ALMA MATER STUDIORUM
UNIVERSITY OF BOLOGNA - ITALY

Ph.D. in Ingegneria Elettronica, Informatica e
delle Telecomunicazioni ING-INF/03

DESIGN AND CONTROL TECHNIQUES OF OPTICAL NETWORKS

Ph.D. thesis by

GIOVANNI MURETTO

Coordinator	*Supervisor*
Prof. PAOLO BASSI	Prof. GIORGIO CORAZZA

ACADEMIC YEAR 2004-2005

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Dottorato di Ricerca in Ingegneria Elettronica, Informatica e delle Telecomunicazioni ING-INF/03

XVII CICLO

TECNICHE DI PROGETTO E CONTROLLO DI RETI OTTICHE

Tesi di Dottorato di

GIOVANNI MURETTO

Coordinatore

Chiar.mo Prof. Ing.
PAOLO BASSI

Tutor

Chiar.mo Prof. Ing.
GIORGIO CORAZZA

ANNO ACCADEMICO 2004-2005

To Maria Degreve and her words.

Contents

Acknowledgments

I would like to thank all the members of Network Research Group. In particular Professor Giorgio Corazza, Professor Carla Raffaelli and Professor Franco Callegati for the professionalism they taught me and for the opportunity they gave me to publish and to travel around the world. Doctor Walter Cerroni for the constant help that he always gave me. Ing. Paolo Zaffoni and Ing. Michele Savi for sharing this experience with me. Overall I want to thank for the friendship I received in this three years. Thanks to my family for the support and for the effort made for me.

Bologna - March 12, 2007

Summary

The world of communication has changed quickly in the last decade resulting in the the rapid increase in the pace of peoples' lives. This is due to the explosion of mobile communication and the internet which has now reached all levels of society. With such pressure for access to communication there is increased demand for bandwidth. Photonic technology is the right solution for high speed networks that have to supply wide bandwidth to new communication service providers. In particular this Ph.D. dissertation deals with DWDM optical packet-switched networks.

The issue introduces a huge quantity of problems from physical layer up to transport layer. Here this subject is tackled from the network level perspective. The long term solution represented by optical packet switching has been fully explored in this years together with the Network Research Group at the department of Electronics, Computer Science and System of the University of Bologna. Some national as well as international projects supported this research like the Network of Excellence (NoE) e-Photon/ONe, funded by the European Commission in the Sixth Framework Programme and INTREPIDO project (End-to-end Traffic Engineering and Protection for IP over DWDM Optical Networks) funded by the Italian Ministry of Education, University and Scientific Research.

Optical packet switching for DWDM networks is studied at single node level as well as at network level. In particular the techniques discussed are thought to be implemented for a long-haul transport network that connects local and metropolitan networks around the world. The main issues faced are contention resolution in a asynchronous variable packet length environment, adaptive routing, wavelength conversion and node architecture. Characteristics that a network must assure as quality of service and resilience are also explored at both node and network level.

Results are mainly evaluated via simulation and through analysis.

Chapter 1

Introduction to optical networking

Communication has become fundamental in nowadays lifestyle, in particular wireless communication and Internet became necessary for everybody routine. Furthermore new information forms appear for the last years and new services are quickly proposed in the market assuming to work with wide bandwidth. This increasing data traffic has stimulated academic and industry research on this field. The reason why optical networking is being widely explored is that it promises a huge breakthrough in terms of capacity. Consequently optical switching is what telecommunication providers need to satisfy the always increasing bandwidth request. From the network point of view optical fiber is the physical support that offers wide bandwidth needed in the current use of Internet. In addition the introduction of Dense Wavelength Division Multiplexing (DWDM) technique allows to efficiently utilize the available fiber bandwidth, increasing the aggregate system capacity and throughput over that existing fiber[NW96]. Basically it permits to split the entire bandwidth of a fiber in many spatially equivalent wavelengths. The information flows can then share these wavelengths improving the efficiency of the bandwidth exploitation. Furthermore many recent achievements in optical technology allow to design new switching techniques. Wavelength circuit switching is already implemented in real networks but unlikely it will be able to handle the near future networks traffic. In this sense optical packet switching (OPS) is a medium-long solution that is being

explored in the current researches and it is the concept which the work of this thesis is based on. Circuit switching in optical are based on the concept of *lightpaths* [RS] that is a sequence of wavelengths reserved to connect two end points of the network. Packet switching considers a finer granularity represented by optical packets meant as aggregation of IP datagram coming from external electronic networks. OPS can be considered a flexible solution for these high-speed core network implementation thanks to their ability of exploiting resources with statistical multiplexing. OPS is thought to serve backbone networks that convert and assembly the electronics packets of the external world into optical burst at the edge nodes. The new optical units are then sent to the network where optical core nodes process them on a per packet basis. The implementation of OPS networks involves efforts in different areas, ranging from components to systems and to traffic models that are able to represent application needs. Different applications put constraints on optical switching systems, as demonstrated by significant research efforts [GRG+98],[DDC+03],[aN06]. All these issues are strongly influenced by the issue regarding the network scenario that is the packet format. This is still an issue under discussion in the scientific community and the alternatives that appear to be more appealing are *asynchronous variable length packets* (AVLP), *synchronous and fixed length packets* (FLP) and, more recently, variable length packets fit into trains of slots that are switched as a whole (*called slotted variable length packets* or SVLP). All these formats have pros and cons, mainly related to the synchronization and interworking with legacy network issues[CCM+04]. The choice made in this thesis is the AVLP. Under this assumption a crucial topic investigated is contention resolution. Contention arises when two or more packets contend for the same resource at the same time. This leads to the so called *Wavelength and Delay Selection* problem that is studied at the single node level in chapter 2. Wavelength conversion is also needed to optical packet switching and the key component is represented by the Tunable Wavelengths Conversion (TWC) whose application to solve contention in the wavelength domain has been widely studied[DHS98]. This problem is presented together with proposals for feasible node architectures in chapter 3. The dissertation moves then towards a network perspective by studying adaptive routing algorithms for an entire mesh network. Problems strictly correlated with this last issue, as quality of

service and resilience, are also explored in chapter 4. The order of these issues does not necessarily reflect the chronological order which they were studied with. But for a matter of clarity it has been decided to show results concerning with a single node approach first, followed by results from network point of view. Last chapter is dedicated to some other works related of course with optical packet switching but not strictly including in the first three.

Chapter 2

WDS algorithms and their impact on packet sequence in optical packet-switched networks

This chapter deals with optical packet switches with limited buffer capabilities, subject to asynchronous, variable-length packets and connection-oriented operation. The focus is put on buffer scheduling policies and queuing performance evaluation. In particular a combined use of the wavelength and time domain is exploited in order to obtain contention resolution algorithms that guarantee the sequence preservation of packets belonging to the same connection. Simple algorithms for strict and loose packet sequence preservation are proposed. Their performance is studied and compared through loss probability, packet sequence preservation and delay jitter.

2.1 Introduction

The work is performed assuming a network architecture consisting of Optical Packet Switching (OPS) facilities exploiting a DWDM transmission plane and designed to carry IP traffic by means of integration with a Generalized MultiProtocol Label Swithcing (GMPLS) control plane [Man03]. The network operation is therefore connection-oriented and

the OPS nodes switch Label Switched Paths (LSPs). Standard routing protocols are used to set up LSPs and optical technologies are used in switching and transmission, providing very high data rate and throughput. Each LSP represents a top-level explicitly routed path formed by an aggregation of lower-level connections including several traffic flows [KR02]. Optical packets are assumed to be asynchronous and of variable length, meaning that they may encapsulate one or more IP datagrams. This implies the availability of an all-optical switching matrix able to switch variable-length packets. We do not deal with specific implementation issues and we consider a general OPS node architecture with full connectivity and wavelength conversion capabilities. Contention resolution may be achieved in the time domain by means of optical queuing and in the wavelength domain by means of suitable wavelength multiplexing. Optical queuing is realized by Fiber Delay Lines (FDLs) that are used to delay packets contending for the same output fiber in case all wavelengths are busy. In [CCC01] it has been shown that, by using suitable contention resolution algorithms able to combine the use of the time and the wavelength domain, it is possible to improve the performance up to an acceptable level, with a limited number of FDLs. We refer to them as *Wavelength and Delay Selection* (WDS) algorithms. They will be classified in section 2.4.1 in a packet sequence perspective. Moreover these concepts may be effectively extended to a connection-oriented network scenario, for instance based on MPLS. In this case, a suitable design of dynamic allocation WDS algorithms permits to obtain fairly good performance, by exploiting queuing behaviors related to the connection-oriented nature of the traffic, but with significant savings in terms of processing effort for the switch control with respect to the connectionless case [CCRZ03]. The main drawback of previously proposed WDS algorithms is that *out-of-sequence delivery* of packets belonging to the same traffic flow cannot be avoided. The occurrence of out of order delivery raises performance problems for the end-to-end transport protocols and/or issues of implementation complexity if re-ordering at the edges of the optical network should be implemented. A possible approach to these problems, with minimum impact on the OPS network, may be to assume that the solution is left at the endpoints of the communication or at the edge-nodes of the OPS cloud. This assumption in our view is not very realistic. Moreover, a too high packet loss probability would

just make things even worse. Therefore we argue that, up to a certain extent, congestion as well as the effects that it has on the packet stream, such as packet dropping, delay jitters and out-of-order delivery, should be controlled directly in the OPS network nodes. We realize that, while packet loss probability is a performance parameter that does not require specific discussion, the measure of delay jitters and also the concept of out-of-sequence packet need some more precise definitions in order to be used as a performance indicators, as we provide in next sections.

2.2 Networking Scenario: MPLS/OPS

The Multi-Protocol Label Switching (MPLS) architecture [RVC01] is based on a partition of the network layer functions into *control* and *forwarding*. The control component uses standard routing protocols to build up and maintain the forwarding table, while the forwarding component examines the headers of incoming packets and takes the forwarding decisions. Packets coming from client layers are classified into a finite number of subsets, called *Forwarding Equivalent Classes* (FECs), based on identification address and quality of service requirements. Each FEC is identified by an additional *label* added to the packets. Unidirectional connections throughout the network, called *Label Switched Paths* (LSPs), are set up and packets belonging to the same FEC are forwarded along these LSPs according to their labels. On each core node, simple label matching and swapping operations are performed on a precomputed LSP forwarding table, thus simplifying and speeding up the forwarding function. On the other hand, optical packet switching (OPS), in order to be feasible and effective, requires a further partitioning of the forwarding component into *forwarding algorithm* and *switching* [G+98]. The former corresponds to the label matching that determines the next hop destination and the latter is the physical action of transferring a datagram to the proper output interface. The main goal of this separation is to limit the bottleneck of electro-optical conversions: the header is converted from optical to electrical and the execution of the forwarding algorithm is performed in electronics, while the payload is optically switched without electrical conversion. This chapter considers a DWDM network integrating MPLS and OPS, which relies on optical routers that

exploit the best of both electronics and optics. Standard routing protocols are used as the (non critical) routing component, MPLS labels in the forwarding algorithm (where strict performance limits are present) and, finally, optical technologies are used in switching and transmission, providing very high data rate and throughput. In order to avoid scalability problems, we assume that each LSP represents a top-level explicitly routed path, formed by an aggregation of lower-level connections including several traffic flows [KR02], and that the number of LSPs managed by a single optical core router is not so high to affect the correct label processing. We also assume the availability of an optical switching matrix able to switch variable-length packets [C+01]. This chapter is not supposed to deal with implementation issues. Therefore, a generic non-blocking architecture for an OPS node is assumed, which provides full wavelength conversion. The switch is assumed to use a feed-forward optical buffering configuration [HCA98], realized by means of B fiber delay lines. Basically, an output queuing approach is assumed: each wavelength on each output fiber has its own buffer. This results into a logical queue per output wavelength. In principle, a set of delay lines per output wavelength could be deployed, leading to a fairly large amount of fiber coils. However, a single pool of FDLs may be used in WDM for all the wavelengths on a given fiber or for the whole switch. The delays provided are linearly increasing with a basic delay unit D, i.e. $D_j = k_j D$, with k_j an integer number and $j = 1, 2, \ldots, B$. For the sake of simplicity we assume a *degenerate buffer* [HCA98] with $k_j = j - 1$ (see Fig. 2.1), but the ideas presented here are valid also for *non-degenerate buffers*, i.e. with different arrangements of k_j. This buffering scheme is applicable both to a feed-forward and to a feed-back architecture of the switching matrix and is chosen for the case study because it is the most typical architecture for FDL buffer. Nonetheless the concepts behind this work are not bound to this architecture and may be applied seamlessly to other buffer architectures.

The typical behavior of the Switch Control Logic (SCL) is that when a packet arrives and a wavelength is available, it is obviously served immediately. When a packet arrives and all wavelengths are busy it is buffered (i.e. delayed). Call t the time of the arrival and t_f the time at which the chosen wavelength will be available to serve the new packet; in principle the new packet should be delayed by an amount equal to $t_f - t$.

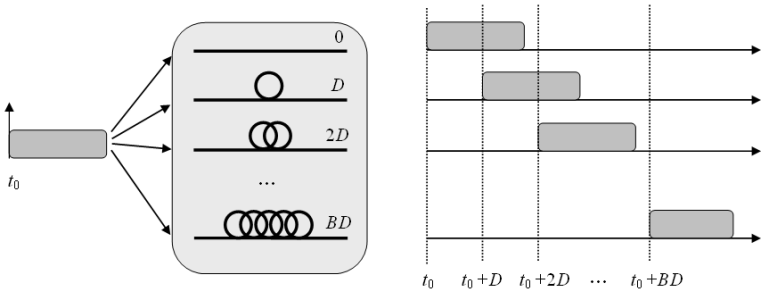

Figure 2.1: Degenerate buffer structure.

Unfortunately a finite set of delays is available, due to the limited time resolution of the delay lines. Therefore the new packet is going to be delayed by an amount

$$\Delta = \left\lceil \frac{t_f - t}{D} \right\rceil D \tag{2.1}$$

As a result, for $\tau = \Delta - t_f + t \geq 0$, the output line is not used while there would be a packet to transmit. As explained in [Cal00], this creates gaps between queued packets that can be considered equivalent to an increase of the packet service time, meaning an artificial increase in the traffic load (*excess load*). It has been demonstrated [TYC+00],[CCC01] that a WDS algorithm that aims at minimizing those gaps gives better performance in terms of packet loss probability with respect to other policies. Nonetheless no analysis was provided concerning delay jitters and out-of-sequence packet delivery on these algorithms and this is the scope of the work illustrated in this chapter.

2.3 Problems due to out-of-order packet delivery

As already outlined in the introduction, it is well known that packet losses as well as out-of-order packet deliveries and delay variations affect end-to-end protocols behavior and may cause throughput impairments [JID+03] [LG02]. In particular, the problem of packet re-sequencing

is not negligible in optical packet-switched networks, especially when optical packet flows carrying traffic related to emerging, bandwidth-demanding, sequence-sensitive services, such as grid applications and storage services, are considered. When considering TCP-based traffic these phenomena influence the typical congestion control mechanisms adopted by the protocol and may result in a reduction of the transmission window size with consequent bandwidth under-utilization. In particular the TCP congestion control is very affected by the loss or the out-of-order delivery of bursts of segments. This is exactly what may happen in the OPS network where traffic is typically groomed and several IP datagrams (and therefore TCP segments) are multiplexed in an optical packet, because optical packets must satisfy a minimum length requirement to guarantee a reasonable switching efficiency. Therefore out-of-order or delayed delivery of just one optical packet may result in out-of-order or delayed delivery of several TCP segments triggering (multiple duplicate ACKS and/or timeouts that expire) congestion control mechanisms and causing unnecessary reduction of the window size. Another example of how out-of-sequence packets may affect application performance is the case of delay-sensitive UDP-based traffic, such as real-time traffic. In fact unordered packets may arrive too late and/or the delay required to reorder several out-of-sequence packets may be too high with respect to the timing requirements of the application. These brief and simple examples make evident the need to limit the number of unordered packets. In general out-of-order delivery is caused by the fact that packets belonging to the same flow of information can take different paths through the network and then can experience different delays [BPS99]. In traditional connection-oriented networks, packet reordering is not an issue since packets belonging to the same connection are supposed to follow the same virtual network path and therefore are delivered in the correct sequence, unless packet loss occurs.

