Discrete Event Simulation of a large OBS Network

Stein Gjessing
Simula Research Laboratory &
Dept. of Informatics, University of Oslo,
P-O. Box 1080, N-0316 Oslo, Norway
steing@simula.no

Arne Maus
Dept. of Informatics
University of Oslo,
P-O. Box 1080, N-0316 Oslo, Norway
arnem@ifi.uio.no

Abstract – *Optical Burst Switching (OBS) is a much researched paradigm for the next generation optical Internet. We have made a detailed discrete event simulation model of OBS networks. Among other things our model includes self similar traffic sources, burst assembly with fixed and variable length bursts, burst scheduling, wavelength conversion and fiber delay lines. In this paper we simulate a large optical burst-switched backbone network using the COST 239 network topology that connects 11 European cities. The performance of this realistic network is investigated, mainly by varying the load and the number of channels (lambdas). We propose and evaluate a new method for the utilization of otherwise unused network capacity by very low priority traffic. Other interesting results include how total burst loss increase when high priority bursts are used.*

Keywords: *Optical burst switching, discrete event simulations, traffic modeling, burst loss, burst scheduling.*

1 Introduction and motivation

Optical Burst Switching (OBS) is a much research paradigm for the next generation optical Internet [1]. OBS is performed on Wavelength Division Multiplexed (WDM) fibers, is more fine-grained than optical line switching, and more coarse-grained than Optical Packet Switching (OPS). The main motivation behind optical switching is to transport the data with minimal delay by keeping it in the optical domain. A control packet precedes the data burst in the network and reserves resources on the links and in the switches for the data burst. Only the control packet is converted from optical to electrical (and back) in each switch.

Analyses of OBS network have been performed mainly by analytical models, or relatively simple simulation models often looking at the traffic over a singe link with a few traffic sources. However, also some more detailed simulation models have been developed [2]. Schlosser at al. have simulated a two node OBS network with Poisson and Pareto distributed traffic [3], and Ahmad and Malik have built an OBS simulator using the Ptolemy framework [4]. In [5] a two node OBS network is simulated in order to study blocking probabilities, and the NSF network in the USA has been modeled by several researchers [6,16]. Most of the analytical work reported uses Poison arrival rates, e.g. [1,5,8,16]. More complex traffic has been used in some papers, e.g. by analyzing OBS edge routers using traffic generated by Markovian processes [7]. Blocking probabilities and QoS have been researched by many; notable research are reported in [5,9,10,15].

The main contribution of this paper is a detailed realistic large scale discrete event simulation and traffic model of a core OBS network and some initial results obtained from running this simulator on a large network. As will be detailed in the next section, we model a large number of traffic sources with variable (IP) packet sizes, burst assembly (fixed or variable sized), burst scheduling with the possibility of wavelength conversion and fiber delay lines, QoS using longer control packet lead time (CPT), and deflection routing. Few such detailed and realistic discrete event simulation models have been published before. The most detailed model we know of uses much shorter bursts [2]. By running our simulator we find performance properties that are not revealed by analytic models or simpler simulations. Initial findings reported in this paper includes the effect on regular bursts from high priority (QoS) bursts, optimal bursts sizes, and a new method to send very low priority bursts that utilize otherwise unused bandwidth and hence do not affect regular bursts at all.

The traffic load onto an OBS core network comes from IP-subnets and Ethernets. It is well known that Ethernet and IP traffic exhibit self similar properties. The statistical properties of the bursts in an OBS core network however are not well known. A contribution of our work is to implicitly find these parameters by making a detailed simulation model of the burst assembly process, using synthetically generated Ethernet and Internet self similar traffic.

This paper is organized as follows. In the next section we present and discuss our simulation model. In section 3 we present a case study involving a large European network, and section 4 presents and discusses results obtained from running traffic in this network. Finally in section 5 we conclude and point at possible further work.

2 The OBS simulation model

We have used the J-sim framework [11], and implemented a full OBS discrete event simulation model on top. We build on the Component model as well as the Packet and Link concepts of J-sim. The data sources and burst assembly modules, as well as the OBS-switches and schedulers are built from scratch. Network data for a specific scenario, including topology, link propagation times and routing (forwarding) tables are read from a file at system start up time.

