

An OMNeT++ Model for the Evaluation of OBS Routing Strategies

A. L. Barradas
Center for Electronic, Optoelectronic and
Telecommunications (CEOT)
University of Algarve
8005-139 Faro, Portugal
abarra@ualg.pt

M. C. R. Medeiros
Center for Electronic, Optoelectronic and
Telecommunications (CEOT)
University of Algarve
8005-139 Faro, Portugal
cmedeiro@ualg.pt

ABSTRACT

Optical Burst Switching (OBS) has been proposed as a cost-effective paradigm for supporting, with adequate flexibility, the increasingly high transmission capacity required by the forthcoming next generation of optical internet networks. However, OBS efficiency can be reduced by resource contention of bursts directed to the same transmission links, leading to burst loss. This paper presents a discrete event simulation model and its implementation using OMNeT++ aimed at the evaluation of different routing strategies in terms of link contention. The simulation model is used to compare the performance of novel path selection strategies proposed in [1] with the traditional routing strategy that uses the shortest path. The simulation results confirm that the proposed path selection strategies are effective in reducing the overall network burst drop probability.

Categories and Subject Descriptors

C.2.1 [Computer-Communication Networks]: Network Architecture and Design—*Optical Burst Switching (OBS)*

General Terms

Algorithms, Design, Performance

Keywords

OBS, Routing Optimization, Path Selection Strategies, Discrete Event Simulation, Burst loss

1. INTRODUCTION

Optical Burst Switching [3] has been proposed as a switching paradigm with a number of attractive advantages over the previously proposed switching technologies for all optical transport, namely, Optical Circuit Switching (OCS) and Optical Packet Switching (OPS). By combining the merits of both while avoiding their shortcomings, OBS has attracted considerable attention from researchers as an optical architecture to support huge bandwidth demands in optical

backbones using Wavelength Division Multiplexing (WDM). Among its merits, OBS presents the following advantages when compared with their more coarse-grained OCS and more fine-grained OPS counterparts: high bandwidth utilization, low setup latency, moderate switching speed requirements, medium processing complexity and adequate adaptivity to busy traffic [9].

In OBS the basic transport unit is a burst containing a certain number of IP packets grouped by destination address criteria and assembled at the ingress node. Each data burst can be regarded as an optical super packet traveling from source to final destination without any optical-electrical-optical (OEO) conversion. For the burst delivery attempt, OBS uses out-of-band signaling with an offset time delay, leading to a separation between Control Packet Header (CPH) and data burst in both space and time, a distinguishing mark of this switching paradigm that allows for great network manageability and flexibility [2].

However, OBS presents low reliability since it generally uses a one-way reservation protocol where data bursts are transmitted without confirming the network resources reservation along the entire burst path. Therefore data bursts may contend for the same resources leading to burst drop. This is the primary cause of burst loss in OBS and it happens when the number of overlapping burst reservations at an output port of a core node exceeds the number of data wavelengths available at a specific time. Figure 1 presents a simplified view of an OBS network architecture and we refer the reader to [11, 6] for further details on the OBS technology.

Burst drop has impact on OBS performance, such as bandwidth utilization and latency, since dropping naturally leads to rescheduling of lost data and to the holding of the network resources used for data burst transmission from the source node until the dropping point. Data burst dropping can be minimized by appropriately choosing the paths that bursts must follow, that is, an effective choice of paths can lead to an overall network performance improvement.