2.4 The wavelength and delay selection problem

The electronic SCL takes all the decisions regarding the configuration of the hardware to realize the proper switching actions. Once the forwarding

component has decided to which output fiber the packet should be sent, determining also the network path, the SCL:

- chooses which wavelength of the output fiber will be used to transmit the packet, in order to properly control the output interface;
- decides whether the packet has to be delayed by using the FDLs or it has to be dropped since the required queuing resource is full.

These decisions are also routing independent and all the wavelengths of a given output fiber are equivalent for routing purposes but not from the contention resolution point of view. The choices of wavelength and delay are actually correlated: since each wavelength has its own logical output buffer, choosing a particular wavelength is equivalent to assigning the packet one of the available delays on the corresponding buffer. This is what we call the *Wavelength and Delay Selection* (WDS) problem. Here we assume that, once the wavelength has been chosen using a particular policy, always the smallest delay available after the last queued packet on the corresponding buffer is assigned. The smallest delay available on a given wavelength can be easily computed using the smallest integer greater than or equal to the difference between the time when the wavelength will be available again and the packet arrival time, divided by the buffer delay unit as expressed in formula 2.2. This operation provides also the gap between the current and the previous queued packet. The choice of the wavelength can be implemented by following different policies, producing different processing loads at the SCL and different resource utilizations:

- **Static**. The LSP is assigned to a wavelength at LSP setup and this assignment is kept constant all over the LSP lifetime. Therefore packets belonging to the same LSP are always carried by the same couple of input/output wavelengths. Contention on the output wavelength can only be solved in time domain by using delay lines. In this case the WDS algorithm is trivial and requires minimum control complexity. On the other hand resource utilization is not optimized since it is possible to have a wavelength of an output fiber congested even if the other wavelengths are idle.

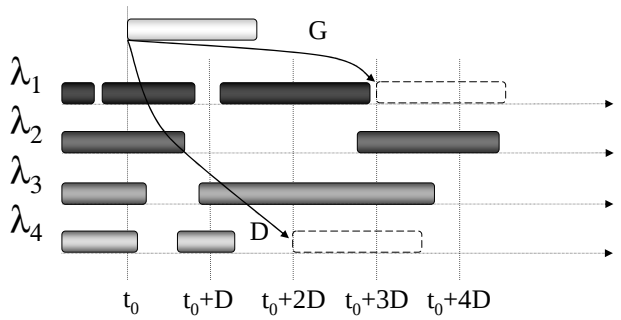

Figure 2.2: Example of the operation of a D and a G type algorithm for a fiber with 4 wavelengths and 4 delays

- **Dynamic**. The LSP is assigned to a wavelength at LSP setup but the wavelength may change during the LSP lifetime. Two approaches are possible:

  - **per-packet**. The wavelength is selected on a per-packet basis, similarly to the connectionless case, choosing the most effective wavelength in the perspective of optimizing resource usage. It requires some processing effort on a per-packet basis, therefore this alternative is fairly demanding in terms of processing load on the SCL, which must be carefully dimensioned;

  - **per-LSP**. When heavy congestion arises on the assigned wavelength, i.e. when the time domain is not enough to solve contention due to the lack of buffering space, the LSP is temporary moved to another wavelength. When congestion disappears, the LSP is switched back to the original wavelength. This alternative stays somewhat in between, aiming at realizing a trade-off between control complexity and performance

It is obvious that the per-packet alternative is the most flexible but it impacts the network more harmfully in packet sequence perspective.

2.4.1 Some WDS algorithms

In this section we introduce two classes of WDS algorithms that are characterized by a very similar computational complexity. We consider:

- *delay oriented algorithms (D-type)*, that aim at minimizing the waiting time of a queued packet and therefore act according to the principle that, when a packet has to be queued, it will join the shortest available queue (the shortest delay provided by the FDL buffer);

- *gap oriented algorithms (G-type)*, that aim at minimizing the gaps between packets and, consequently, maximizing the throughput of the switching matrix, acting according to the principle that, when a packet has to be queued, it will be sent to the delay that is closest to the transmission end of the preceding packet.

The two approaches are briefly sketched in figure 2.2, for the case of an output fiber with 4 wavelengths and 4 delays (D, 2D, 3D, 4D).

For both classes we consider three types of algorithms, characterized by a different approach to the packet sequence issue:

- *no sequence (NS type)* - the WDS algorithm does not consider LSPs and freely schedules packets without caring about their sequence;

- *loose sequence (LS type)* - the WDS algorithm is designed to grant at least a loose sequence (subsequent packets in regions 1 to 4);

- *strict sequence (SS type)* - the WDS algorithm is designed to grant a strict sequence (subsequent packets in regions 1 to 3 only).

This is shown in the example of figure 2.3 for the same case as in figure 2.2. In the following we will address any algorithm by means of the combination of the related "type" letters. For instance the D-NS algorithm will be delay oriented and unaware of the packet sequence issue.

It is obvious that by forcing the WDS algorithms to check and preserve the packet sequence, at some extent this will limit the amount of scheduling choices and therefore the achievable degree of load balancing. This will cause worse performance in term of congestion resolution capabilities and therefore a worse packet loss probability. In other words with these algorithms we trade congestion resolution performance with transparency on the packet flow.

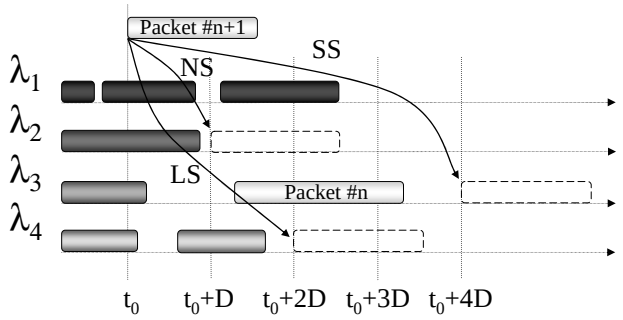

Figure 2.3: Example of the operation of NS, LS and SS type algorithms for a fiber with 4 wavelengths and 4 delays

2.4.2 Out-of-order evaluation

To check the likelihood of out-of-sequence packet delivery when one of the WDS algorithms mentioned above are implemented, we set up the two hops network scenario shown in Fig. 2.4. Every switch is identically set up with 16 wavelengths per link, an optical buffer of $B = 4$ FDLs and a granularity equal to the average packet length. The inter-arrival packet generation follows a Poisson model while the packet size is exponentially distributed with an average value corresponding to a transmission time in the order of $1\mu s$, a typical value for optical packet switching technologies. We focus on packet size only in terms of duration because the simulators used here are built to be independent from the packet size in terms of bytes and from the bit-rate. Since one of the benefits of optical switching is the transparency to the bit rate, what really matters is indeed the average packet duration and the inter-arrival time distribution. Obviously, once the optical packet duration is set, the higher the bit-rate, the higher the average packet size in bits, leading to the need for traffic grooming. The input load on each wavelength is fixed to 0.8 and the traffic distribution is uniform. The traffic input at the edge switches is ideal in the sense that packets belonging to the same LSP arrive in order on the same wavelength. At each switch, packets are processed by the selected WDS algorithm, sent to the next hop and the resulting amount of out-of-sequence packets is also evaluated. Table 2.1 shows the packet reordering distribution for the dynamic G-NS algorithm and

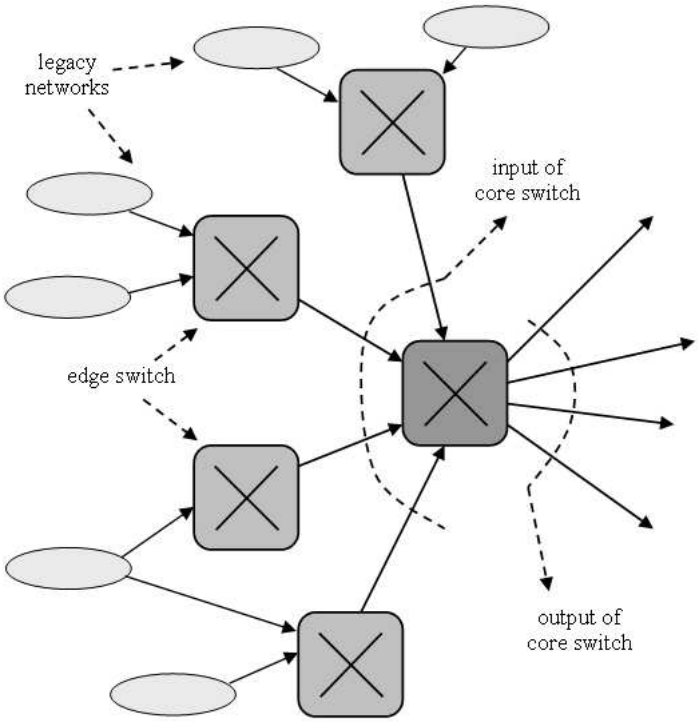

Figure 2.4: Network scenario to evaluate the amount of out-of-order packets

a static algorithm that works as explained in section 2.4.

These results, even though related to a simple network architecture, are meaningful to show that G-NS algorithm is not able to avoid sequence breaking. The percentage of packets out-of-sequence of three or more locations is already not null at the input of the core switch. By assuming n switch in series along a path this percentage is expected to increase accordingly. Previous studies [JID+03] [LG02], confirm that just a small percentage of out-of-sequence (such as that caused by EQWS algorithm) may impact harmfully on the network performance. A possible solution could be to assume that this problem is solved at the egress edge nodes that should take care of re-sequencing the various packet flows. This assumption in our view is not very realistic. It can be feasible for some flow of high value traffic, but it is unlikely that it will happen for all the flows of best effort traffic, because of the amount of memory and

Table 2.1: Out-of-order percentages at the input and output ports of the core switch, comparing a static and G-NS algorithms.

Algorithm	**Input**	**Output**
Static	0	0
MINGAP	3.6498	6.9948

processing effort that would be necessary. Therefore we argue that it becomes fundamental to control out-of-order delivery of packets directly in the OPS network nodes.

2.5 Effects of WDS algorithms on the packet sequence

The WDS algorithms may affect the packet stream in different ways by preserving strict or loose sequence as described before 2.4.1. Moreover a number of other cases are possible as a consequence of the capability of *parallel* packet transmission that goes with the idea of load balancing between wavelengths of the same fiber (possible alternatives are shown in figure 2.5).

Therefore some measuring framework is necessary to compare the behavior of different algorithms. This framework has been originally proposed in [CMR+04], and is briefly recalled here.

For a generic packet P_i crossing a given OPS node, let t_i be the arrival time at the node input, s_i the departure time from the node output and $d_i = d_p + k_i D$ the delay introduced by the node itself, due to the packet header processing time (d_p, fixed) and the possible delay inside the FDL buffer ($k_i D$, where $k_i = 0, 1, \ldots, B$). Obviously $d_i = s_i - t_i$.

Let assume that two generic subsequent packets belonging to the same traffic flow P_n and P_{n+1} arrive in order, i.e. $t_{n+1} > t_n$. Let $\Delta t_n = t_{n+1} - t_n$ and $\Delta s_n = s_{n+1} - s_n$ be the relative packet offsets at the node input and output respectively. The *jitter* between packets P_n and P_{n+1}, representing the packet offset variation due to the node crossing, may be defined as

$$J_n = \Delta t_n - \Delta s_n \tag{2.2}$$

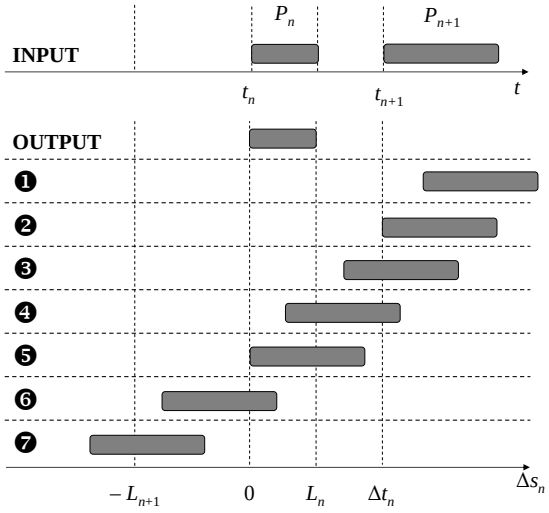

Figure 2.5: Examples of jitter between subsequent packets

Equation (2.2) may also be written as

$$J_n = d_n - d_{n+1} = (k_n - k_{n+1})D = h_n D \tag{2.3}$$

where $-B \leq h_n \leq B$. The behavior of J_n for two particular packets P_n and P_{n+1}, with length L_n and L_{n+1} respectively, is shown in figure 2.6, where the x axis has been divided in seven different regions, according also to the cases shown in figure 2.5:

1. $\Delta s_n > \Delta t_n$ when the packet sequence is always guaranteed since P_{n+1} experiences more delay than P_n ($J_n < 0$);

2. $\Delta s_n = \Delta t_n$ when the node is transparent and P_n and P_{n+1} have the same offset at the input and output ($J_n = 0$);

3. $L_n \leq \Delta s_n < \Delta t_n$ when P_{n+1} experiences less delay than P_n ($J_n > 0$) but at the output it is still behind the tail of P_n (i.e. $s_{n+1} \geq s_n + L_n$);

4. $0 < \Delta s_n < L_n$ when the head of P_{n+1} partially overlaps the tail of P_n;

5. $\Delta s_n = 0$ when P_{n+1} completely overlaps P_n ($J_n = \Delta t_n$);

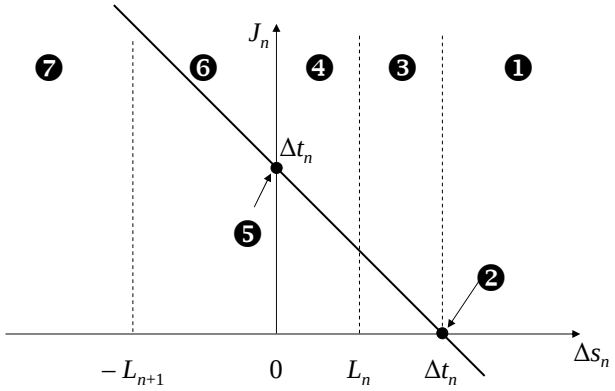

Figure 2.6: Behavior of the jitter depending on relative packet offset at node output

6. $-L_{n+1} < \Delta s_n < 0$ when P_{n+1} has overtaken P_n but they are partially overlapping (i.e. $|\Delta s_n| < L_{n+1}$);

7. $\Delta s_n \leq -L_{n+1}$ when P_{n+1} has completely overtaken P_n (i.e. $s_n \geq s_{n+1} + L_{n+1}$).

The previous formalization allows to evaluate the delay jitter distribution as well as the amount of out-of-order packets, that depends on the specific definition of packet sequence. For instance, in case overlapping packets are not considered in sequence, then the in-sequence regions will be 1, 2, and 3. If some overlapping is allowed, then the sequence is guaranteed also in region 4. The same for region 5, in case packets arriving at the same time are not considered out-of-order.