2.1 Traffic generation

In order to model the aggregate of IP traffic arriving at an OBS core network ingress node, a large number of Pareto sources are active in each ingress node [12]. A large number of sources are needed in order to realistically model a self similar source. In the experiments reported in this paper we use between 50 and 350 sources per ingress node, each source draw on and off periods according to a Pareto distribution with Hurst parameter 0.9. The traffic generated by a Pareto source is controlled by a parameter called the load factor, also called the ingress load. The load is the relation between the mean on period and the mean off period. E.g. if the load is 0.2, the source is 5 times longer in an off period than in an on period. Whenever the source is in the on mode, IP packets are generated with constant intervals. The size of the IP packets are varied.

The destination address of each IP packet is set depending on the traffic matrix. Then, whenever a Pareto source starts a new on-period, a destination address is chosen according to the traffic matrix. This address is used for all packets generated by this source for the duration of this on period.

The ingress nodes may also generate traffic with simpler statistical properties, e.g. Poisson arrival rate or CBR (really constant packet rate) traffic. The destination address (egress node identification) may be chosen on a packet by packet basis or on a stream basis.

2.2 Burst assembly

Each ingress node has a set of buffers for each possible destination (egress node). A configurable parameter decides whether the bursts are of fixed or variable size. If the load is light, and the maximum burst size is large, it will take a long time to fill up a buffer. Hence a timer, called the burst time out timer, is set whenever the first IP packet is put into the buffer. When this timer expires, the burst is sent. When variable sized bursts are used, the size of a burst can not change after the control packet has been sent. When using fixed sized bursts, the burst buffer can be filled with new packets until the burst is sent. The size of a burst includes extra bytes (padding) to cater for the time it takes to configure the wavelength converters (see below) and also keep the bursts separate.

2.3 The OBS switches

A switch contains one network card for each (full duplex) link. Routing tables are initiated at system startup time. Any switch in the network may also be an ingress and an egress node. Whenever a burst is ready to be sent from an ingress node, a control packet is sent ahead. All control packets are sent on a lambda used for control packets only. The control packet is converted to the electrical domain when it reaches a new switch. Here it calls the scheduler to reserve resources for the forwarding of the burst from the input to the correct output link. There are many variants of resource reservation. E.g. should the resources be reserved immediately, and how should they be released? In this paper we use the Just-Enough-Time (JET) principle, i.e. the control packet preceding the data burst reserves the resources just for the time they are needed [13]. The optical to electrical conversion of the control packet and the scheduling, takes more time in the switch than the all optical switching of the burst. This extra time, called the control packet forwarding time, is configurable.

In general a burst can be scheduled in every void (a vacant time slot on a channel) that is long enough and available at the correct time. In order to increase the probability that a usable void is found on the output fiber, we have equipped the switches with wavelength converters. In this way the scheduling algorithm may set up a wavelength conversion for the data burst, so that the outgoing wavelength contains a usable void. If no free wavelength can be found, we have implemented fiber delay lines (FDL), which act as small buffers, and delay the data in the switch until a usable void is found. Each switch may be equipped with a number of such delay loops that can be serialized, and the scheduler tries to reserve as few loops as possible. If the output scheduling still has not succeeded in finding a usable void, the burst may be deflected by being transmitted on another link. The burst may come back (and then hopefully the original output fiber has a void available) or the burst may take another rout to the destination. We have implemented a simple deflection algorithm that chooses an output link at random, and prevent indefinite looping by the fact that when the data burst is overtaking the control packet (remember that the data burst is traveling all optical, and hence gains on the control packet each time it is switched) they are both discarded. Also in the case that the control packet is not able to reserve the needed resources for the data burst, the data burst (and the control packet) is discarded by the switch.

3 The COST 239 Experiment

In this paper we present an initial performance evaluation of a large OBS core network. We use the COST 239 network topology (figure 1) consisting of 11 nodes (European cities) connected by 26 (bidirectional) links [14]. The number of lambdas is varied over the experiments, from 5 channels to 35 channels. Each lambda transmits at one Gbit/s, hence the capacity of the links vary form 5 to 35 Gbit/s. We simulate full lambda conversion between incoming and outgoing lambdas, but only when explicitly stated we use Fiber Delay Lines. The control packet forwarding time is set to 10µs. The control packet lead time (CPT) used by the ingress switch is 200µs if nothing else is stated. Since the control packet forwarding time is 10µs, a burst can be forwarded through 20 switches before it must be discarded because the CPT becomes zero and the burst overtakes the control packet.

