of k periodic burst arrival process with gaussian burst sizes and variance tending to 0. Therefore as long as $\rho < 1$, the burst loss probability tends to 0.

Figure 3 shows the loss probability versus the burstifier timer value. The 95% confidence intervals are added to the figure. It can be observed that T_{out} has no effect in a bufferless scenario.

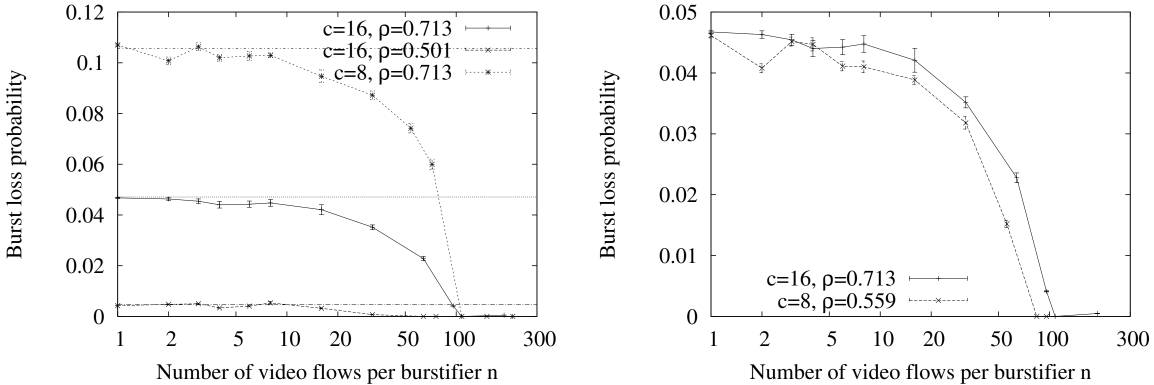
Figure 4(a) shows the burst loss probability for scenarios with different load and number of wavelengths in the output port. The Erlang-B value keeps as a consistent approximation for low aggregation levels, independent of the values of ρ and c. Figure 4(b) shows two scenarios of c and ρ values that result in the same Erlang-B loss value. Both plots start at the same point but the decay rate is different. The multiplexing level where the loss probability becomes negligible corresponds to a number of burstifiers below the number of wavelengths, or $n > N/c$.

Thus, in a bufferless OBS edge node, aggregating traffic from N video streams into



(a) Burst loss probability versus number of flows (b) Burst loss probability versus timer

Figure 3: Burst loss probability in a $\rho = 0.7$ and 16 wavelengths scenario



(a) Effect of ρ and c (b) For ρ_1,c_1,ρ_2,c_2 such as $B(c_1, c_1\rho_1) = B(c_2, c_2\rho_2)$

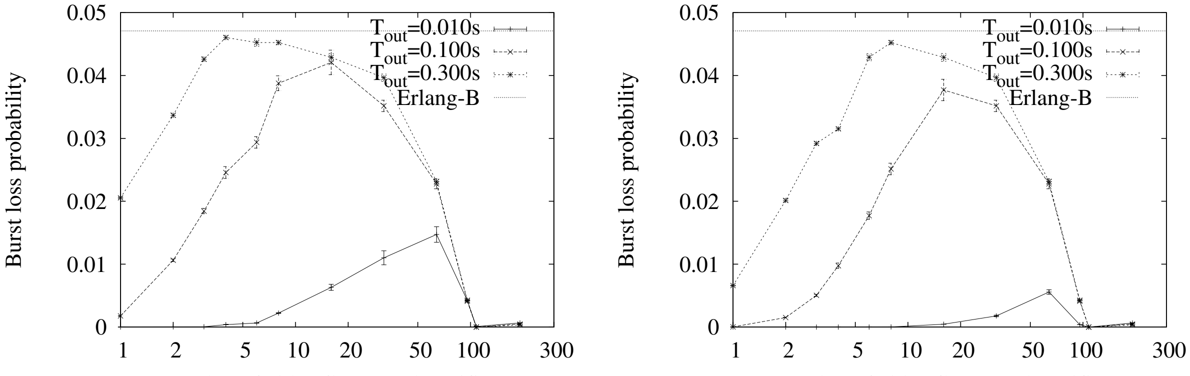
Figure 4: Burst loss probability vs number of flows per burstifier

$k = \frac{N}{n}$ burstifiers the burst loss probability at the edge node can be reduced. If this number of burstifiers is below c losses will be negligible. However, the number of different burstifiers cannot be reduced below the number of destination egress nodes. The best burst loss probability is therefore obtained using the largest available number of video flows per burstifier, or equivalently the lesser number of distinct burstifiers.