2.6 Numerical Results

In this section we provide some numerical results showing the impact of different WDS algorithms on traffic performance in terms of both packet loss probability and delay jitter, according to the framework discussed in section 2.5. These results have been obtained by simulation, using an event-driven software specifically written for this purpose. The evaluation refers to the case of loss and jitter introduced by a single OPS

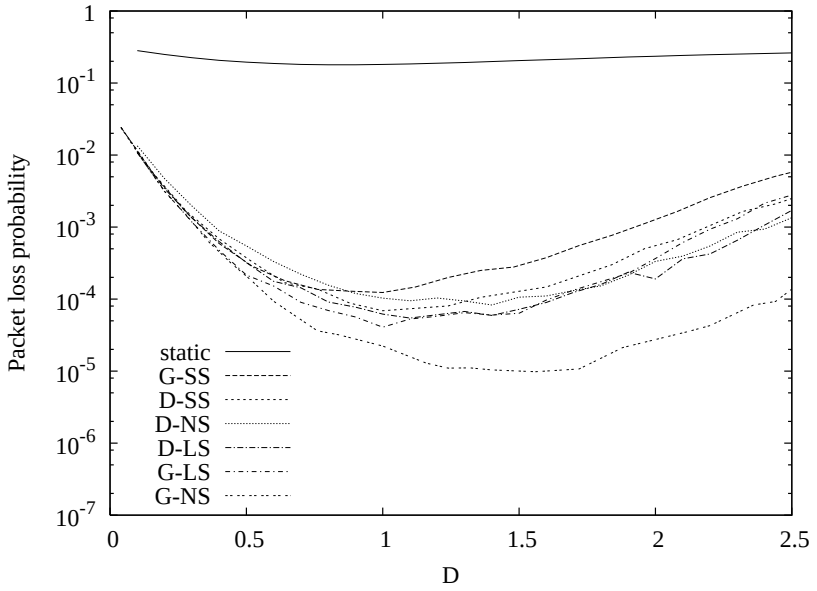

Figure 2.7: Packet loss probability for different WDS algorithms vs. the buffer delay unit D normalized to the average packet length

node, subject to Poisson arrivals of packets with exponentially distributed length. The value of the arrival packet rate and the average packet length are chosen in order to lead to an average load per wavelength equal to 0.8. The reference node has four input/output fibers, each carrying 16 wavelengths, and a 3-fiber FDL buffer. On each wavelength the traffic flow consists of packets belonging to 20 different connections (LSPs), on which the delay jitter is evaluated. The WDS algorithms are compared including a case in which no load balancing is applied, called the Static choice.

As already demonstrated in a previous work [CCC+04], figure 2.7 shows how the different WDS algorithms presented in section 2.4.1 provide different degrees of performance in terms of packet loss probability. The static choice, due to the lack of flexibility and non-effective exploitation of wavelength multiplexing, is the one that shows the poorest performance, with very high blocking rates. On the opposite, the G-NS choice gives the smallest loss rate when the FDL buffer is dimensioned optimally, i.e. when the buffer delay unit D is close to the average packet

length. This is due to the best use of the wavelength dimension made by this algorithm.

In between these two cases are the other choices, providing basically almost the same level of performance. In particular, it is worth to notice that there is no significant improvement in the packet loss rate when comparing the algorithms keeping the strict sequence (SS) to those allowing a partial overlap (LS).

With reference to the packet sequence, figure 2.8 shows the jitter distribution for the WDS algorithms considered. The results for gap-based and delay-based WDS algorithms are grouped, since they actually give the same jitter distribution. Such distribution has been evaluated on all packets that were not dropped due to congestion. The possible values of the jitter, normalized to the buffer delay unit D, range between -3 and 3 according to equation (2.3) for $B = 3$. The histograms show that the G-type and D-type algorithms do not introduce any jitter in more than 70% of the times, while in the static WDS case the null jitter happens less frequently (slightly more than 30%). This is somehow related to the different levels of packet loss provided by the algorithms: the gap-oriented and delay-oriented ones make a better use of the resources, so that many packets find a wavelength available at their arrival, which means that these packets are not buffered and cross the switch transparently.

In order to evaluate the actual impact of the algorithms on the packet sequence, figure 2.9 depicts the jitter distribution of the G-type algorithms and the static choice evaluated only on packets that were delayed (but not dropped) due to congestion. The histograms clearly show how the dynamic WDS algorithms introduce more jitter than the static one, due to the more intelligent use of the optical buffer.

Finally, figures 2.10, 2.11 and 2.12 show the jitter distribution over the different regions defined in section 2.5. As expected, static WDS succeeds in maintaining the packet sequence, as do the SS algorithms. The LS and NS algorithms also show their behavior with respect to the packet sequence. However, confirming what has been shown previously in figure 2.8, although the latter WDS policies cause some packets to get out of the node unordered, the most frequent behavior is the one related to region 2, which means that congestion happens rarely and the packets are often transmitted transparently across the node.

2.7 Last comments

In this chapter we have discussed the effect of scheduling algorithms on the sequence and time framework of a packet stream, in the scenario of an OPS network exploiting the time and wavelength domains for congestion resolution. The results provided show that it is possible to implement algorithm preserving the sequence and limiting the delay jitter with a limited degradation of the overall packet loss probability.

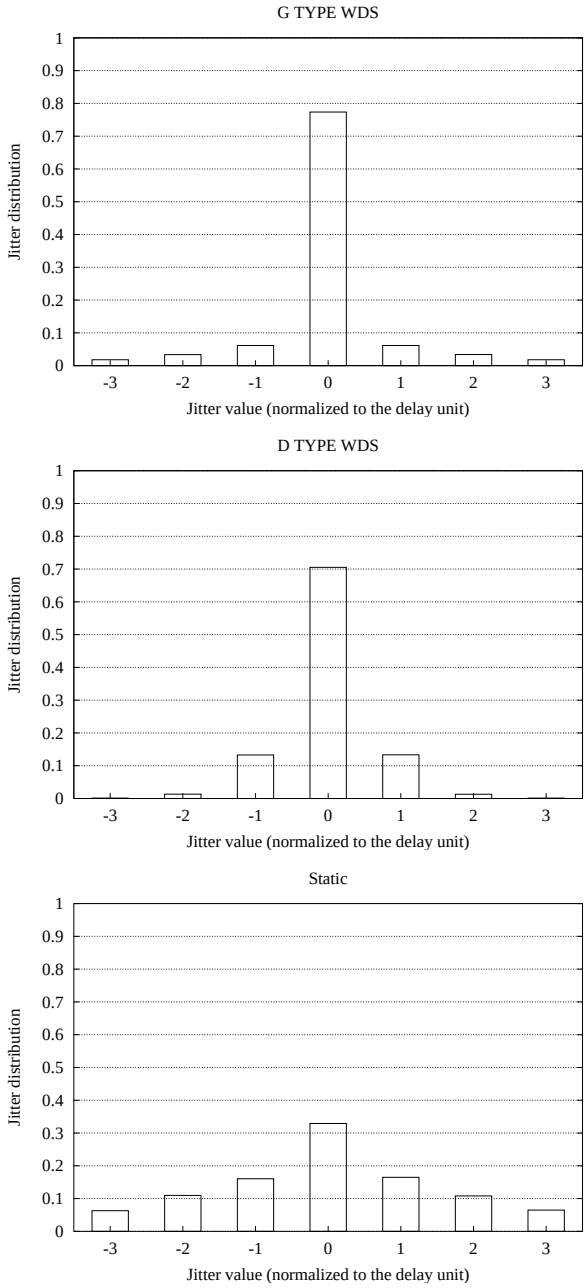

Figure 2.8: Delay jitter distribution for different WDS algorithms computed on all non-dropped packets

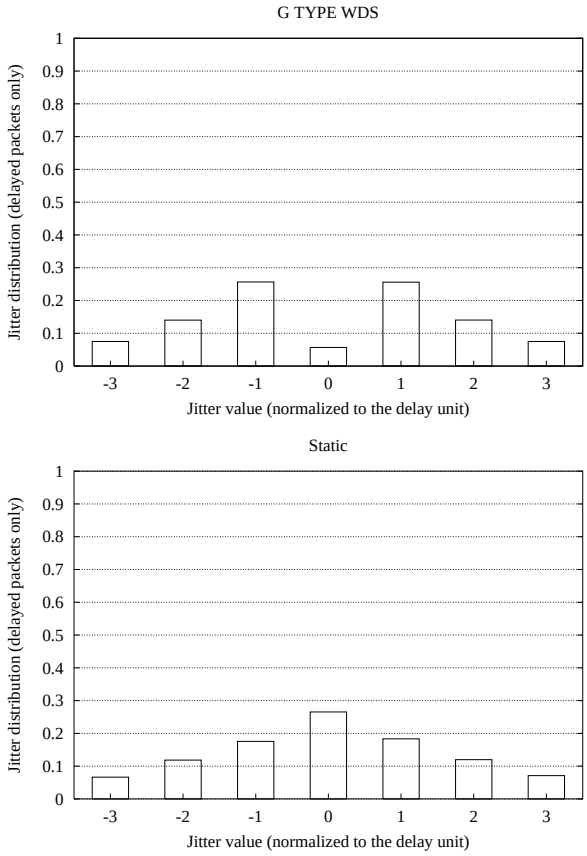

Figure 2.9: Delay jitter distribution for different WDS algorithms computed on delayed packets only

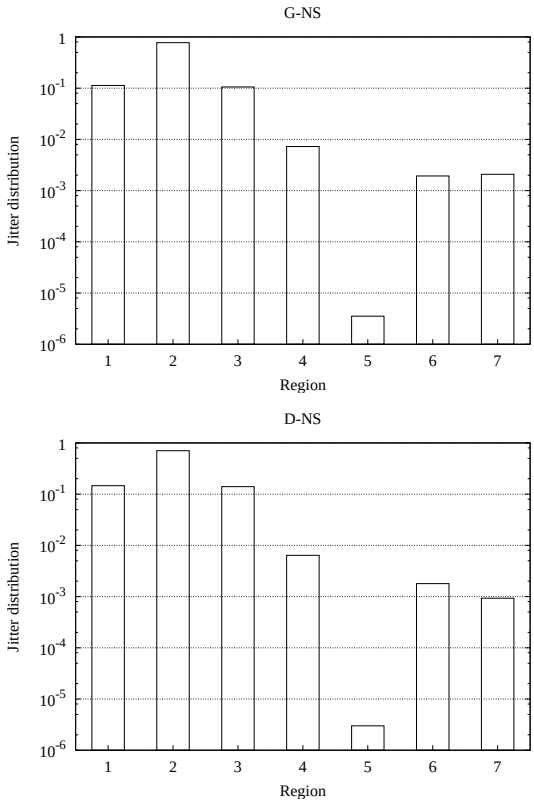

Figure 2.10: Delay jitter distribution for G-NS and D-NS WDS algorithms over the different regions shown in figure 2.6

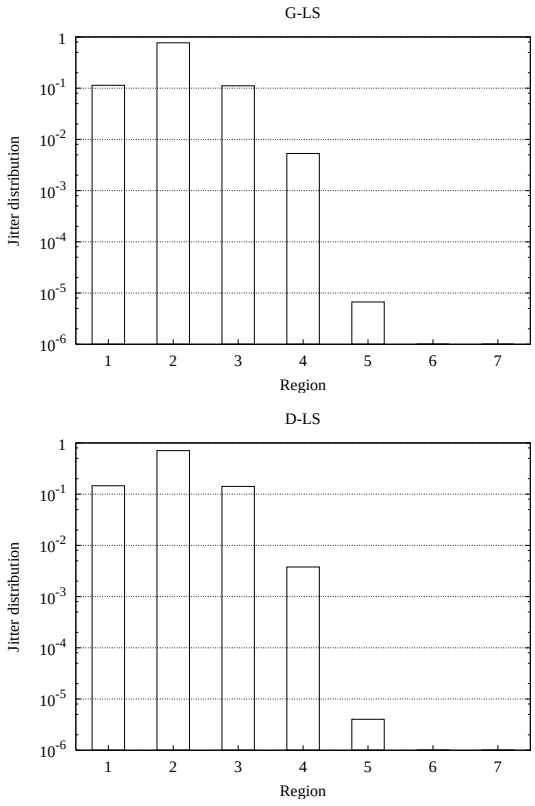

Figure 2.11: Delay jitter distribution for G-LS and D-LS WDS algorithms over the different regions shown in figure 2.6

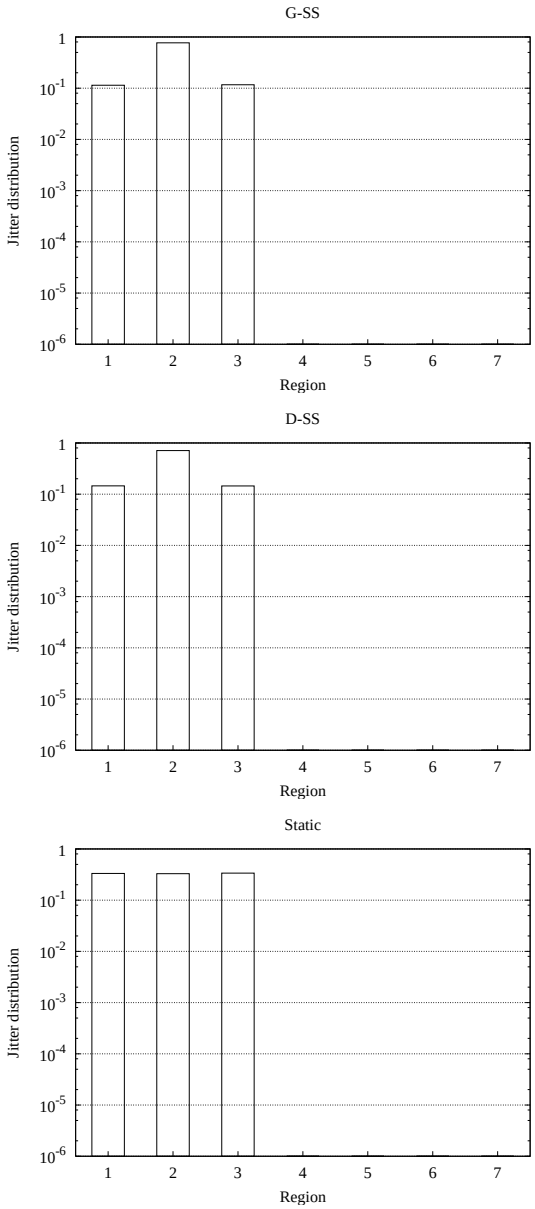

Figure 2.12: Delay jitter distribution for G-SS, D-SS and static WDS algorithms over the different regions shown in figure 2.6

Chapter 3

Contention resolution in WDM optical packet switch with multi-fiber scheme and shared converters

3.1 Introduction

One of the recognized key point in OPS design is contention resolution, intrinsically related to any packet switched system. It is well known that optical systems offer the wavelength domain to face with this problem, in addition to time and space domains. Wavelength conversion is needed to apply this concept to optical packet switching and the key component is represented by the TWC whose application to solve contention in the wavelength domain has been widely studied[DHS98]. Contention resolution in the time domain has been considered having the transposition of a well known concept in electronic optical packet switches. Anyway in the optical context this approach has some peculiarities that are different from the electronic contexts, being the queueing function implemented by means of a finite number of delays[GRG+98]. More conventional are the approaches based on space switching that typically adopt matrixes of optical gates and deflection routing[CTT99][YMD00]. To enhance space exploitation for contention resolution, some proposals, mainly appeared in the optical circuit switched context[OMSA98][SML04][WD97], adopt

multiple fiber links on each output interface. This work aims at combining different contention resolution schemes in the same optical packet switch, and tries to exploit the advantages arising from the interactions among these different approaches to design optical packet switches optimized in a cost and performance trade off. Two different switch architectures, based on tunable wavelength converters shared within the output interface or within the entire node are proposed, both equipped with multi-fibre interface. In this chapter the switch is assumed to work asynchronously with variable packet length but the option is still valid with slotted time and fixed packet length. The introduction of the multi-fiber scheme allows to reduce the number of wavelengths used on each fiber to achieve the required switch performance, that means cheaper components for termination (muxes and demuxes) and for wavelength conversion (TWC)[MR06b]. The multi-fiber switch architecture can become, however, more complicated and larger due to the introduction of some components like Mux/Demux and FDLs. The compromise between complexity, cost and efficiency is then very important. In any case in a multiservice scenario different performance are required depending on specific applications. In such contexts contention resolution is coupled to techniques that support quality of service differentiation[WD97].