The routing is done according to a shortest path algorithm, taking into account the propagation delay on the links according to figure 1. In this paper we do not use deflection, hence a burst is lost if the control packet is not able to find a vacant time slot (a void) on one of the lambdas (channels) on the output link.

All 11 cities are ingress nodes (generating traffic), egress nodes and internal switching nodes in the network. The traffic matrix is symmetric all-to-all. Whenever a Pareto source starts a new on-period, a destination address is chosen at random between all the other nodes in the network. The time between packets generated by the Pareto source when in its on period is 10.7µs. The sizes of the generated IP packets vary from 80 bytes to 1500 bytes with a mean size of 500 bytes.

In order to easily compare the network performance when different number of channels are used (all links have the same capacity) the traffic generated is also changed accordingly. In all experiments the number of Pareto sources per ingress router is 10 times the number of channels. In the first experiments described in the sequel (called the base case), the number of channels per link is 10 (10Gbit/ links). Hence, all ingress nodes contain 100 Pareto sources. The load is varied from around 0.02 to 0.5. Using 10 channels, an ingress load of 0.1 means that each ingress node generates about 4 Gbit/sec in total, or 400Mbit/s destined for each of the other 10 egress nodes. In this article we mostly use variable sized bursts with a maximum burst size of 50 000 bytes. If nothing else is stated, the burst time out value is 2ms.

In some of the experiments reported, we use a CBR stream of small constant sized bursts. Node 1 (London) is the ingress node and node 9 (Vienna) the egress node of this stream. The burst size is 2000 bytes and they are sent with an interval of 30µs. It takes 16µs to transmit a 2000 byte burst. Hence this stream occupies about half of one channel, or 5% of the total capacity of the 10 channels used in this experiment. FDLs are used in one of the experiments with the CBR bursts, and then only to delay CBR bursts.

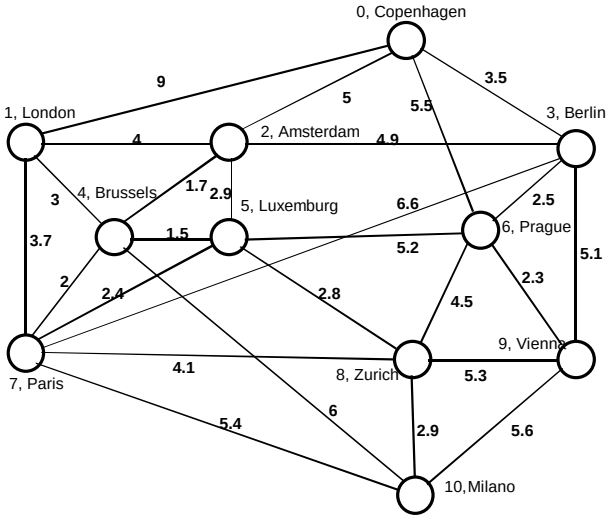

Figure 1. The Cost 239 network between 11 European cities. Optical propagation delay in milliseconds on each link.

4 Results from the COST Experiment

In this section we look at the most interesting results from our simulations using the COST 239 network topology.

4.1 Varying the load

First, as our base case, all lines are fixed at 10Gbit/s, ie. there are 10 channels per link (10 lambdas). We start by a very light load, which is gradually increased. Figure 2 shows the relative packet loss rate as a function of the offered ingress load. As long as the traffic generated is below a load of 0.07 (i.e. each of the 11 ingress nodes generates 2.8Gbit/s), there is no packet loss anywhere in the network. When the load has increased to 0.15, the drop rate is approximately 1%, and at load 0.2 (8Gbit/s generated by each node) the drop rate is around 2%. For higher loads the drop rate increases more sharply and seems to increase linearly. These results conform to previous analytical results [5].

Looking closer at the individual links, we find a very early congestion on the link from node 5 (Luxemburg) to node 6 (Prague). In figure 3 we see how much of the traffic offered onto this link is able to successfully be transmitted,

e.g. with an ingress load of 0.2 there is a burst drop rate of about 10 %. Also here the burst drop rate seems to increase almost linearly from load 0.1 to load 0.4.