From a network architecture perspective, the optimal situation for the edge node is one in which the video flows are destined to a reduced number of egress nodes. Service clients could be relocated in order to reduce the number of destination egress nodes. However, a more suitable architectural solution would be to distribute the video servers into different access networks, using different ingress edge nodes. The clients for each video server would be selected such that they could be clustered into as few destination egress nodes as possible. This represents using topology in order to improve performance. Of course, buffering at the edge node is also another option, as presented on the following section.

4. The combined effect of buffering and the multiplexing level

Buffer memory in an OBS core network is only feasible nowadays using Fiber Delay Lines (FDLs). However, at the edge nodes, before electro-optical conversion, data packets and bursts can be stored using electronic memory. In this section the performance of an edge node with an electronic buffer is studied.
Figure 5 shows the burst loss probability versus the number of video flows per burstifier n in a finite buffer ingress node with buffer size $b = 5Mb$ (5(a)) and $b = 10Mb$ (5(b)). For large values of n, when the number of burstifiers decreases the behavior is the same as in the bufferless case. The loss probability decays till the point where it is negligible around n such that $\frac{N}{n} = c$. On the other side, the burst loss probability decreases also when n decreases. This can be explained based on the average burst size. In the case of n video streams per burstifier the average burst size generated will be $s_b = nT_{out}r$ being r the average rate of an individual video stream. The average burst size increases with n making the same buffer size able to accommodate less bursts, decreasing the effective buffer size. As the burst size increases it reaches a point where the buffer size cannot accommodate a single burst. At this point the behaviour of the system is equal to a bufferless one.



(a) for $b = 5Mb$ (b) for $b = 10Mb$

Figure 5: Brust loss probability vs flows per burstifier in a finite buffer edge node

Therefore, in an edge node with buffer the worst case performance is given by Erlang-B formula. This result may be improved by reducing the timer value. The best results in burst loss probability are reached for very large and very small number of flows per burstifier. The worst case is obtained when the multiplexing level per burstifier n_W verifies $b = n_W T_{out} r$. The number of burstifiers for this worst case is $k_W = \frac{NT_{out}r}{b}$.
The best performance with a multiplexing level n lower than n_W would be achieved using $n = 1$ (one burstifier per video stream). It results in smaller bursts and therefore a better buffer usage. The best performance with n larger than n_W is obtained with every possible flow in the same burstifier ($k = 1$). However, it is enough that $k\ lec$ to achieve near zero losses. However, the number of burstifiers cannot always be reduced to this degree as it depends on the number of destinations. In practice, the range of values for n to choose is in $[1, \frac{N}{d}]$ where d is the number of different destinations. The decision between a large number of burstifiers or a single one depends on whether the loss probability for $n = \frac{N}{d}$ (see Figure 5) offers a lower loss probability than for $n = 1$.

Therefore, the use of buffering at the edge nodes not only reduces the burst loss probability but also allows the reduction of the multiplexing level required. This reduction implies a larger number of burstifiers, each one for a potential different destination, allowing distribution topologies with a larger number of egress nodes.
It's interesting to note that when buffering is used, the burstifier timeout value becomes an important parameter because it has a direct effect on the average burst size. The average burst size decides the average number of bursts that can be stored in the buffer and so its effectiveness.
If a given scenario imposes a maximum number of flows per burstifier lesser than n_W, figure 5 shows that the best option is to use a different burstifier for each video stream and use a minimum timer. Although this operating point is optimum with the present model, it has some practical limitations. This configuration scenario will result in very small burst, however, the very fundamental characteristic of OBS is the creation of large switched entities (bursts), hence, the technology will force a minimum burst size. In order to increase the burst size from this optimum and unfeasible working point two options are available. As the average burst size is $E[s_b] = nT_{out}r$, increasing the multiplexing level n aggregates more flows in the same buffering time, incresing the burst size. The result will be a loss probability increase, followin the curve in Figure 5. The second approach keeps one video stream per burstifier but increases the timer value, with the result of more frames from the same video flow increasing the burst size. As Figure 5 also shows, changing to a different plot with larger timeout also increases the loss probability.