3.2 Description of multi-fiber node architectures

Wavelength conversion in optical networks is key issue. Tunable wavelength converters (TWCs) are essential for the performance of the packet switch block[DHS98]. WDM needs to shift some packets from a wavelength to another in order to increase bandwidth sharing. However the increasing optical bandwidth makes conversion not easy from the technology point of view. Experimental results have shown that performance of these wavelength converters strongly depends on combination of the input and output wavelengths. That is, for a given input wavelength, translations to some output wavelengths result in an output signal which is significantly degraded[ELS06]. Moreover the wider is the range that a converter has to work with the more expensive it results. Thinking of a full wavelength conversion can therefore become not affordable. The multi-

fiber solution seems to suit with this aspect. This scheme was already explored for wavelength switching networks[LS00]. The investigation of this approach for optical packet switching in asynchronous networks is rather new. A reason to take into account this structure is that a large number of fibers are already contained in a cable underground[Odl00] so no further digging would be necessary. Furthermore multi-fiber proves to be efficient either in terms of performance and conversion cost. A study of the multi-fiber architecture with shared converters is presented in this work. Two architectures are proposed that implement different schemes for wavelength converters sharing. The first one applies the shared-per-link policy and is sketched in figure 3.1. It employs as many pools of converters as the number of output interfaces, each shared among the wavelength channels belonging to the same interface. The architecture presented in figure 2 applies the shared-per-node option and is sketched in figure 3.2. A single pool of converters is available and shared among all node channels. The external setting is the same for both architectures. It consists of N input and N output, equipped with F fibers carrying M wavelengths each. This configuration provides $F \times M$ wavelength channels per output interface. In the first case (shared per link) R indicates the number of TWCs that are available to wavelength channels switched to the corresponding interface. In the second case (shared per node) C represents the number of converters that belong to the single pool shared among all node's channels. In both cases a set of links without converters is also provided to forward packets that do not need conversion. Each switch fiber is equipped with a small FDL buffer to provide additional contention resolution in time domain. Looking at the architecture from left to the right, the general switch behavior can be described as follows: in the de-multiplexing phase channels are separated at the input ports and then kept separated until they will be again multiplexed at output ports. After the demultiplexing phase the first optical switch selects the proper output interface which is identified by the switch control on the basis of the packet destination address. The packet might be sent to the converters pool or not depending on the need of wavelength conversion. A second switch selects the right fiber within the interface. Before reaching the output ports and being multiplexed with the other wavelengths, the packet can be delayed by FDL queues associated to each fiber. The first optical switch stage, as presented in figure 1 and 2, is quite large,

being it ($N \cdot F \cdot M \times N \cdot F \cdot M$). To overcome this problem, this stage can be organized into parallel planes, one for each wavelength employed, thus reducing the required size of each plane to ($N \cdot F \times N \cdot F$) providing additional de-multiplexing and multiplexing functions[MR06a]. A good compromise between efficiency and feasibility is fundamental when designing such architectures. As it will be shown later, the higher the number of fibers F is, the better the switch performs. But increasing F means also increasing the number of other components as Mux/Demux and associated $FDLs$. For a matter of space and complexity this components can't be too many within a single switch so a good trade-off must be reached. As regards the wavelengths assignment to fiber at a given interface, the following different solutions can be adopted:

1. the $F \times M$ wavelengths used at the switch interface are all different;
2. the same set of M wavelengths is repeated on each of the F fiber.

In case 2 converters need to work within a narrower band compared with case 1 or even compared with the single link per interface option where all wavelengths are necessarily distinct. Consequent feasibility and cost reduction can be so achieved[MR06a].

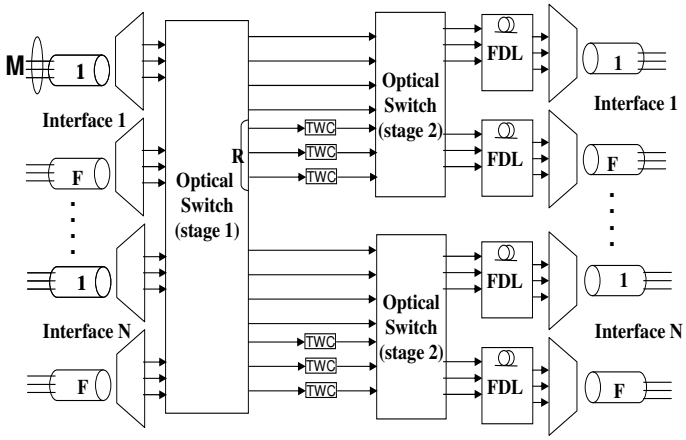

Figure 3.1: Switching node architecture with N input/output fibers, M wavelengths per fiber, a set of shared per link wavelength converters and FDLs.

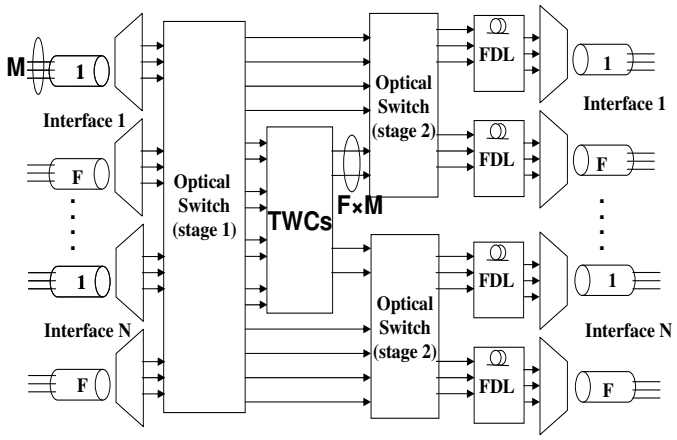

Figure 3.2: Switching node architecture with N input/output fibers, M wavelengths per fiber, a set of shared per node wavelength converters and FDLs.

3.3 Combined contention resolution algorithms

Contention resolution in the multi-fiber architecture takes place in the wavelength, time and space domains. Some of these algorithms have been already analyzed in the OPS context for single-fiber switch [CCRZ04]. The multi-fiber configuration allows to physically separate the contention domains and, possibly, to reuse the same wavelengths on the different output fibers of the same interface (see case 2 in previous section). According to this choice, two different contention resolution schemes can be applied, namely WT (Wavelength before Time) and TW (Time before Wavelength). In the former case contention resolution in the wavelength domain is first trusted and then the time domain (buffer exploitation) is considered; in the latter case contention resolution in the time domain is applied first to the whole set of F fibers of the interface according to the Wavelength and Delay Selection (WDS) algorithms [CCRZ04] and then the wavelength and the fiber are chosen accordingly.

3.3.1 WT contention resolution scheme

When this scheme is applied the switch control checks first if the optical packet can be forwarded without wavelength conversion. If more fibers match this requirement on the addressed output interface, the WDS algorithm is applied to choose the one that minimizes the gap. If, on the other hand, the arrival wavelength is congested on all the fibers of the addressed interface, the minimum gap algorithm is applied to the whole set of remaining wavelength channels of the interface, in order to identify the wavelength to convert to. Clearly this case gives priority to the wavelength conversion rather than the buffer optimization.

3.3.2 TW contention resolution scheme

This scheme considers the WDS algorithm first. When a packet arrives at the switch, the control unit checks among all the wavelengths of the output interface and chooses the one that minimizes the gap in the corresponding optical buffer. If more than a wavelength provides the same minimized gap the shortest one among them is considered. The min-gap algorithm is then applied to all wavelengths of the output interface and the buffer usage will result optimized, in relation to the algorithm applied. To this end a bigger effort is required to the converters. It will be seen later which is the trade-off between these two techniques in terms of number of converters and performance.

3.4 Analytical model for the buffer-less case

In this section an analytical model will be presented for the asynchronous multi-fiber buffer-less case. To this aim an approach based on Markov chains could be adopted, although it would have critical complexity as the number of switched channels increases. So a different approach is here proposed to achieve quite good matching with lower complexity. The model is based on the Equivalent Random Theory [Wil56][Fre80]. The contention resolution scheme adopted is the WT. The model is first validated for the shared per link architecture and afterward it is extended to the shared per node case. With the assumption of asynchronous network and variable packet length the incoming traffic is taken Poisson

and the packet size distribution as exponential. These assumptions are quite realistic as shown in previous works [IA02].The total load is equally distributed toward the output channels. For a matter of clarity all the variables included in the model will now be listed and commentated. The model will be described immediately after. First of all let's anticipate the general expression for the packet loss probability which is:

$$P_{Loss} = P_u + P_{tr} \cdot (1 - \frac{P_u}{P_{tr}}) \cdot P_{bwc} \tag{3.1}$$

where P_u is the probability of having all the output channels busy independently of the state of the converters.
P_{tr} is the probability that a packet needs a converter to be sent because its incoming wavelength is busy on the ouput interface.
P_{bwc} is the packet loss experienced by the converters.
A_0 is the average load on incoming wavelengths.
A_1 is the load on a tagged outgoing wavelength.
A_+ is the portion of traffic that comes from the set of converters after conversion to the tagged outgoing wavelength.
A_{tr} is the portion of traffic directed to the converters from a single busy wavelength.
V_{tr} is the variance of traffic A_{tr}.
A_{wc} is the total traffic that is directed to the pool of converters.
z peackedness defined as the ratio between variance and the mean of variable A_{tr}.

The process to solve the analytical problem is the following. A tagged outgoing channel is considered. This channel is loaded with an amount of traffic A_1 that results in:

$$A_1 = A_0 + A_+ \tag{3.2}$$

that is the sum of the average input load per wavelength A_0 plus a part of traffic A_+ that comes from the set of converters after conversion to the tagged wavelength [CM05]. Here, instead, the probability P_u of having all the output channels busy independently of the state of the converters can be calculated using the Erlang B-Formula with $F \cdot M$ servers loaded with $F \cdot M \cdot A_0$ as:

$$P_u = B(F \cdot M; F \cdot M \cdot A_0) \tag{3.3}$$

P_{tr} is the probability that a packet needs a converter to be sent because its incoming wavelength is busy. If there are wavelength channels available at the output ports the packet looks for a different wavelength and uses a converter. If there are no wavelength channels available the packet is discarded. P_{tr} is calculated as the joint probability that the F wavelengths (one on each fiber) of the same color of the tagged packet are busy and there is at least a wavelength free at the output stage.

$$P_{tr} = (1 - P_u) \cdot B(F; F \cdot A_1) \tag{3.4}$$

A_1 in this case is assumed Poisson and as long as A_+ is a small fraction of A_0 this assumption is quite tolerable[CM05]. A_{tr} is the portion of traffic directed to the converters from a single wavelength and is expressed as:

$$A_{tr} = A_0 \cdot P_{tr} \cdot (1 - \frac{P_u}{P_{tr}}) \tag{3.5}$$

where the term $(1 - \frac{P_u}{P_{tr}})$ takes into account the fraction of overflow traffic that does not incur in output overbooking and that is already taken into account by P_u. The set of converters is loaded by the overflow traffic concerning all output interfaces and is calculated as the total traffic A_{wc} that is directed to the R converters, easily deduced from the expression 3.5 of A_{tr}:

$$A_{wc} = M \cdot F \cdot A_{tr} \tag{3.6}$$

The traffic A_{wc} is not exponential [MR06a] and has been characterized by the Equivalent Random Theory[Wil56][Fre80]. This theory allows to use the Erlang B-Formula for non-Poisson traffics if they are normalized to the peackedness z. This parameter is calculated as the ratio between the variance and the mean value of A_{tr} (see formula (3.5) and (3.8)). It is an index of the variability of the traffic with comparison with the Poisson distribution for which it results $z = 1$. The 'peaky' traffic that loads the converters has a greater variability than Poisson traffic and so $z > 1$. The variance of the traffic A_{tr} is evaluated through the formula [Wil56]:

$$V_{tr} = A_{tr} \cdot (1 - A_{tr} + \frac{F \cdot M \cdot A_1}{F \cdot M - F \cdot M \cdot A_1 + A_{tr} \cdot F \cdot M + 1}) \tag{3.7}$$

taken from the Equivalent Random Theory and applied to the multi-fiber scheme. The peackedness z can be then expressed as:

$$z = \frac{V_{tr}}{A_{tr}} \tag{3.8}$$

The packet loss probability P_{bwc} experienced by the converters can be then expressed as [Fre80]:

$$P_{bwc} = B(\frac{R}{z}; \frac{A_{wc}}{z}) \tag{3.9}$$

Finally the overall packet loss probability is formulated as:

$$P_{Loss} = P_u + P_{tr} \cdot (1 - \frac{P_u}{P_{tr}}) \cdot P_{bwc} \tag{3.10}$$

where, again, $(1 - \frac{P_u}{P_{tr}})$ takes into account that part of traffic that does not occur in output contention. The extension to the shared per node case is quite straightforward. The same approach is indeed adopted. The only changes affect the expression of the variance of the traffic A_{tr} and of the total traffic A_{wc} directed to the converters that become:

$$A_{wc}^{node} = M \cdot F \cdot N \cdot A_{tr} \tag{3.11}$$

and

$$V_{tr}^{node} = A_{tr} \cdot (1 - A_{tr} + \frac{F \cdot M \cdot N \cdot A_1}{F \cdot M \cdot N - F \cdot M \cdot N \cdot A_1 + A_{tr} \cdot F \cdot M \cdot N + 1}) \tag{3.12}$$

being the pool of converters in this case shared among all $N \times F \times M$ channels. The validation of the model through comparison with simulation results will be shown in the results section.

3.5 Quality of service in multi-fiber architectures

In this section some techniques to support quality of service for the aforementioned node architectures are presented. As it will be shown, by reserving node resources as converters and/or wavelengths, it is possible

to achieve good differentiation between distinct classes of service. These techniques are independent of the converters sharing policy and can be applied to both architectures considered in this work. Let's assume that two classes with different priorities have to be served. They will be referred as High Priority Class(HPC) and Low Priority Class(LPC). As the converters reservation is concerned, let's now assume that a threshold T_1 is considered for the set of C converters (with $T_1 < C$). Upon each arrival the control unit checks out the class that the packet belongs to and the number S of free converters. This number is then compared to T_1. If $S \leq T_1$ the packet is served by one of the converters, if necessary, only if it belongs to HPC. In this case LPC packets cannot be converted. On the other way, if $S > T_1$ there is no restriction for the LPC packets and they can be converted by any TWC. The effectiveness of this method depends of course on T_1, that determines the percentage of converters used by the HPC but also on the total number of wavelength converters and on the percentage of HPC packets with respect of the total traffic. The same principle can be applied to the wavelength domain, meaning that the output channels with their associated $FDLs$ can be reserved by the HPC. When a packet is processed, the control unit verifies how many channels are free in the corresponding buffer. If this number is less than or equal to the threshold T_2 only HPC packets are served whereas the LPC packets are rejected. Obviously T_2 must be less than the total number of channels per interface $F \times M$. Both T_1 and T_2 can be greater than zero and can be either the same or not. So a joint reservation of the converters and of the output channels can be implemented to optimize the design of the quality of service of the two classes. Next section will show some numerical examples as results of the application of these techniques.