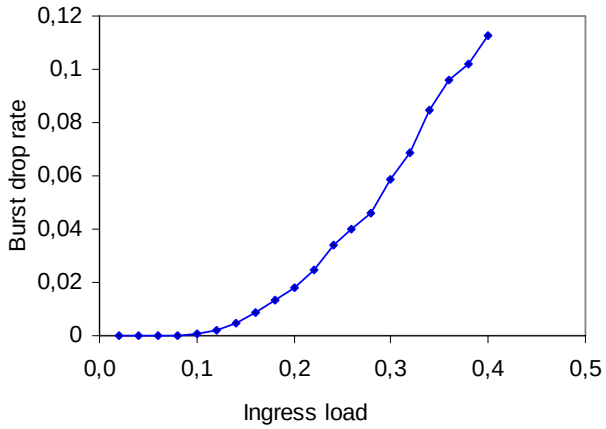

Figure 2. Total network burst drop rate with varying ingress load. 10 channels per link

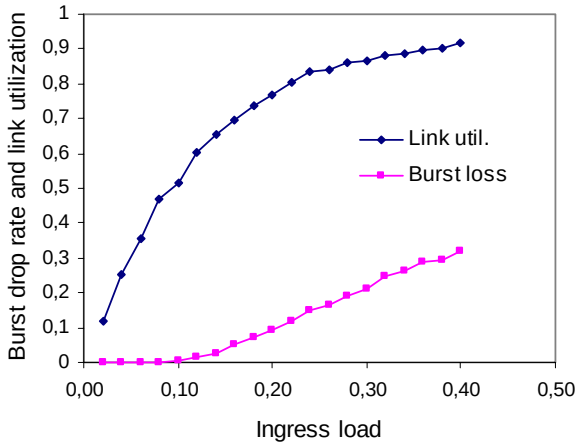

Figure 3. Link utilization and burst drop rate on link 5-6 as a function of the ingress load.

4.2 Transferring a stream of small bursts

The term "horizon scheduling" is used when indefinitely long voids into the future are allocated. If the start of the void is before the start of the burst, the burst may be scheduled in this void, creating a similar indefinitely long void after the burst as well as a small void in front of the burst. Such small voids in front of bursts may only be utilized by small bursts with short CPTs. This will be the topic of section 4.2.2.

4.2.1 High priority bursts

By increasing the CPT of high priority (QoS) bursts by the transmission time of the regular bursts, a 100% guarantee of transmission success for these bursts are ensured (assuming there are not too many such QoS bursts). The down side is, however, that regular bursts may be dropped. The purpose of our next experiment is to investigate this dependency by observing the CBR stream of small bursts from London to Vienna. We let these QoS bursts have a CPT of 600µs as opposed to 200µs for the regular bursts, whose mean transmission time is 400µs. Looking back at the base case (figure 3), we remember that when the load is light (0.1), only 50% of the link is utilized. The QoS plot in figure 4 illustrates that the extra stream of 5% of the full link capacity increases the relative utilization to 1.1. At ingress loads lighter than 0.24 the extra stream adds to the utilization of the link. But when the load increases above 0.24 and the link is already 80% utilized, the extra high priority stream makes the total performance of the link degrade. The QoS plot in figure 5 confirms that no packets in this stream are lost.

4.2.2 Very low priority bursts

By realizing that horizon scheduling is almost always used, we are looking for a way to use 'spare' capacity between the control packet and its burst, for another stream of data, hoping to transfer more data without disturbing the regular burst. This can be done by transmitting short bursts with short CPT, such that the combination of the two is transmitted faster than the CPT of the regular bursts. We use the same stream of small bursts from London to Vienna as above, but this time we use a CPT of 50 µs. This is almost the smallest CPT possible for a burst that will travel through 3 switches. In order to perform an initial investigation of the usefulness of Fiber Delay Lines (FDLs), we also include a scenario where the low priority bursts my use FDLs in order to have a higher transmission success rate (still without affecting the regular bursts). In order to compare to regular bursts, we also send the small burst stream with the same CPT (and hence the same "QoS class") as the regular bursts. Hence, three new scheduling schemes are tried (in addition to the above QoS scheme):

- Low: CPT = 50µs
- Low-FDL: same as Low + the use of FDLs
- Same: CPT = 200µs (same as the regular bursts)

Of special interest are the Low and Low-FDL cases, because no regular bursts are blocked by them. Figure 4 shows how the traffic on the congested link is increased. However, figure 5 shows that our new low priority bursts have a very high probability of getting dropped. Hence, it is clear that such low priority bursts may only be used for traffic where a very high drop rate is acceptable.