In this paper we present a discrete event simulation model for the evaluation of routing strategies intended to minimize burst drop. The implementation of the model under OMNeT++ is discussed. The simulation model is used to compare the performance of the novel path selection strategies proposed in [1] with the traditional routing strategy

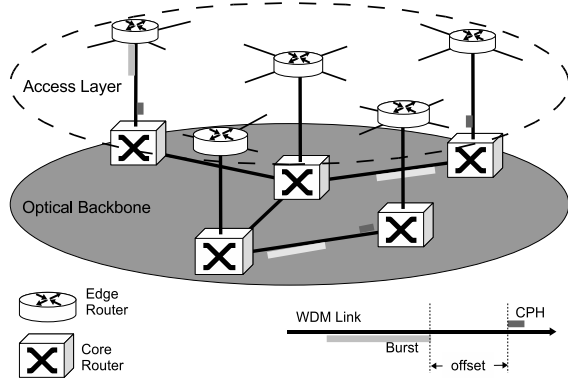


Figure 1: A simplified view of an OBS network architecture.

that uses the shortest path. Our model [1] requires only *a priori* topological network information, thus reducing burst loss probability while avoiding state dissemination protocol penalties. We consider two routing strategies. For both strategies, the path between source and destination is chosen from a set of possible paths. The sets of possible paths are the result of an optimization problem with the following objectives: (strategy 1) minimize the Maximum Congested Link (MCL) and (strategy 2) minimize the Maximum End-to-End Congested (MEC) path. Both optimization problems are solved using Integer Linear Programming (ILP). The results of the optimization are integrated in the overall network model implemented in OMNeT++.

Our results show that the novel strategies reduce burst drops when compared with the traditionally used shortest path and demonstrate that the simulation model here presented is suitable for OBS studies under the scope of the above mentioned objectives.

The paper is organized as follows. After this section, which presents a succinct introduction to some OBS intrinsic aspects, Section 2 presents the OBS simulation model. Section 3 briefly explains the path selection strategies we adopted. In Section 4 the performance of those path selection strategies is evaluated and, finally, the concluding remarks.

2. THE OBS NETWORK MODEL

The simulation model described in this paper involves the two following working stages conceptually represented on Fig.2: a first stage, during which the optimization problem is formulated and solved using ILP, and a second stage, wherein the network simulation takes place. After the first stage, whose algorithms are presented on Section 3 and from which an optimized routing solution is produced (in a text file), the second stage uses the OMNeT++ simulator¹, a C++ based public-source, component-based, modular and open-architecture simulation environment primarily tailored for modeling communication networks [10].

¹OMNeT++ is an advanced discrete event simulation system developed and supported by András Varga and a strong research community (<http://www.omnetpp.org>).

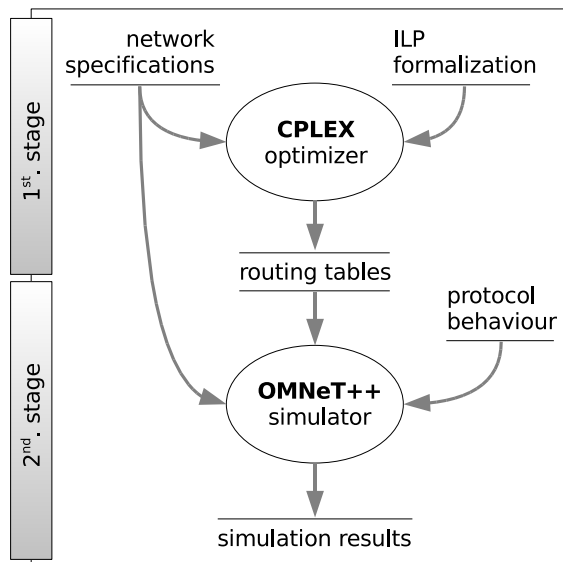


Figure 2: A conceptual view of the simulation model in two stages.

The architecture of our OBS model, similar to the one presented in [6], assumes that each node can support both new input traffic as well as *in transit* traffic passing all-optically, meaning that each node consists of both a core router and an edge router as shown in Fig.1. The network under study uses the COST 239 topology shown in Fig.3 as a large optical backbone interconnecting 11 European cities with 26 bidirectional links and an average node degree of 4.7. The nodes are connected by links representing optical fibers having $W=16$ wavelengths per link and 10 Gbit/s of transmission capacity per wavelength. The adopted traffic is based on bursts modeled by OMNeT++ messages and assumes a Poisson pattern with a threshold-based assembly method, generating messages with sizes 100×10^3 bytes.