3.6 Numerical results

3.6.1 General performance

In this section simulation results and model validation are discussed. An ad hoc simulator written in C language has been developed and run. Simulation set up is as follows: a 4×4 switch is considered. The total number of output channels $F \times M$ for each interface is kept constant

at 32. Six different configurations for the multi-fiber architectures are taken into account depending on the number of wavelengths multiplexed on the fibers. These six configurations can be labeled with the couple (F, M) equal to $(1, 32), (2, 16), (4, 8), (8, 4), (16, 2)$ and $(32, 1)$. The first case represents the single-fiber case and the last configuration is the other extreme case where each fiber carries only one wavelength. The incoming traffic is Poisson distribution and the packet length is exponential with a mean of 500 bytes. The bit rate of operation is 2.5 Gbit/s. The average load per wavelength is equal to 0.8. Unless stated the buffer granularity D will be taken equal to the average packet length. Shared per link architecture is first analyzed. In figure 3.3 the packet loss probability (indicated with PLP in the following) is plotted as a function of the number of converters for different buffer capacities. All six configurations for the couple (F, M) are taken into account. The buffer parameter B indicates the number of FDLs in the optical buffer. It varies in the four graphs from 1(a) to 8(d) with intermediate values of 2(b) and 4(c). B equal to 1 is the buffer-less case when buffer can provide only cut through (zero queue length). In these first figures the wavelength repetition option and the WT algorithm for contention resolution are adopted. The multi-fiber scheme improves the performance when the number of fibers increases for limited converters availability. This benefit is due to the isolation of contention resolution in different space domains (the distinct fibers) that provides the presence of more than an instance of the same wavelength on different fibers of the same interface. The configuration for (F,M)=(32,1) is independent of the number of converters since the wavelengths are all the same. Performance tends to saturate, as known, with a sufficiently high number of converters and the less is M the faster is the saturation.

3.6.2 Contention resolution algorithms

In figure 3.4 the WT and TW techniques are compared for shared per link converters by considering the repetition of the same set of wavelengths on different fiber. B is equal to 4. Configurations with $(F, M) = (16, 2)$ and $(8, 4)$ are plotted. A different behavior of the two techniques is evident with distinct region of convenience. TW performance improves much quicker with the number of converters but requires more TWC at

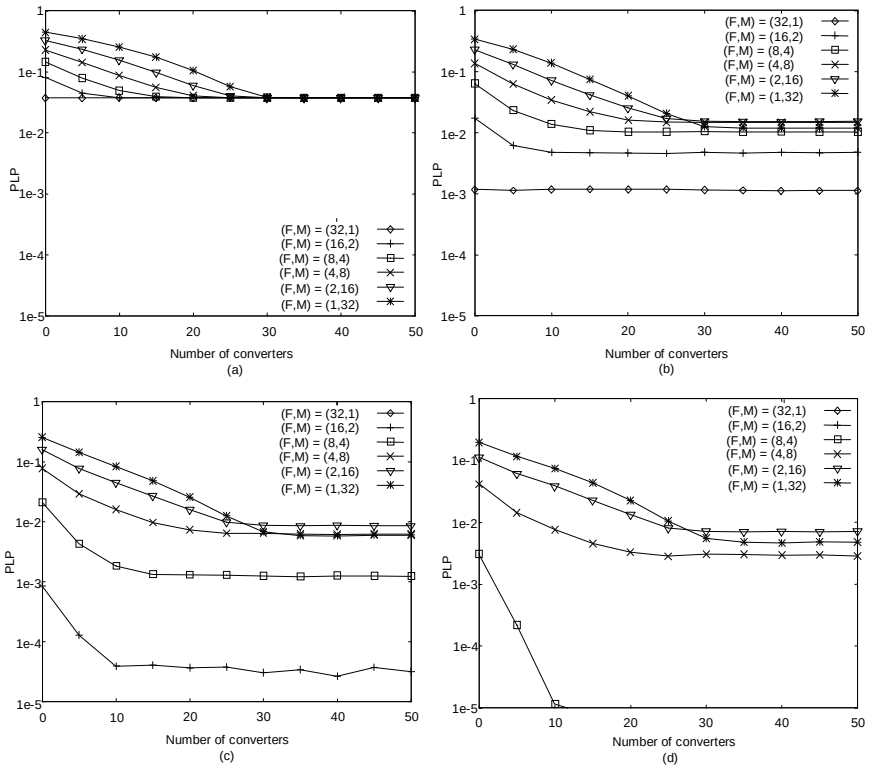

Figure 3.3: Packet loss probability as a function of the number of converters per output interface, shared per link architecture, WT algorithm, wavelength repetition and $B = 1(a), 2(b), 4(c), 8(d)$.

the crossing point. This is because with only few converters available the blocking is more concentrated on the set of converters so it is more important to optimize their use. As R increases and, in particular, it becomes much greater than the number of channels $F \times M$, the set of converters becomes a lossless system, while contention mainly occurs in the time domain so the TW technique is more suitable to reduce blocking by optimizing the buffer usage. Figure 3.5 shows that the multi-fiber scheme does not have any further advantage when every fiber carries a different set of wavelengths and no wavelength repetition is applied. For this case every packet that finds its wavelength busy must convert and the set of TWCs results to be more loaded than before. As a consequence the performance obtained by varying F and M remains the same of the

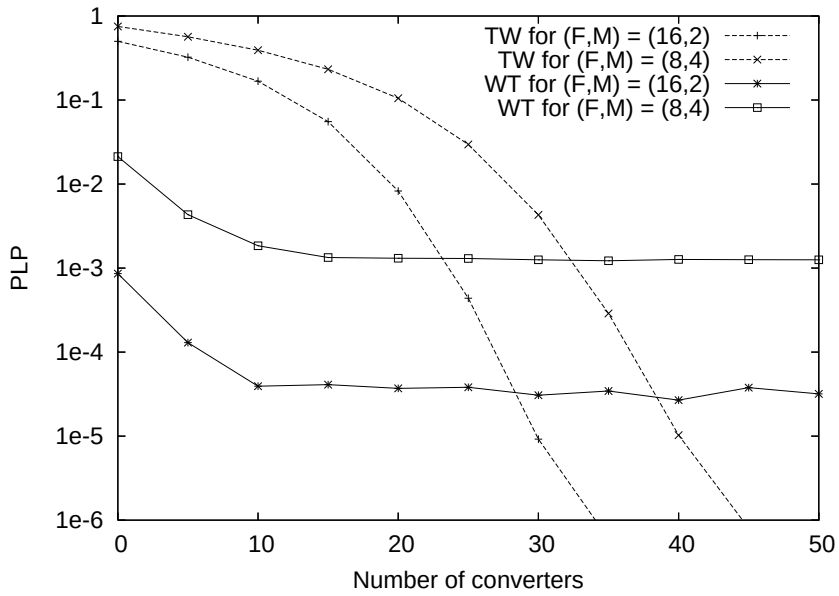

Figure 3.4: Packet loss probability as a function of the number of the converters, comparison between TW and WT for shared per link architecture, B=4, (F,M)=(16,2),(8,4), wavelength repetition.

single-fiber scheme which, as seen before, is less advantageous than the multi-fiber scheme. In figure 3.5 both TW and WT techniques show a trend similar to the previous figure. As said the three curves representing the three different configurations overlap each other. It turns out that the scheme without repetition is not appealing either in terms of loss probability or in terms of conversion.

3.6.3 Architecture comparison

Performance for the shared per node structure is plotted in figure 3.6 and a comparison with the shared per link results (figure 3.3) is shown in figure 3.7. To fairly compare the two architectures the total number of converters is the same. For the shared per link option this number is equal to $N \cdot R$ whereas for the shared per node option it is equal to C as indicated in section 3.2. In figure 3.7 the number of converters in x axis is meant to be the average number of converters per interface which

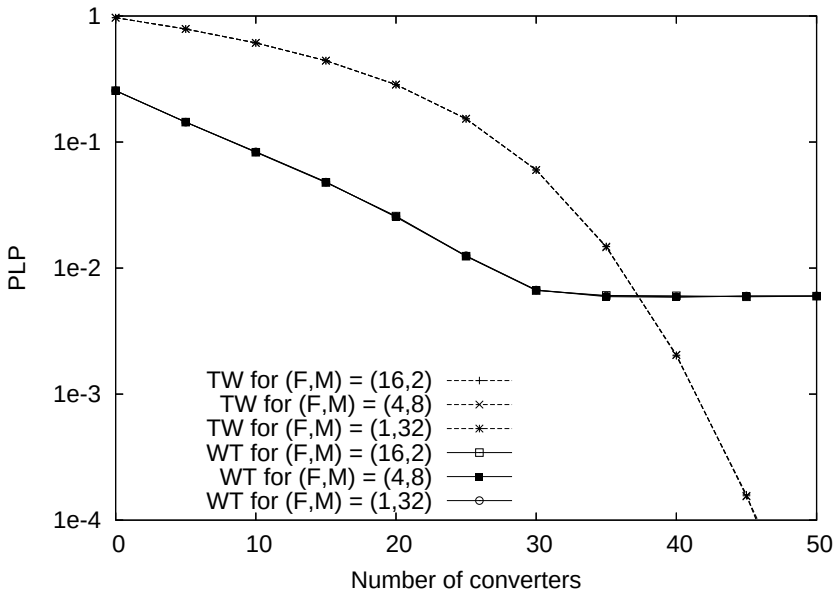

Figure 3.5: Packet loss probability as a function of the number of converters, comparison between TW and WT techniques for shared per link architecture, B=4, no wavelength repetition.

corresponds to $\frac{C}{N}$ for the shared per node as an indication of the average number of converters dedicated to each interface. Only the case with $(F, M) = (8, 4)$ is plotted when $B = 4$ and $B = 8$. These results show that the performance saturates earlier when the converters are shared per node bringing a further benefit in comparison with the shared per link case. So the more the converters are shared, the best they work and as long as the switch complexity is acceptable this option should be adopted.

3.6.4 Varying buffer granularity

An investigation of the switch behavior as a function of the delay unit D is now shown. So far this parameter was taken equal to the average packet length. By varying it, it was observed that this parameter can significantly influence the performance depending on which configuration is taken into account. In figure 3.8 the packet loss probability is plotted

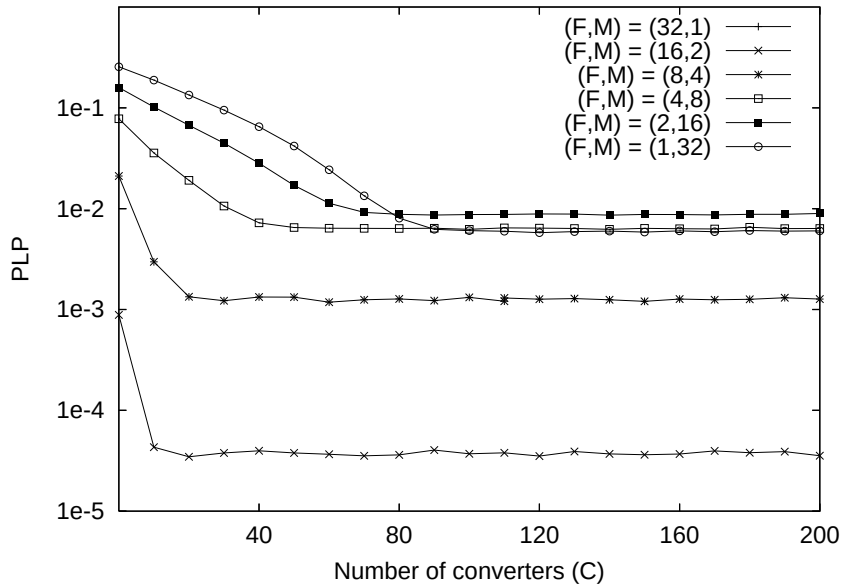

Figure 3.6: Packet loss probability as a function of the number of converters per output interface, with shared per node architecture, B=4, WT technique and wavelength repetition.

for all possible configurations as a function of the delay unit normalized to the average packet length which will be indicated as D_n. Again the shared per link policy is used with a fix number of converters equal to $R = 30$. Buffer length is $B = 4$ and WT technique with wavelength repetition is applied. Each curve has got its minimum corresponding to a different value of D_n. For instance $(F, M) = (16, 2)$ and $(F, M) = (8, 4)$ reach their minimum for $D_n = 1$ and $D_n = 0.7$, respectively. This could be a problem in the switch design phase since the configuration must be known a priori in order to optimize the buffer space. Another investigation on the dependency of the buffer granularity is illustrated in figure 3.9 and 3.10 for the two configurations $(F, M) = (16, 2)$ and $(F, M) = (8, 4)$ respectively. Each curve is obtained by varying the number of available converters ranging from 0 up to 30.

Another difference can be found between the two configurations regarding the number of converters needed to obtain the best performance. For $(F, M) = (16, 2)$ the minimum is reached with about 10 wavelength

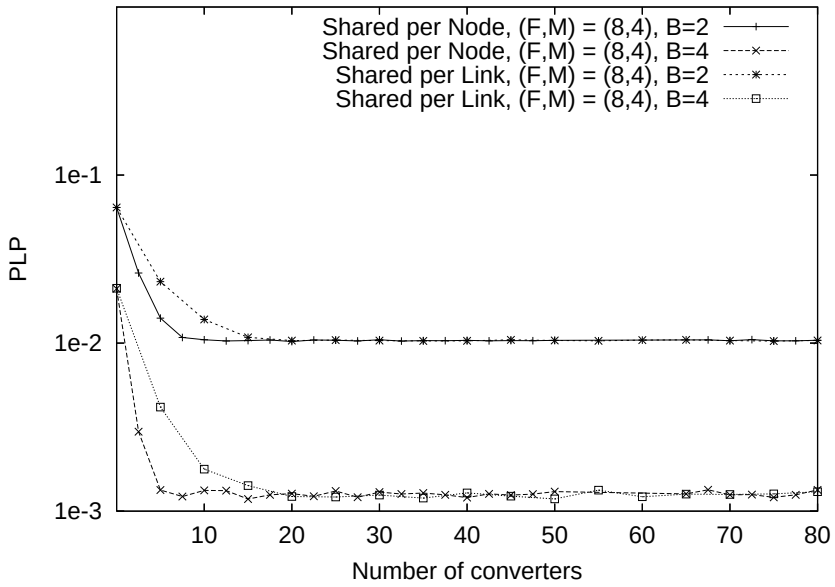

Figure 3.7: Packet loss probability as a function of the number of converters per output interface, comparison between shared per link and per node architectures, B=(4,8), (F,M)=(8,4), WT technique and wavelength repetition.

converters and performance does not improve by adding more converters. $(F, M) = (16, 2)$ needs about 20 converters to get to its minimum. So, in general, configurations with less wavelengths per fiber reach the best performance with less converters.