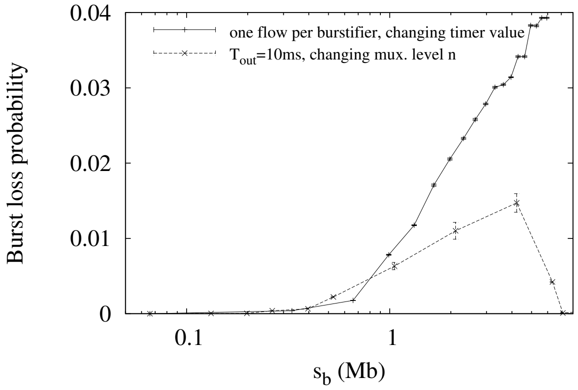


Figure 6: Burst loss probability versus timer value (5Mb buffer)

One of the two options could offer the possibility to reach a minimum average burst size keeping the loss probability as low as possible. Figure 6 shows burst loss probability obtained (y axis) when an average burst size (x axis) is reached increasing the multiplexing level or the timer value. The scenario with $5Mb$ buffering is simulated. A curve shows the effect on the loss probability of keeping $n = 1$ (one flow per burstifier) and increasing the timer value T_{out}. The second curve shows the effect of keeping $T_{out} = 10ms$ and increasing n. It can be observed that the loss probability increases slower by adding more video flows inside the burstifiers than by adding more time to gather more frames from the same flow. The effect can be explained noting that in the first case (increasing n) the burst is formed getting more frames from different streams, while in the second case (increasing T_{out}) the burst grows by adding more frames from the same stream. In the former the burst size gets a reduction of the variance as a result of the multiplexing of independent frame sizes. In the latter the long range dependence of the video traffic results in a much slower variance reduction. Hence, the queueing results will improve with the reduced variance case.

5. Conclusions

This paper has shown the effect of the video flow multiplexing on burst loss probability at an ingress node in a scenario of video transport over an OBS network. Burst losses introduced by contention at a bufferless edge node can be reduced by multiplexing video flows on a small number of burstifiers. However the number of burstifiers must be at least equal to the number of different egress nodes. If the number of destinations cannot be reduced, losses at the edge node can be approximated by the Erlang-B formula.

Burst losses can be reduced below Erlang-B value by using buffered ingress nodes. For an edge node with buffer, the burst loss probability presents a local maximum with the level of multiplexing, making the behavior more complex. Depending on the maximum allowable number of burstifiers (given by the number of different destinations) it may be better to use just one single burstifier for every flow or group the largest possible number flows per burstifier. The number of flows per burstifier becomes a critical parameter to plan the infrastructure for a video on demand distribution network using an OBS core.

In the case of a large number of destinations it is convenient to use a single burstifier per flow and use small timer values. In order to fill the minimum burst length imposed by the OBS technology it is better to multiplex more streams than to increase the timer value to get the same average burst length.

References

[1] J. Choi, J. Choi, and M. Kang, “Dimensioning burst assembly process in optical burst switching networks,” *IEICE Transactions on Communications*, vol. E88-B(10), pp. 3855–3863, (2005).

[2] T. Vargas, J. Guerri, and S. Sales, “Effect and optimization of burst assembly algorithms for video traffic transmissions over OBS networks,” in *Proceedings of BROADNETS08*, (London, 2008).

[3] F. Espina, D. Morato, M. Izal, and E. Magaña, “The effect of burst formation timers on video streaming over optical burst switched networks,” in *Proceedings of BROADNETS08*, (London, 2008).

[4] L. Xu, H. G. Perros, and G. N. Rouskas, “Performance analysis of an edge optical burst switching node with a large number of wavelengths,” in *In 18th International Teletraffic Congress (ITC)*, 2003.

[5] L. Xu, H. G. Perros, and G. N. Rouskas, “A queueing network model of an edge optical burst switching node,” in *IEEE INFOCOM 2003*, April 2003.

[6] A. Zalesky, E. Wong, M. Zukerman, H. L. Vu, and R. Tucker, “Performance analysis of an OBS edge router,” *Photonics Technology Letters, IEEE*, vol. 16, pp. 695–697, Feb 2004.

[7] H. Li and I. L.-J. Thng, “Edge node buffer usage in optical burst switching networks,” *Photonic Networks Communications*, no. 13, pp. 31–51, 2007.

[8] G. V. der Auwera, P. T. David, and M. Reisslein, “Traffic Characteristis of H.264/AVC Variable Bit Rate Video,” tech. rep., (Arizona State University, March 2007).