The bursts are forwarded through the core backbone reproducing the relevant actions of the *Just-Enough-Time*(JET) [3] scheduling scheme. The control information processing time is assumed to be $10 \mu\text{s}$ per core node. Always assuming full wavelength conversion capability, simulations were done for different traffic loads starting from a very light value of 0.05 and gradually increasing until 0.95 (Erlangs). If burst scheduling fails the burst is simply dropped and no further contention resolution method is adopted after the wavelength conversion attempt. The model employs source routing, a complete routing decision taken at the ingress edge node. Like the approach adopted on [12], the control information, including the path over which the burst must travel, is carried on the CPH that precedes the transmission of each burst. Here, the adopted path is the result of one of the path selection strategies discussed on Section 3 and previously loaded into routing tables during the initialization phase of the simulation model. All the adopted values are parametrized and reconfigurable, meaning that they can be easily changed to assume different working scenarios. Together with the network topology description, the OBS model, which is essentially composed of OBS capable

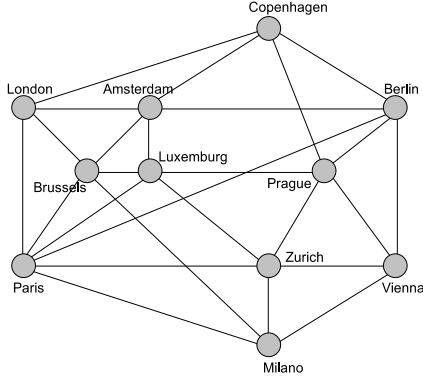


Figure 3: The COST 239 network topology.

nodes interconnected by optical fibers, is based on two main OMNeT++ compound modules, EdgeNode and CoreNode, that will be presented from the functional point of view on the next subsections.

2.1 The Edge Node

Depending on their sending/receiving status, edge nodes can be viewed either as ingress or egress nodes. When acting as ingress nodes, edge nodes are responsible for aggregating the incoming packets into bursts, for taking the initial (and also permanent) routing decision and for scheduling the bursts for transmission on outgoing channels. When acting as egress nodes, they perform the inverse operation, i.e., edge nodes are responsible for the disaggregation of the bursts back into packets and send them up for processing. In our model we assume *the burst* as the basic transport unit of interest. Hence, the issue of the packet aggregation policies is presently considered out of scope. From that fact, it is worth to mention that the traffic generator submodule is already a burst generator, generating messages based on a Poisson process with symmetric all-to-all traffic matrix. Whenever a Poisson process timer expires a new burst is generated, a destination address is chosen at random between all other nodes in the network, a route to the destination node is taken from the source node's routing table and an initial wavelength is selected among the free ones. Please note that the above-mentioned source routing decision is our first way of addressing contentions on network with an *a priori* action on the space domain. The burst, together with all its relevant information, is then retained on a queue system organized by destination address and the signaling process starts with the sending of a CPH on the appropriate channel. The CPH is always transmitted before the corresponding burst and apart from it by an offset time. The model calculates this offset time in order to allow the CPH to be processed at each subsequent node before the burst arrival and in such a way that an optical path can be properly reserved for burst delivery.

The internal structure of the edge node is shown in Fig.4, where a configuration for only 3 wavelengths is presented although 16 are used in our simulation study. Its functionality