3.6.5 Performance with quality of service

Last simulation results concern with the quality of service obtained as described in section 3.5. The switch architecture considered is the shared per node one (figure 2). The fraction of HPC is 25 percent. In figure 3.11 the packet loss probability is shown for two classes (HPC and LPC) and for two configurations $(F, M) = (16, 2)$ and $(4, 8)$ as a function of the converters percentage reserved to HPC class. In this case no wavelength is reserved. The total number of TWCs is 20. Wavelength repetition and WT technique are applied. Service differentiation by incrementing

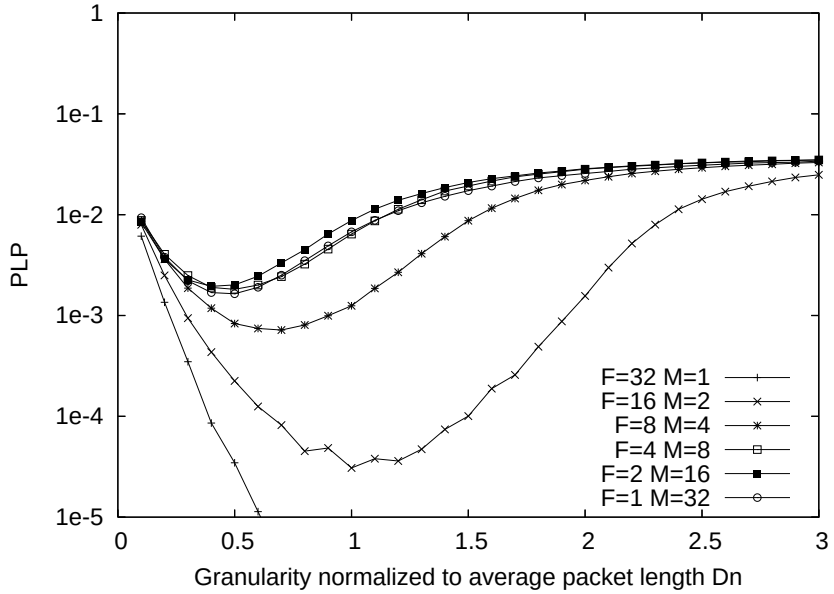

Figure 3.8: Packet loss probability as a function of the delay unit normalized to the average packet length, shared per link architecture, R=30, B=4, WT technique and wavelength repetition.

the percentage of TWC for HPC class is achieved only for those configurations that are sensitive to the use of the converters. When very few wavelengths (i.e. $M = 2$) are used on each fiber and the wavelength repetition is applied, the converters load is very low and the related reservation scheme does not influence the performance of the two classes that indeed remain the same. On the other hand when many wavelengths are used on a fiber, the converters reservation is efficient to obtain service differentiation between the two classes as it can be seen for the case $(F, M) = (4, 8)$. In figure 3.12 the percentage of wavelength converters is fixed at 25 and their total number is on the x axis. $(F, M) = (2, 16)$ and $(4, 8)$ are considered being more sensitive to the wavelength reservation. This graph shows how the loss probability behaves varying the total number of converters from 0 to 100. The best service differentiation is given for a specific number of total converters after which the performance of the two classes converges to the same value. With an infinite number of converters there is no more service differentiation because LPC packets

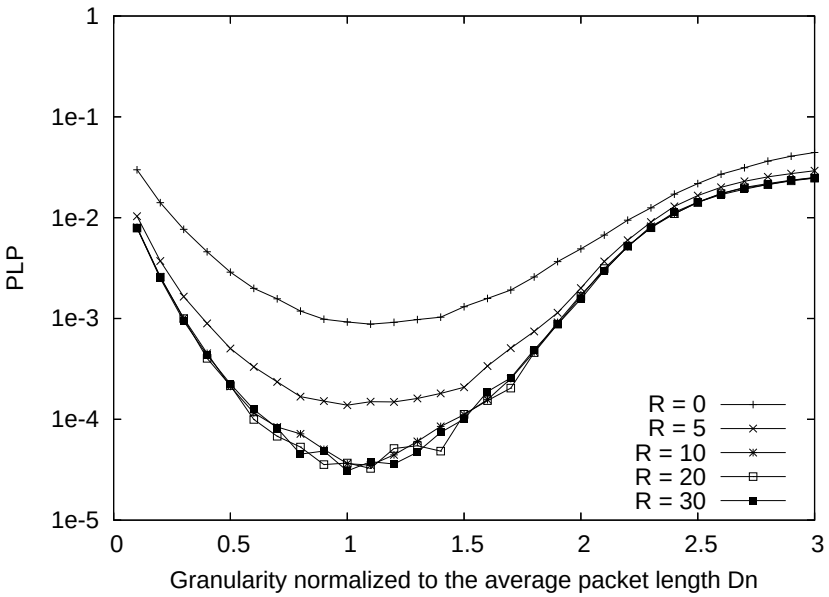

Figure 3.9: Packet loss probability as a function of the delay unit normalized to the average packet length, shared per link architecture, B=4, (F,M)=(16,2), WT technique and wavelength repetition.

can always be converted and sent toward the buffer stage where there is no class differentiation.

Figure 3.13 refers to the case of joint reservation of both TWC and wavelength channels. Packet loss probability is still plotted as a function of the total number of converters. Percentage of converters is fixed at 25. Wavelength percentage reservation is equal to 5 and 25. By comparing with figure 3.12, it can be seen that due to wavelength reservation the service differentiation is kept for an infinite number of converter as well. The behavior of the previous figure is avoided and the HPC loss probability is always distinct from the LPC corresponding curve.

3.6.6 Model validation of the buffer-less case

As model validation is concerned, figure 3.14 refers to an average load per wavelength equal to 0.5 and figure 3.15 to 0.8 for the shared per link configuration.

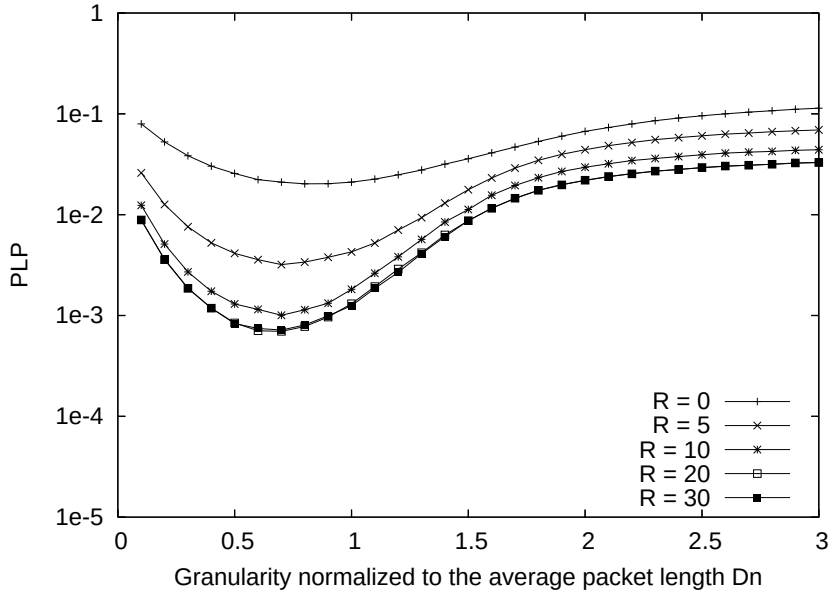

Figure 3.10: Packet loss probability as a function of the delay unit normalized to the average packet length, shared per link architecture, B=4, (F,M)=(8,4), WT technique and wavelength repetition.

Validation for the shared per node architecture is plotted in figure 3.16 and 3.17 again for average load per wavelength equal to 0.5 and 0.8 respectively.

In general the model well matches with the simulation results. The little difference that can be seen especially for the single-fibre case is due to the approximate evaluation of the variance of the traffic offered to the wavelength converters.

3.7 Final considerations

In this chapter contention resolution schemes for optical packet switching are considered and applied to architectures with shared wavelength converters and multi-fibre interfaces. The proposed architectures are able, depending on the sharing scheme applied, to reduce the bandwidth of the tunable wavelength converters employed and the overall switch cost.

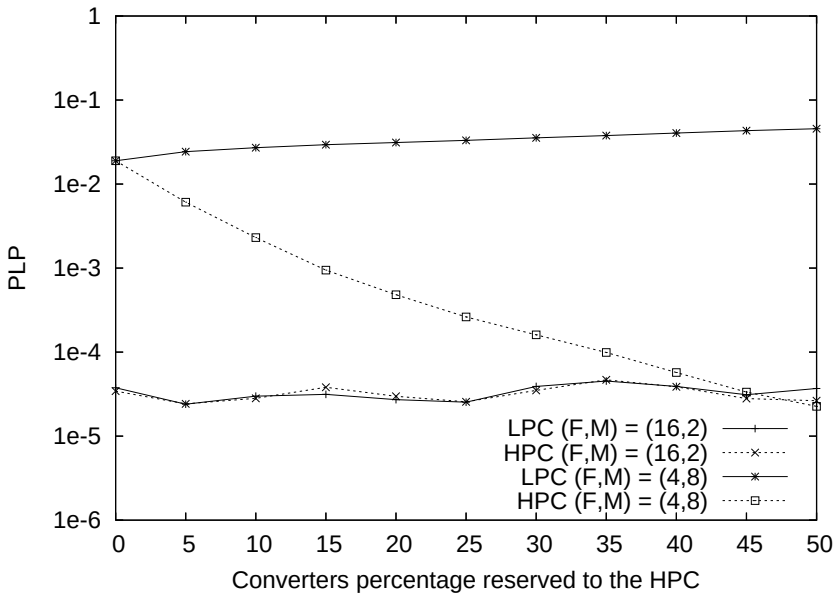

Figure 3.11: Packet loss probability as a function of the percentage of number of converters reserved to HPC, shared per node architecture, (F,M)=(4,8),(16,2), B=4, C=20, 25% of HPC packets, wavelength repetition and WT technique.

Performance is thoroughly evaluated by simulation and by an original analytical model in the special case of buffer-less configuration. The effectiveness of wavelength and delay selection algorithms is discussed and techniques to support Quality of Service by means of wavelength converters reservation are proposed. In conclusion, the multi-fiber scheme allows enhancing switch feasibility through wavelength converters bandwidth reduction. At the same time the price to pay in terms of space matrix and buffer complexity must be carefully evaluated. To this end the insight of matrix implementation is needed: this task is under study and left for future contributions.

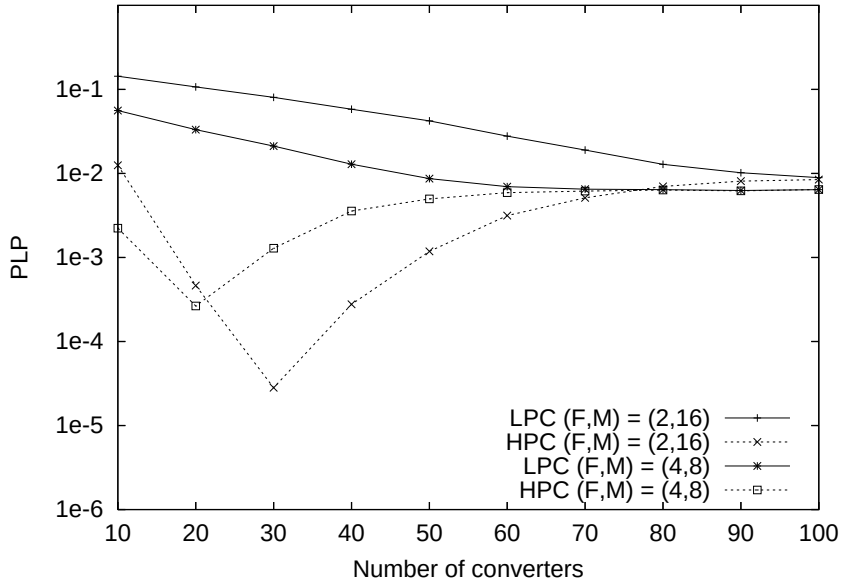

Figure 3.12: Packet loss probability as a function of nthe umber of converters, shared per node architecture, (F,M)=(2,16),(4,8), B=4, 25% of HPC packets, 25% of converters reserved, wavelength repetition and WT technique.

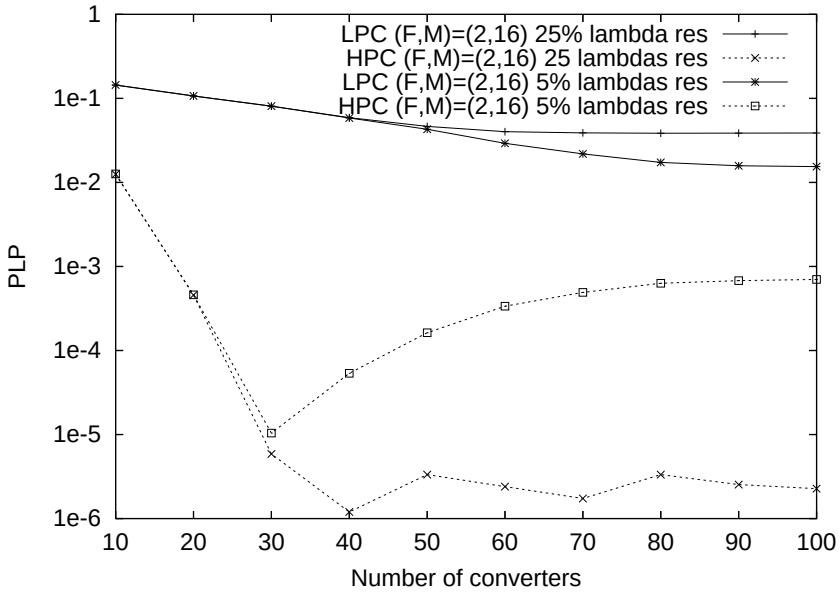

Figure 3.13: Packet loss probability as a function of the number of converters, shared per node architecture, (F,M)=(2,16),(4,8), B=4, 25% of HPC packets, 25% of converters reserved, wavelength repetition and WT technique.

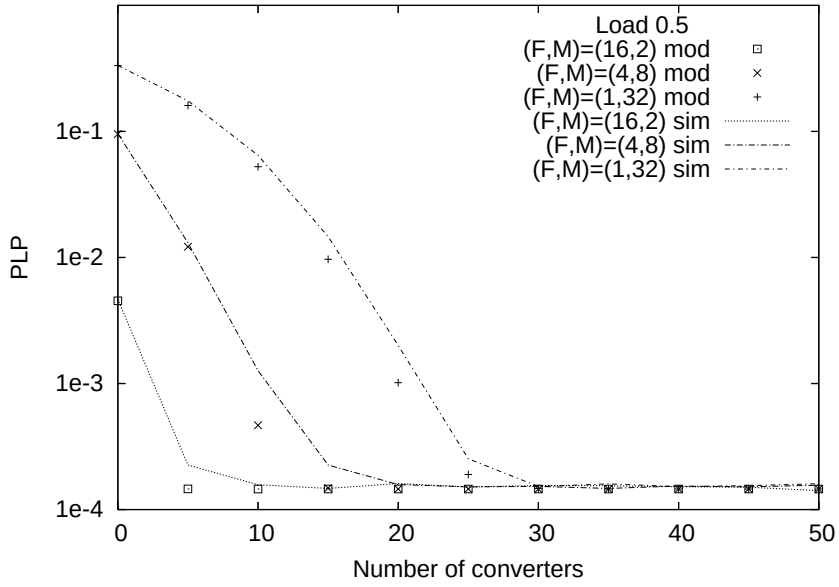

Figure 3.14: Packet loss probability as a function of the number of converters, comparison between simulation and analytical model, shared per link architecture, (F,M)=(2,16),(4,8),(16,2), B=1, wavelength repetition and WT technique, average load per wavelength 0.5.

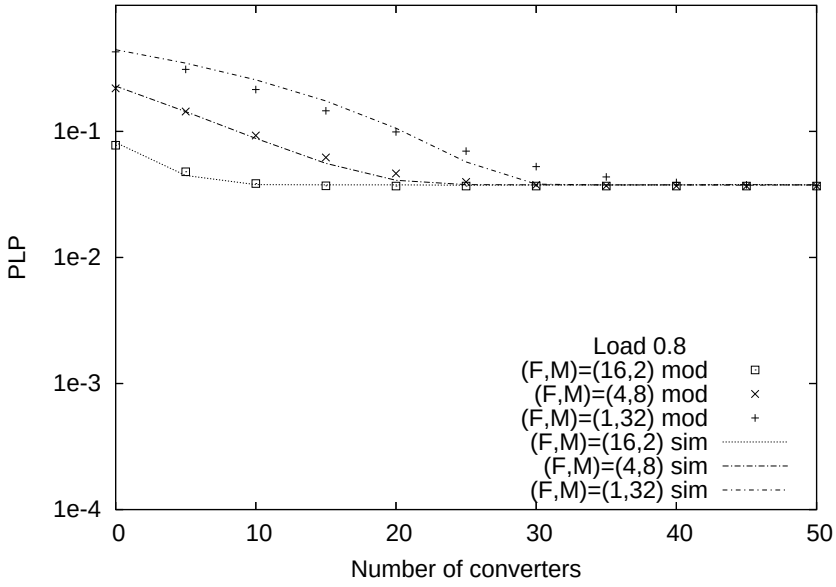

Figure 3.15: Packet loss probability as a function of the number of converters, comparison between simulation and analytical model, shared per link architecture, (F,M)=(2,16),(4,8),(16,2), B=1, wavelength repetition and WT technique, average load per wavelength 0.8.