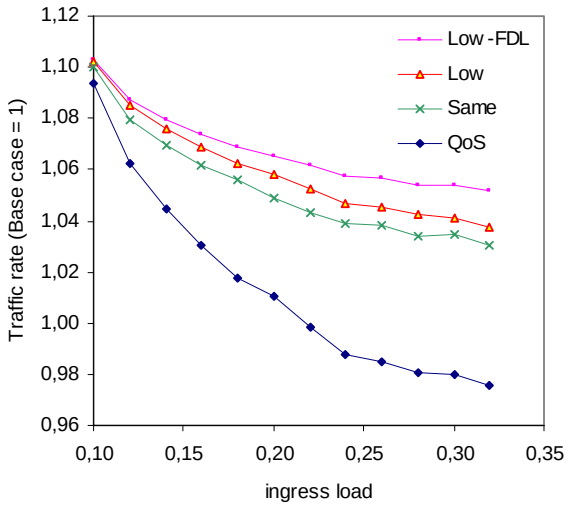

Figure 4. Total traffic on link 5-6 when sending additional small bursts, as a function of the ingress load

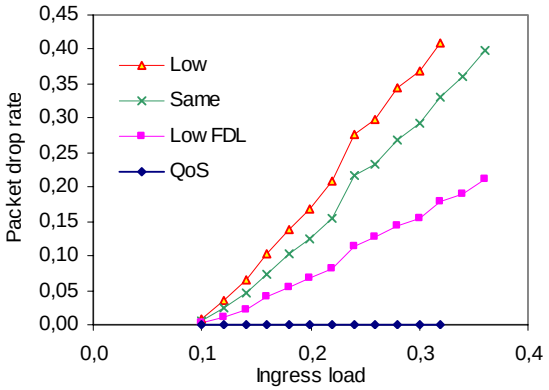

Figure 5. The drop rate on link 5-6 for the small bursts

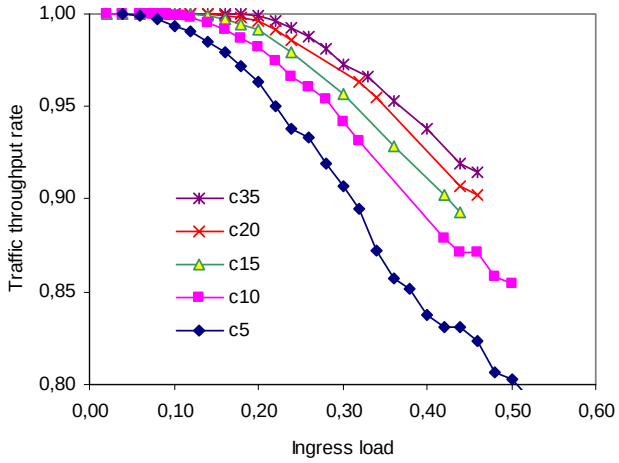

Figure 6. Network throughput as a function of channels and ingress load

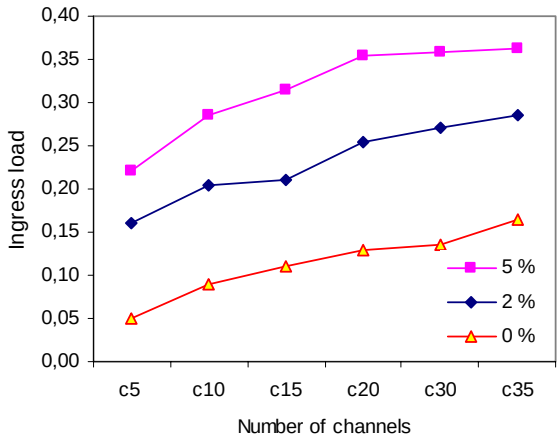

Figure 7. Drop rates as a function of the number of channels

4.3 Varying the number of lambdas

In this experiment we investigate some effects of increasing the number of lambdas (channels). Remember that the number of traffic sources in each ingress node is proportional to the number of channels. When the number of channels is increased, figure 6 shows that the total throughput in the network is kept higher than for lower rates. This is caused by the fact that there is a higher probability of finding a usable void when the number of channels increases. Figure 7 shows e.g. that the drop rate is 0% all the way up to an ingress load of 0.13 for 35 channels. This means approximately a doubling of the link utilization (with no burst loss), compared to 10 channels. The results presented here are according to the analytical results given by other research [1,8].

5 Conclusions and Further Work

In this paper we have presented and discussed our detailed OBS simulation model and some experiments using the European COST 239 network topology. We have shown that it is possible, not knowing the arrival pattern of sizes of bursts, to simulate self similar Ethernet and IP traffic (using a large number of Pareto sources) and thereby creating a realistic model of the burst assembly and burst time out processes.