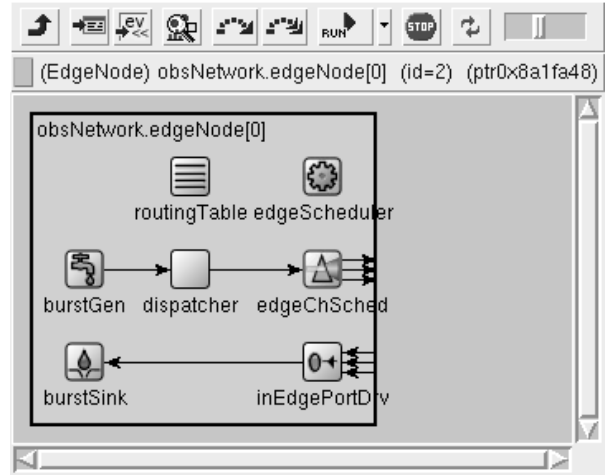


Figure 4: The EdgeNode submodules (with only 3λ).

is implemented by the following submodules:

RoutingTable holds the routing information of the node and all the related protected and public functions. In our study, the routing information is made of complete paths extracted from the previous mentioned ILP solution text file. This table is stored using STL vectors and is loaded only once during the initialization phase of the simulation. The module takes two parameters: the path selection strategy and the number of routes considered to each destination.

BurstGen is responsible for the traffic that each node generates. This traffic represents IP packets arriving from several different sources already assembled on the access network into burst units. The traffic pattern and the loading factor are among the most important parameters of this submodule.

Dispatcher initiates the signaling process and manages a system of queues where bursts are retained for a certain offset time. The offset time is calculated based on the processing delay introduced by the core nodes and the number of hops that the CPH will travel.

EdgeChSched is the output port driver of the edge node. This submodule transmits CPHs and bursts to the core backbone after finding a free wavelength. The number of available wavelengths and the presence of wavelength converters are among the relevant parameters of this submodule of the access network.

InEdgePortDrv receives CPHs and bursts from the core backbone. It is the signaling end point, it establishes the location where some CPH related statistics are obtained and from where data bursts are forwarded to be disassembled.

BurstSink is the submodule that receives the data bursts and collects some data burst related statistic values. This submodule represents the place for which bursts are sent to be disassembled into IP packets.

EdgeScheduler is the submodule that collects some aggregated edge node statistical information.

2.2 The Core Node

Core nodes are responsible for processing the CPH signaling messages, for switching the bursts from an input to an output port without OEO conversion and for handling burst contentions. Signaling in OBS is typically implemented using out-of-band, where the CPH associated with a data burst is transmitted on a different wavelength from the group of wavelength channels used to transmit data bursts. We use λ_0 for the transmission of the CPHs. Several signaling schemes have been proposed by researchers but *Just-in-Time* (JIT) and *Just-Enough-Time* (JET) are two of the most popular protocols using distributed signaling on OBS. These are both one-way and source initiated signaling schemes, which means that the bursts are sent to the core network without waiting for acknowledgments regarding the success or failure of the reservation attempts. Although closely related, they differ on the duration of the reservations. JIT uses immediate reservation with the data channel being reserved immediately since the moment the CPH reaches the node, while JET delays the channel reservation until the burst arrival. This technique, together with the implicit release, makes JET more efficient than JIT regarding bandwidth utilization, resulting in lower blocking rates and low end-to-end delay [6]. For these reasons our model is presently running under a JET-kind behavior scheme, but it can easily be converted into a JIT-kind behavior.

Together with burst forwarding without leaving the optical domain, core nodes are also responsible for taking contention resolution actions. Contention occurs when multiple bursts from different sources are destined for the same output port at the same time [4]. Adding to the initially path selection strategies adopted on the edge nodes, the handling of burst contentions used on the core nodes assume (by default) full wavelength conversion. This means that any incoming wavelength can be shifted to any outgoing wavelength, i.e. there is no wavelength continuity constraint on the end-to-end requests. As a result, only if there is no wavelength available on the output port the burst will be dropped without any further contention resolution action.

The internal structure of the core node is illustrated in Fig.5, where a configuration for only 3 wavelengths is presented although our simulation study uses 16 wavelengths. It's functionality is implemented by the following submodules:

InCorePortDrv is the entry point of the core backbone. It is an input port driver that receives CPHs and bursts from the access network and forwards them, after an increment in the number of hops, to the proper switch unit. This submodule is also the place where information related to the node demands is obtained.