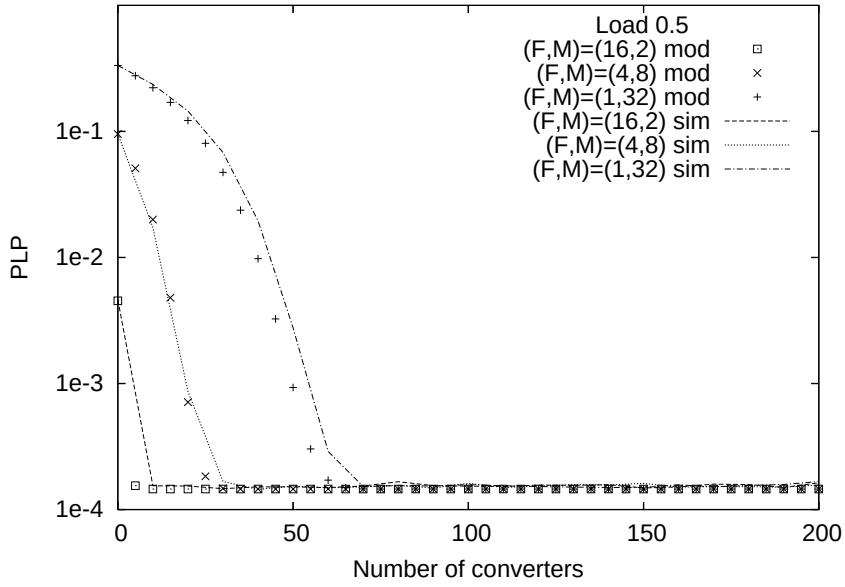

Figure 3.16: Packet loss probability as a function of the number of converters, comparison between simulation and analytical model, shared per node architecture, (F,M)=(2,16),(4,8),(16,2), B=1, wavelength repetition and WT technique, average load per wavelength 0.5.

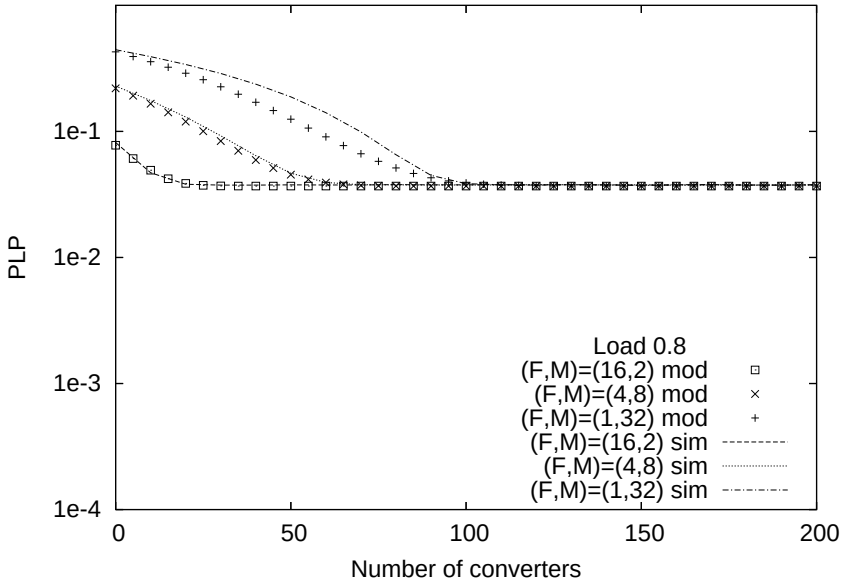

Figure 3.17: Packet loss probability as a function of the number of converters, comparison between simulation and analytical model, shared per node architecture, (F,M)=(2,16),(4,8),(16,2), B=1, wavelength repetition and WT technique, average load per wavelength 0.8.

Chapter 4

Adaptive routing in WDM optical packet-switched networks

4.1 Introduction

In this chapter a core optical packet switching network is first considered to investigate the effectiveness of statistical multiplexing of packets on different sets of wavelength paths. Several routing algorithms are proposed and compared that make use of wavelength multiplexing at various levels, ranging from the single fiber, to a set of fibers including shortest and even longer network paths. The aim is to design adaptive routing that outperforms conventional shortest path routing by exploiting the wavelength domain. Dynamic management consists in using available network resources by exploiting Wavelength Division Multiplexing (WDM) with full wavelength conversion in relation to possible relevant congestion arising in the Fibre Delay Line (FDL) buffer, which is used to solve contentions between optical packets. Secondly the chapter considers the problem of service differentiation in an optical packet switched backbone. A QoS routing approach based on different routing and congestion management strategies for different classes of service is analyzed. Congestion resolution is achieved by using the wavelength and time domain and QoS differentiation in the single node is achieved by resource reservation in the wavelength domain. This is combined with the previ-

ous alternate routing at the network level. In the chapter the proposed strategy show to guarantee very good performance to the high priority traffic with very limited impact on low priority traffic. Last part is dedicated to the role that adaptive routing can have on one of the main requirements for an OPS network which is its reliability, that is the capability to recover from network device or link failure. Dynamic management of packet routing is studied and applied with the aim of keeping packet loss as low as possible in the presence of single link failures. A key parameter is the delay time to detect the failure itself and call the recovery procedure that must be chosen to optimize resource utilization. Dynamic recovery is done by using shared alternative paths classified by their cost expressed in number of hops to reach the destination. Two different techniques are considered, namely *Link Protection* (LP) and *Dynamic Routing* (DR), and compared. A simple analytical model for packet loss probability on the failed link as a function of recovery delay is presented and evaluated by comparison to simulation results. An example of network design is finally described to cope with loss probability constraints in the presence of failure.

4.2 Network scenario for adaptive routing

The reference network scenario is represented by an optical packet switched network that acts as a backbone for interconnection of peripheral networking areas [DDC+03]. Each physical link between optical packet routers is assumed to assure bi-directional connections through a couple of fibres each supporting Dense Wavelength Division Multiplexing. The Optical Packet Routers (OPR) employ optical switching matrix with a classical architecture based on an all-optical space switch interconnecting the input/output DWDM fibers. Assuming for simplicity a square switch:

- M is the number of input and output fibers;
- W is the number of wavelengths on each in/out fibre.

This leads to a $(M \times W) \times (M \times W)$ switching fabric, where the $(M \times w)$ inlets/outlets are grouped in M bundles of W wavelengths. The M fibers correspond to the matrix next hop destinations, while the

W wavelengths are equivalent in a routing perspective. As a consequence once the forwarding component of the OPR has decided the output fiber to satisfy the routing constraints, the output wavelength can be freely used for congestion resolution in order to optimize the switch performance. Furthermore, congestion resolution can be implemented in the time domain using a delay buffer [HCA98]. A similar delay buffer can be implemented in several ways, with feed-forward or feedback architecture, and with simple fiber delay lines (FDLs) or other hardware architectures. Nonetheless, the basic logical behavior is the same: a packet is delayed by a certain amount of time by sending it to the proper hardware resource that can host up to k packets at a time. For the sake of simplicity, in this chapter the possibility to have more than one access to the optical buffer per packet is not considered, as it would be possible with a feedback architecture and re-circulation of packets. Congestion resolution is implemented by means of scheduling algorithms that aims at balancing the load on the w wavelengths, minimizing loss of packets. Once the forwarding algorithm has determined the next hop fiber, the switch control logic schedules the packet for a given wavelength or for the optical buffer if it has to be delayed. Ideally, in the buffer a packet should be delayed by an amount of time that is just enough to have it ready when the transmission wavelength will be available. Unfortunately the buffer has a limited time resolution and it may well be that the delays available are just an approximation of the required one. As a consequence gaps between packets may arise, reducing the available bandwidth at the output and therefore being detrimental for performance. In the case of asynchronous variable length packets several scheduling algorithm for congestion resolution have been studied, at various level of complexity. The *Void Filling algorithm* proposed in [TYC+00] keeps track of the whole spectrum of packets scheduled and, when a new packet has to be delayed, tries at first to fit it into one of the existing gaps, if any. The *Minimum Gap (MinGap)* algorithm proposed in [CCC01] does not consider the gaps and just keeps track of the times the last scheduled packet will end transmission on each wavelength. A new packet is scheduled on the wavelength where the gap is minimized (i.e. requiring the delay that is best approximated by the delays available). The *Minimum Length* (MinLen) algorithms does not consider the gaps but focus on the queue length and aims at reducing the overall latency by queuing a new packet

into the shortest queue. The Void Filling algorithm is better performing but requires a lot of processing effort, the MinGap and MinLen latter is less complex but also less performing, therefore the choice is a matter of trade-off. The use of the Minimum Gap algorithm is the assumption on the chapter.

4.3 Algorithms for adaptive routing

Generally speaking, routing algorithms can be *static* or *adaptive (dynamic)* [CGK99][BSS95]. The former algorithms define the routing tables and the consequent forwarding once and for all, whereas the latter route traffic by exploiting information regarding the state of the network. The adaptive algorithms may be further specialized depending on the number and on the costs of the paths they will consider to take the forwarding decision. The cardinality of the set of paths considered per source/destination pair may range from just the shortest-paths up to a pool counting all the available paths, shortest or not [RM02]. This section focuses on adaptive routing algorithms for an optical packet switched network with WDM links. The routing algorithms are adaptive in the sense that they will consider an alternative to the shortest path when congestion arises. Moreover they will use the wavelength domain to solve congestion in the network and therefore minimize the packet loss probability. In particular the proposed algorithm have been designed to combine the flexibility of alternate routing with the power of multiplexing packets over a large set of wavelength. The starting assumptions for this work are that the algorithms:

- start from the knowledge of the network topology and of the traffic matrix.
- are adaptive and distributed, i.e. depending on the network state information and applied at each node at the time a packet needs to be routed.

The former assumption is justified by the fact that, in a scenario of a wide-area optical network providing geographical connectivity among major European cities, the network topology is strictly constrained by

political, economical and logistical factors. Moreover the traffic matrix can be assumed to be defined a priori in relation to the user requirements. The latter is typical of existing packet switched network like the Internet and has proved to be a flexible and fault proof solution.

The proposed algorithms start from a set of known data that are obtained by the simple knowledge of the network topology in a few correlated steps.

- Calculate the length, measured in number of hops, of the set of existing paths per source-destination pair.
- Count and sort the set of existing paths according to their cost (hop count).
- Select the default set (used at the first attempt to route a packet) and the alternate sets of routing paths (used as an alternatives when the default set is congested) according to the different routing strategies. Different algorithms will use different alternate subsets, both the number and the cost of the paths included in the sub-sets characterize each algorithm.
- At each node create a routing table where is indicated the output port (fiber) per each path to a given destination.

Starting from the routing table and the default and alternate sub-sets of routes, defined according to the previous steps, the routing algorithms perform, on a per packet basis, two main steps:

- Search for a wavelength of the fibers corresponding to the routing paths in the default sub-set to which the packet can be assigned for transmission, either immediately or after being delayed of a proper amount in the optical buffer: this will be called the wavelength selection phase of the algorithm.
- If no such wavelength is available and the wavelength selection is unsuccessful.
- Search the alternate sets (if any) of paths and repeat the wavelength selection;

- If an available wavelength is not found neither in the default set nor in the alternate sets of routes the packet is dropped.

Starting from these concepts the chapter aims at investigating the benefit that is obtainable in term of network performance by using both alternate routing and wavelength multiplexing in a combined way. With wavelength multiplexing or sharing the used strategy is addressed by the forwarding algorithm to choose the wavelength to transmit the packet among the set of wavelengths on the next hop identified by the routing algorithm. Two wavelength strategies have been considered:

1. *Partial sharing*: wavelength multiplexing is performed within a single fiber, corresponding to a path between source and destination that is identified by the routing algorithm beforehand.

2. *Complete sharing*: wavelength multiplexing is extended to all fibers (paths) of a given set of routes, therefore both the default and alternate set of routes may include several choices.

In terms of alternate routing four possible alternative will be taken into account:

1. *NA (No Alternative) or Single Link Choice (SLC)*: in which the routing algorithm just uses a default set that includes one shortest paths between peers.

2. *SA (Single Alternative)*: in which a default and an alternate set of routes are considered, both including shortest paths; in this case the alternate set exists if and only if several shortest paths of the same length exist between peers.

3. *MA (Multiple Alternative)*: in which a default and several alternate sets of routes are considered. In the algorithms described in the following will be limited to the two sets, the former including the other shortest paths not included in the default set, the latter including the paths of length equal to the shortest plus 1 hop.

4. *All Link Choice (ALC)*, in which several paths of increasing length are included in the default set and the forwarding algorithm choose between them according to a strategy that is not driven by wavelength multiplexing and not by routing issues.

By differently coupling the previously defined wavelength sharing techniques (PS and CS), with the link selection techniques (SA and MA), some different routing algorithms can be defined, which are expected to affect in a different way the network performance. In the remaining part of the section the proposed algorithms for adaptive routing will be described in depth, by presenting first those adopting the PS technique and then those based on a CS approach.

4.3.1 PS algorithms

In this section some routing algorithms that are based on the Partial Sharing principle previously explained are described. The all consider a default set of routes including only one shortest path between source and destination. The alternate sets are defined according to the algorithm and may be:

1. an empty set.
2. a set including paths of length equal to the shortest if available but not included in the default set.
3. a set including paths equal to the shortest plus 1 hop.

- *SL (Single Link choice).* It's the easiest routing algorithm. This algorithm is put in this section even if there is no real alternate routing but just partial sharing of wavelengths on the shortest path is used. In fact it considers only one routing table that implements the shortest path without considering any other alternative. Therefore this algorithm uses only the default set of routes that includes the shortest path calculated with standard algorithms such as Djikstra. When a packet arrives to a node, the next link is fixed by the routing table and the packet is forwarded to the wavelength chosen with the Minimum Gap algorithm.
- $PS-SA$ *(Partial Sharing with Single Alternative).* This algorithm defines the default set as previously explained but also defines an alternate set including a second shortest path route if any available. It applies partial sharing to the shortest path in the default set. When congestion arises on this fiber, meaning that a packet

loss would occur, before dropping the packet it seeks another wavelength on the second sub-set, including an alternate shortest path if available.

- *$PS - MA$ (Partial Sharing with Multiple Alternative).* This algorithm considers a default set with the shortest path route and two alternate sets including either other shortest path routes if available and/or paths 1 hop longer than the shortest. As in PS-SA the algorithm reacts to congestion on the default route and therefore attempts to forward the packet to the paths in the first and seconds alternate set. It first searches the alternate set including the other existing shortest paths and, in case this is empty or busy as well, will test the alternate set including the longer paths. In both cases the algorithm will choose one of the free wavelengths if any available or that requiring the shortest delay.

4.3.2 CS Algorithms

The algorithms considered in this section differ from the previous ones because mainly they do not have a default set of routes including just one shortest path, but including a full set of paths (shortest or not) in such a way that the choice of the wavelength is the dominant criteria used to decide how to send the packets.

- *$CS - SA$ (Complete Sharing with Single Alternative).* As the previous two cases this algorithm considers only shortest paths but it puts all of them in the default set, therefore it checks all the wavelengths available on shortest paths, without choosing a default one and then waiting for congestion to arise. In practice when a packet arrives to a node this algorithm looks for the best wavelength available among all of those that belong to all shortest paths. If two wavelengths belonging to different links provide the same delay queue a random choice is made.

- *$CS - MA$ (Complete Sharing with Multiple Alternative).* For this algorithm if there is a second shortest path available then the default set is composed by the two shortest paths (alternate set is null) otherwise the default set includes the first shortest path and all the

longer paths (with still no alternate set). This algorithm compares the performance of the first shortest path with the second shortest alternative if available and selects that one that provides the shortest delay. If there is no shortest alternative it looks for those higher cost paths and compares their performances with the first choice. Again if more than one wavelength belonging to different paths provide the same minimum delay a random choice is made.

- *$CS - AL$ (Complete Sharing with All-Links considered).* It's the more dynamic algorithm and requires the most complex control logic. The default set is composed by all paths available, shortest or not. In practice a packet is forwarded on the wavelength that provide the best delay among all of those that belong to all paths available. If more than one link provides the same delay again a random choice is made.