Most previous performance results of OBS networks have been found using analytical methods. Using such methods it is very hard to reason about large complex systems with self similar traffic arrival models. Our results, using a new detailed descrete event simulation model, confirm what other researchers have found using analytical and other simulation models. E.g. the blocking probabilities found completely confirm previous results, and also our initial results concerning how regular traffic is affected by high priority traffic supports earlier results. Finally we have demonstrated and evaluated how very low priority traffic may be sent without disturbing regular burst traffic at all.

In future work we will use more topologies and more experiments, in order to obtain results that are more statistical significant. In addition we will undertake a thorough comparison to previous work that have researched the same OBS properties by analytical and other methods.

Acknowledgements

The authors would like to thank Amund Kvalbein for making the Pareto source module. Our OBS simulation model is based upon another network simulator developed by him and one of the authors. Thanks also to Audun Fosselie Hansen who took part in the initial design of the OBS simulation model.

References

[1] Myungsik Yoo, Chunming Qiao, “A novel switching paradigm for buffer-less WDM networks”, Proceedings Optical Fiber Communication Conference, San Diego, Ca, USA, pp 177 - 179 Feb. 1999.

[2] Fei Xue, Ben Yoo, S.J., “Self-similar traffic shaping at the edge router in optical packet-switched networks”, IEEE International Conference on Communications (ICC 2002), pp: 2449-2453, May 2002.

[3] Schlosser, M. Woesner, H. Schroth, A., “Simulation model for an optical burst switched network” The 13th IEEE Workshop on Local and Metropolitan Area Networks, pp. 103 – 108, April 2004.

[4] Ahmad, S. Malik, S., “Implementation of optical burst switching framework in Ptolemy simulator”, E-Tech 2004, pp: 47 – 52, July 2004

[5] Kaheel A., Alnuweiri H., Gebali F., “Analytical evaluation of blocking probability in optical burst switching networks”, 2004 IEEE International Conference on Communications, pp. 1548 - 1553 June 2004.

[6] Jing Teng, Rouskas G.N., “On Wavelength Assignment in Optical Burst Switched Networks”, First International Conference on Broadband Networks, pp: 24 – 33, Oct. 2004.

[7] Xu L., Perros H.G., Rouskas G.N.. “A queueing network model of an edge optical burst switching node”, IEEE INFOCOM 2003, pp: 2019 – 2029 April 2003.

[8] Turner, J.S., “Terabit Burst Switching”, Journal of High Speed Networks, 1999.

[9] Gauger, C.M. Kohn, M. Scharf, J., “Comparison of contention resolution strategies in OBS network scenarios” Proceedings of 2004 6th International Conference on Transparent Optical Networks, pp. 18 - 21 July 2004.

[10] Myungsik Yoo Chunming Qiao Dixit, S., “The effect of limited fiber delay lines on QoS performance of optical burst switched WDM networks”, IEEE International Conference on Communications pp: 974 - 979, June 2000.

[11] John A. Miller, Andrew F. Seila and Xuewei Xiang, "The JSIM Web-Based Simulation Environment," Future Generation Computer Systems, Vol. 17, No. 2, pp. 119-133. Oct. 2000.

[12] Willinger, W., Taqqu, M.S., Sherman, R., Wilson, D.V., “Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level”, IEEE/ACM Transactions on Networking, Vol. 5, No. 1, pp: 71 – 86 , Feb. 1997.

[13] Myungsik Yoo, Chunming Qiao, “Just-Enough-Time (JET): A high speed protocol for bursty traffic in optical networks”, In Vertical Cavity Lasers, …, 1997 Digest of the IEEE/LEOS Meetings, pp. 26–27, Aug. 1997.

[14] O’Mahony, M.J., ”Results from the COST 239 project. Ultra-High Capacity Optical Transmission Networks”, 22nd European Conference on Optical Communication, pp: 15-19, Sept. 1996

[15] Myungsik Yoo, Chunming Qiao “Supporting Multiple Classes of Service in IP over WDM Networks”, Proceedings Globecom 1999, pp. 1023 – 1027, 1999.

[16] Rosberg Z., Ha Le Vu, Zukerman M., White, J., “Performance analyses of optical burst-switching networks” IEEE Journal on Selected Areas in Communications, pp: 1187-1197, Sept. 2003.