SwitchUnit is the submodule where switching takes place and the incoming CPHs and data bursts are directed to the proper output port towards the next hop. Although for data bursts this is done in the optical domain for CPHs OEO conversion is involved. This submodule holds an internal switching table, loaded during the initialization phase of the simulation and stored in an STL map that relates the target address with its correspondent gate identification. Besides the switching table, the CPH processing delay and the number

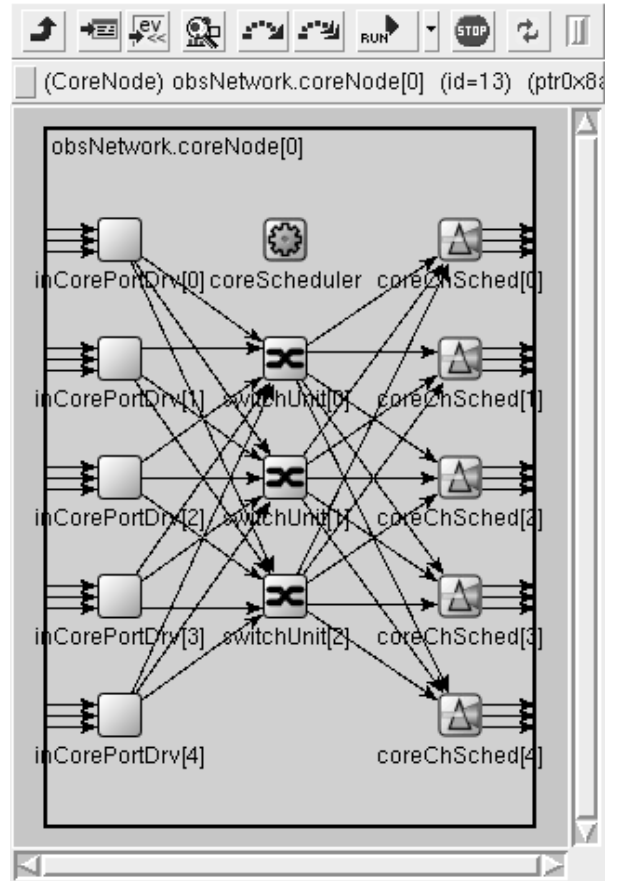


Figure 5: The CoreNode submodules (with only 3 λ).

of ports are among its most important parameters.

CoreChSched is the output port driver of the core node. This submodule forwards CPHs and bursts to the next core node of the backbone or to the local edge node of the current core node. For that, it tries to find a free wavelength by checking the wavelength availability of the port in a word of flags implemented through the use of an STL bit set. This is also the place where contention for resources occurs and the number of drops is obtained. The total number of wavelengths and the presence of wavelength converters are among the relevant parameters of this submodule of the backbone.

CoreScheduler aggregates statistical information related to the current functioning of the core node, for instance, the total number of hits, number of hits per port, the total number of drops or the number of drops per port.

3. THE PROPOSED ROUTING STRATEGIES

Contention resolution schemes in OBS networks can act in three possible dimensions: space, time and wavelength [4]. Our approach acts on the space and wavelength domain, first through the adoption of an optimized routing scheme and after, whenever is needed (and possible), through the use of wavelength conversion. Proposals on the subject can be

broadly classified into two categories: multi-path strategies combined with final delivery schemes to overcome out-of-order burst arrivals at the egress nodes [5, 8], and single-path strategies in which a unique primary path is used to route bursts between each pair of nodes [12, 7]. Our approach adopts single-path strategies with the objective of minimize the overall burst blocking probability by appropriately selecting the path over which a burst must travel. To reach this objective the two earlier mentioned strategies, MCL and MEC, were tested. These strategies were presented on [1] and consist on the results of two ILP problems resolved using the CPLEX² optimizer. For both strategies, input information includes a set of paths for each pair of nodes. Each strategy must select one path for every pair of nodes, from the given set, so that its objective is achieved. That is, for the overall network, $N(N-1)$ paths must be calculated and allocated for burst delivery. MCL and MEC were compared under a dynamic scenario with an OBS network using the shortest path approach, the one used by most OBS studies.

In the following discussing let $\mathcal{G}(\mathcal{N}, \mathcal{L})$ be a network graph, where \mathcal{N} is the set of nodes and \mathcal{L} is the set of links, and let us define a path over which a burst must travel, v , as a connected series of directed links, written as $v : s(v) \rightarrow d(v)$, from source node $s(v)$ to destination node $d(v)$. The set of paths that can be used by a burst from s to d is defined as $\mathcal{V}_{s,d} = \{v : s(v) \rightarrow d(v) \mid s = s(v), d = d(v)\}$ and the set including all $\mathcal{V}_{s,d}$ is defined as \mathcal{V} . We also define $p_l^v = 1$ if link $l \in \mathcal{L}$ is included in v , $p_l^v = 0$ otherwise, and $q^{v,v'} = 1$ if the two paths v and v' share at least one link. A demand matrix T can also be considered, where $t_{s,d}$ represents a relative load from source node s to destination node d . We note that the following formulations are independent of the details of the demand model, which may include the total or average number of demands, over a period of time, or some integer value that reflects the local demand weight over the total network demand.

3.1 Strategy I: Minimize MCL

This strategy is based on the idea that the more a certain link is included in the chosen paths for source-destination pairs, the highest the blocking probability will be. Therefore, paths for source-destination pairs should be selected with the objective of minimizing the blocking probability of the link with highest expected contention value, denoted by ζ_{MAX} . This is achieved by the following ILP optimization problem:

$$\text{Minimize } \zeta_{MAX} \quad (1)$$

Subject to

$$\sum_{v \in \mathcal{V}_{s,d}} \sigma^v = 1, \quad \forall s, d \in \mathcal{N} \quad (2)$$

$$\sum_{s,d} \sum_{v \in \mathcal{V}_{s,d}} \sigma^v \times p_l^v \times t_{s,d} \leq \zeta_{MAX}, \quad \forall l \in \mathcal{L} \quad (3)$$

$$\sigma^v \in \{0, 1\}; \text{ non-negative integer: } \zeta_{MAX} \quad (4)$$

where σ^v is a binary variable that indicates if v is used to carry bursts from node $s(v)$ to node $d(v)$. Constraint (2) states that one path must be found for each pair of nodes. Each path is selected from the corresponding set $\mathcal{V}_{s,d}$ of available paths. Constraint (3) states that the expected congestion at a link must not exceed ζ_{MAX} .

3.2 Strategy II: Minimize MEC

This strategy is based on the idea that blocking may occur at any link traversed by a burst along the path. Therefore, paths for source-destination pairs should be selected so that demands have the smallest probability of contending with other demands at every link from source to destination, minimizing the end-to-end blocking. This is achieved by the following ILP optimization problem, where φ_{MAX} denotes the contend value of the path having the highest number of contends.

$$\text{Minimize } \varphi_{MAX} \quad (5)$$

Subject to

$$\sum_{v \in \mathcal{V}_{s,d}} \sigma^v = 1, \quad \forall s, d \in \mathcal{N} \quad (6)$$

$$\eta^{v,v'} \geq (\sigma^v + \sigma^{v'} - 1) \times q^{v,v'}, \quad \forall v \in \mathcal{V}, \\ , \forall v' \in \mathcal{V} \setminus \mathcal{V}_{s(v),d(v)} \quad (7)$$

$$t_{s,d} + \sum_{v \in \mathcal{V}_{s,d}} \sum_{v' \in \mathcal{V} \setminus \mathcal{V}_{s,d}} \eta^{v,v'} \times t_{s(v'),d(v')} \leq \varphi_{MAX}, \\ , \forall s, d \in \mathcal{N} \quad (8)$$

$$\sigma^v, \eta^{v,v'} \in \{0, 1\}; \text{ non-negative integer: } \varphi_{MAX} \quad (9)$$

where σ^v is a binary variable that indicates if v is used to carry bursts from node $s(v)$ to node $d(v)$, and $\eta^{v,v'}$ is a binary variable that indicates if v and v' have both been selected to carry bursts and share at least one link. Similarly to the previous strategy, constraint (6) states that one path must be found for each pair of nodes. Constraint (7) forces $\eta^{v,v'}$ to be 1 if v and v' share a link and have both been selected to carry bursts. Otherwise, and due to the minimizing nature of the objective function, $\eta^{v,v'}$ will be 0. Constraint (8) states that the contending value of a source-destination pair must not exceed φ_{MAX} .

As already stated, for these path selection strategies a set of paths must be given, as input, for each pair of nodes. In this study we propose the use of the shortest paths with less links in common. That is, if several paths exist with an equal

²CPLEX is an advanced mathematical programming and constraint-based optimization engine from ILOG (<http://www.ilog.com>).

Strategy I: minimize MCL

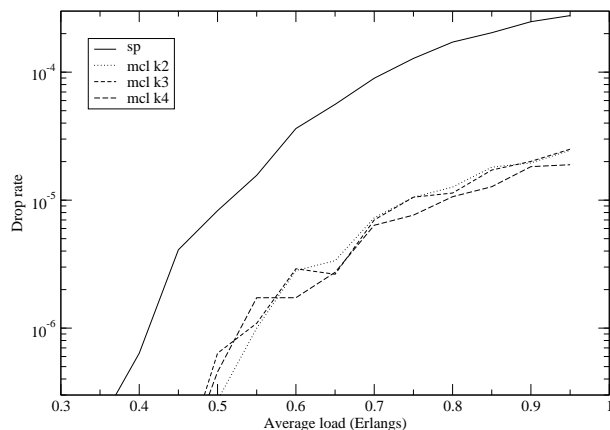


Figure 6: Comparative evaluation of the MCL strategy vs. shortest path.

number of hops then the more distinct ones are chosen. Note that this task is independent of the path selection strategies since different paths can be supplied to the strategies, either resulting from detailed network performance observations or from any other criteria adopted by network administrators.

4. EVALUATION OF THE ROUTING STRATEGIES

Simulations were done for $k=2$, $k=3$ and $k=4$, where k is the number of shortest paths, per pair of nodes, being provided to the path selection strategies. The results are shown in the plots of Fig.6 and Fig.7 where the strategies under evaluation, MCL and MEC, are compared with the traditionally used shortest path approach. From such plots it is easily perceptible that both strategies reduce the number of bursts being dropped, clearly outperforming the results achieved with the shortest path.

The plots also show that while for MCL the drop probability slightly decreases with the increase of K , indicating that the algorithm benefited from the alternative paths given as input, with MEC the opposite happens showing that the longer paths adopted can be a disadvantage resulting in less gain. To some extent this was already expected because burst scheduling is required at each intermediate node and the longer paths, determined by higher values of K , can also correspond to more contention possibilities.

5. CONCLUSIONS

In this paper a simulation framework for the evaluation of routing strategies on OBS networks is presented. The model considers a two stage approach beginning with the optimization of the routing tables of the nodes and following with its evaluation by means of a simplified OBS network model developed using the OMNeT++ simulator.

Two path selection strategies for OBS networks using only topological network information are presented and evaluated. The objective of both strategies is to minimize the burst loss due to resource contention of bursts aimed at the

Strategy II: minimize MEC

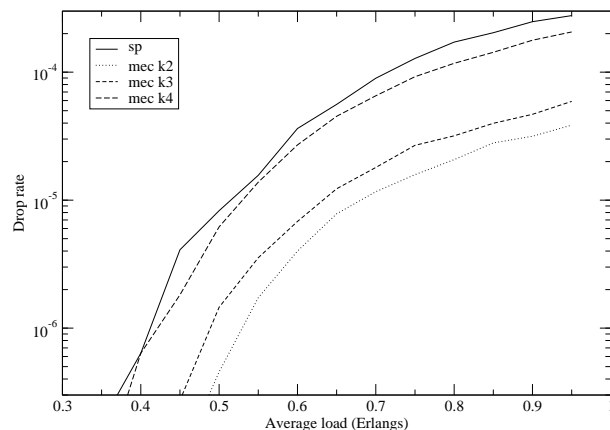


Figure 7: Comparative evaluation of the MEC strategy vs. shortest path.

same output link. It is demonstrated that it is possible to achieve an initial stage of improved performance, measured in terms of burst loss reduction, without incurring into link state dissemination protocol penalties. It is also showed that the simulation model is appropriate for the conducted comparative studies.

Future work can include the adoption of other contention avoidance strategies, including dynamic contention resolution schemes and extending the scope of the model to the packet aggregation policies as well. As might be expected, these goals will increase the complexity of the modules, in both core and edge nodes, but with the OMNeT++ simulation environment and some programming effort it can be positively accomplished.

6. ACKNOWLEDGMENTS

The work reported in this paper was supported in part by the Foundation for Science and Technology (Portugal) within Center for Electronic Optoelectronic and Telecommunications, University of Algarve (Portugal) under the project POSC/EEA-CPS/59556/2004

7. REFERENCES

- [1] A. L. Barradas and M. C. R. Medeiros. Path selection strategies for OBS networks using topological network information. In *ICTON-MW'07*, page Fr1B4, Sousse, Tunisia, December 2007.
- [2] T. Battestilli and H. Perros. An introduction to optical burst switching. *Communications Magazine, IEEE*, 41(8):S10–S15, August 2003.
- [3] Y. Chen, C. Qiao, and X. Yu. Optical burst switching: a new area in optical networking research. *Network, IEEE*, 18(3):16–23, May–June 2004.
- [4] K. C. Chua, M. Gurusamy, Y. Liu, and M. H. Phung. *Quality of Service in Optical Burst Switched Networks*. Springer, 2007.
- [5] S. Ganguly, S. Bhatnagar, R. Izmailov, and C. Qiao. Multi-path adaptive optical burst forwarding. In *High Performance Switching and Routing, 2004. HPSR*.

- 2004 *Workshop on*, pages 180–185, 2004.
- [6] J. P. Jue and V. M. Vokkarane. *Optical Burst Switched Networks*. Springer, 2005.
 - [7] J. Pedro, J. Castro, P. Monteiro, and J. Pires. On the modelling and performance evaluation of optical burst-switched networks. In *Computer-Aided Modeling, Analysis and Design of Communication Links and Networks, 2006 11th International Workshop on*, pages 30–37, June 2006.
 - [8] J. Pedro, J. Castro, P. Monteiro, and J. Pires. On fully exploiting the space domain for contention resolution/avoidance in optical burst-switched networks. In *ICTON 2007*, pages 113–116, Rome, Italy, July 2007.
 - [9] C. Qiao and M. Yoo. Optical burst switching (OBS) - a new paradigm for an optical internet. *J. High Speed Networks (JHSN)*, vol. 8, no. 1, pp. 69–84, 1999.
 - [10] A. Varga. Omnet++ discrete event simulation system. <http://www.omnetpp.org/>.
 - [11] Y. Xiong, M. Vandenhouste, and H. Cankaya. Control architecture in optical burst-switched WDM networks. *Selected Areas in Communications, IEEE Journal on*, 18(10):1838–1851, October 2000.
 - [12] L. Yang and G. Rouskas. Adaptive path selection in OBS networks. *Journal of Lightwave Technology*, 24(8):3002–3011, August 2006.