4.3.3 Numerical results

The reference network topology adopted for numerical evaluations is a simplification of the European Optical Network (EON) considered in the COST 266 project [Cos03]. The network is assumed to connect a set of European cities, with the meshed topology shown in figure 4.1. The network consists of 15 nodes interconnected by 24 optical links at 2.5 Gbit/sec. The links are bi-directional, i.e. are assumed to be characterized by a pair of unidirectional fibers, on which packet-oriented traffic is sent by optical packet routers supporting wavelength routing capabilities and considered to be both source and collector of traffic in the optical network.

In this section results that have been obtained by simulating the network with an ad hoc simulator are shown. In the first simulations when a node generates a packet, its destination is chosen randomly providing a balanced load towards each destination. Each algorithm is tested with an overall number of packets of 50.000.000 equally divided between all the sources. The packet size distribution is exponential with average and minimum length of 500 an 40 bytes respectively. Value of granularity D in bytes was assumed to be 500 bytes as well and the number of FDLs B equal to four for each wavelength. The traffic from each node is supposed to be exponential with Poisson distribution. For each source of

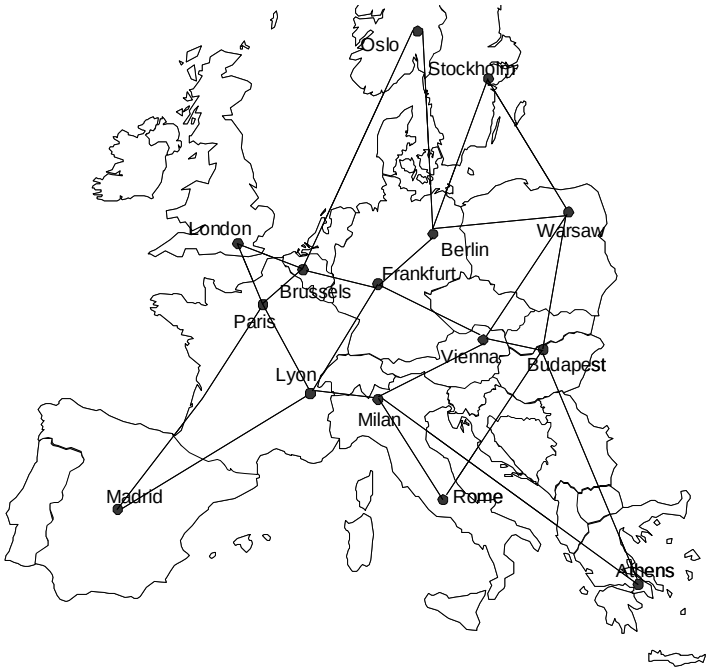

Figure 4.1: The reference network topology

Algorithm	Average # of λs per link	Average # of hops
SL	36.45	2.36
PS-SA	36.45	2.36
PS-MA	36.45	2.36
CS-SA	36.89	2.36
CS-MA	46.33	2.96
CS-AL	48.29	3.01

Table 4.1: Routing algorithms comparison in case of balanced load.

the network the input traffic is generated by 80 M/M/1 queues each one loaded with 0.5, resulting in a 40 Erlangs load from each node. The network dimensioning in terms of wavelengths was performed according to the wavelength sharing policy applied. The procedure used is to allocate as many wavelengths per fiber as required to have an average load per wavelength of 0.8 Erlangs, in accordance to the known traffic matrix. The links involved in this calculation depend on the specific adaptive algorithm and in particular on the default set of links. For instance the first three algorithms have the same default set of links and so the procedure for the wavelength assignment will be the same. The quality parameters considered are the overall packet loss probability (PLP), the average number of hops and the average number of wavelengths per link that basically determines the effective cost of the network. In figure 4.2 results on PLP are plotted mean values of number of wavelength per link and number of hops crossed are shown in table 4.1. These results show the very significant improvement that can be obtained by increasing the level of adaptability, but also the slight increase in the network cost and number of hops that is the price to pay to use longer alternative paths. In fact the number of wavelengths increases in the CS cases and in particularly when longer paths are considered. By providing more and longer routing alternatives the CS algorithms keep the packet within the network longer as shown by the increase of the average number of hops. By doing this they also add traffic to the link in general and therefore need a higher number of wavelength to handle the same total input traffic.

To provide more information to compare the algorithms proposed, the fairness and the ability to deal with un-balanced traffic conditions are also analyzed. Regarding the fairness it can be expected that dif-

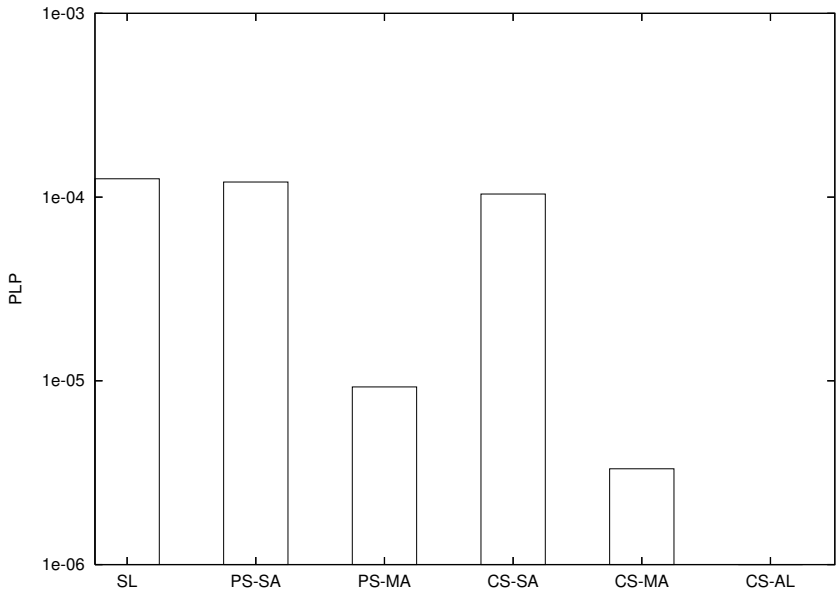

Figure 4.2: Comparison of the average PLP for different routing algorithms in the reference network

ferent links experience different performances depending on the number of wavelengths that they carry and the traffic that they have to handle. A link with a bigger overall load, and therefore with a greater number of wavelengths all loaded with 0.8, works better than a link with less overall load and less wavelengths. This can be explained with the effect of buffers sharing among all the wavelengths assigned to each link which is obviously bigger when there are more wavelengths. For each algorithm besides the packer loss probability (PLP) following parameters are calculated and shown in table 4.2:

- the average packet loss probability among all links;
- the coefficient of variation, that is the ratio between the estimated standard deviation and the average PLP and can be taken as an estimate of the variability of the values from the average[CCRZ03];
- the range that is the difference between the smaller and the larger value of PLP[CCRZ03].

Algorithm	Average PLP	Coeff. of variation	Range
SL	1.31E-4	4.42	3.32E-3
PS-SA	1.24E-4	4.32	3.12E-3
PS-MA	6.57E-6	6.38	2.91E-4
CS-SA	1.21E-4	4.68	3.34E-4
CS-MA	3.09E-6	6.75	1.45E-4

Table 4.2: Average packet loss probability, coefficient of variation and range values for the different algorithms

Since in all the cases there is at least one link whose loss is null, the range value coincides with the maximum value. It can be said that those algorithms that are allowed to use only shortest paths (SL, PS-SA, CS-SA) are the best performing in terms of coefficient of variation whereas those ones that are free to explore even the higher cost paths (PS-MA, CS-MA, CS-AL), provide a lower value of the LP among all the links considered singularly.

To test un-balanced traffic load conditions simulations were performed with 10 per cent of the traffic forced towards some nodes, providing a bigger load for some links. For this test only PS-MA case and CS-MA case were simulated. Again with 50.000.000 packets simulated the results shown in table 4.3 were obtained. As it was expected the CS-MA algorithm reacts better providing the same performance of the previous case. The PS-MA algorithm gets worse by two orders of magnitude instead. In the balanced load case the PS-MA nearly reached the same performance of the CS-MA exploiting less resources but it's not able to assure the same performance in a not balanced case. On the other hand the CS-MA seems to be not affected by this un-balanced traffic. It must also be said that this test depends a lot on which links are going to increase their load and which ones are going to decrease their load instead. In fact, as said before, the links can experience different situations, depending on their position in this particular topology and so the amount and the type of traffic can be different from link to link. Different unbalancing for the network loads were tested and in table 4.3 the average results obtained are shown.

Regarding the performance improvement of the adaptive algorithms, one could think they are just due to increase of network capacity used to

	Average # of ls per link	Average PLP	Average # of hops
PS-MA	36.45	5.68E-4	2.36
CS-MA	46.33	6.98E-6	2.97

Table 4.3: Comparison for unbalanced load

	Average # of ls per link	Average PLP	Average # of hops
SL	48.29	3.53E-2	2.29
CS-AL	36.45	2.08E-1	2.62

Table 4.4: Comparison for CL and CS-AL algorithms with swapped starting conditions in terms of wavelengths assigned per link

guarantee the same load per wavelength. To provide an answer to this remark and prove the real effectiveness of the more dynamic algorithms SL and CS-AL were compared, which can be described as the most static and the most dynamic respectively, by running them on the network set up for the other. In practice the starting conditions between the two cases were swapped. Referring to the first table, the SL algorithm will run with an average number of wavelengths of 48.29 and so the CS-AL with 36.45 wavelengths per link. The result (table 4.4) of the CS-AL performance was quite predictable, having much less wavelengths to exploit comparing to the previous case. Not so predictable was the performance of the SL algorithm instead. In fact it doesn't improve at all but actually the PLP increases quite a lot. It can be said that the improvement of the CS-AL in the previous conditions wasn't only thank to a greater number of wavelengths assigned but it comes from two main aspects: the dynamic algorithm in itself which definitively exploits the resources better and the approach that is used that is to calculate the number of wavelengths by knowing how the algorithm would work and so predicting how many wavelengths would be needed to each link to face the overall input traffic.

So far the simulations were made supposing the number of FDLs equal to four, that means maximum delay equal to three granularities. With the next simulations it was tested how the algorithms change their performances in function of this value. Algorithms for 1, 2, 3, 4 and 5 fiber delay lines were evaluated. Note that the first case basically means absence of buffers, being the maximum delay equal to zero. In figure 4.3 the results show that all algorithms perform with the same order

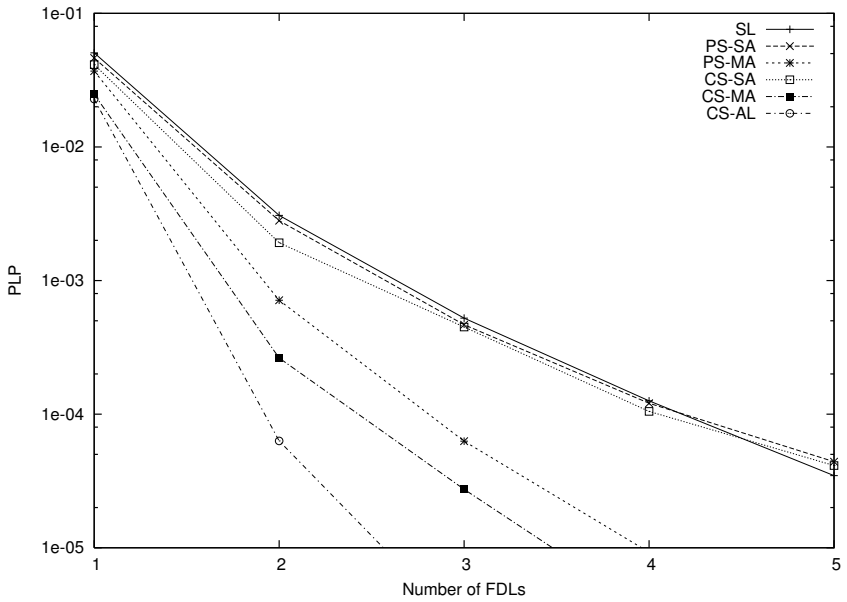

Figure 4.3: PLP as a function of the number of delay lines in the network nodes, comparing the various algorithms

of magnitude when there is no buffer available. As soon as the buffer dimension increases those algorithms with multiple alternative improve more than those using only the shortest paths. In particular the figure shows that the performance gap between the more static (SL, PS-SA, CS-SA) and the more dynamic algorithms (PS-MA, CS-MA, CS-AL) widens with the increase of the optical buffer size, proving the better use made by these algorithms of the queuing resources available in the network.

4.4 QoS in Optical Packet Switching

One of the emerging needs in present day networking is the support of multimedia applications, which demands real time information transfer with very limited loss to provide the end-users with acceptable quality of service (QoS). At the same time economics require an efficient use of the network resources. Assuming that internetworking will be provided by the IP protocol, and accounting for its inability to manage QoS, techniques for QoS differentiation must be implemented in the transport

networks. Significant effort has been developed to define QoS models. In backbone networks the most interesting solutions proposed to solve the QoS problem deal with a limited number of service classes collecting aggregates of traffic flows with similar requirements. This approach can greatly improve scalability and reduce the operational complexity [BBC+98]. Aggregate QoS solutions such as DiffServ are a viable approach also for OPS networks although, because of the limitations of the optical technology, the number of QoS classes must be kept small to minimize operational efforts. In fact, complex scheduling algorithms are not applicable because of the peculiarity of queues in the optical domain, which usually provide a very limited queuing space being implemented by means of delay lines that do not allow random access [HCA98]. This means that traditional priority-based queuing strategies are not feasible in OPS network, and QoS differentiation can be achieved only by means of resource reservation strategies. Previous works show QoS differentiation in an OPS network can be provided with good flexibility and limited queuing requirements by means of resource reservation both in the time and wavelength domains [CCR02][CCRZ04]. These works deal with algorithms for QoS management implemented at the switching node level. Other opportunities arise when considering the routing decisions. In this chapter the study of QoS differentiation mechanisms at the network level is extended, by investigating how the network topology properties can be exploited together with suitable QoS algorithms in order to differentiate the quality along the network paths.

4.4.1 QoS management in OPS networks

A network capable of switching asynchronous, variable-length packets or bursts is assumed. Therefore the results presented in the following may refer both to an OBS network implementing queuing in the nodes [GBPS03] or to an OPS network [OSHT01]. In the following it will generally be referred to an OPS network and assumed that two classes of traffic exist, namely high priority (HP) and low priority (LP). Optical switches are assumed to resolve congestions by means of the wavelength and time domains. The issue of switching matrix implementation is not explored but again a general switching node with N input and N output fibers, carrying W wavelengths each is considered. The switch control

logic reads the burst/packet header and chooses the proper output fiber among the N available. Packets contending for the same output are multiplexed in the wavelength domain (up to W packets may be transmitted at the same time on one fiber) and, if necessary, in the time domain by queuing, implemented with fiber delay lines (FDLs). The FDL buffer stores packet waiting to be transmitted but does not allow random access to the queue. Therefore the order of packets outcoming from the buffer can not be changed and priority queuing is not applicable. Thus QoS management must rely upon mechanisms based on a-priori access control to the optical buffers. In general, after the output fiber has been determined, the switch control logic must face a two-dimensional scheduling problem: choose the wavelength and, if necessary, the delay to be assigned to the packet. This problem is called the wavelength and delay selection (WDS) problem. An optimal solution to the WDS problem is hardly feasible, because of computational complexity and heuristics were proposed in the past [CCRZ04][CCC01][CCRZ03] as said in section 4.2. Here the minimum gap algorithm [CCC01] is again used. In this scenario QoS differentiation is achievable in the node by differentiating the amount of resources to which the WDS algorithms is applied. [CCRZ04] showed that this can be done adopting either a threshold-based or a wavelength-based technique. In the former case, the reservation is applied to the delay units. The WDS algorithm drops incoming LP packets if the current buffer occupancy is such that the delay required is greater than or equal to the threshold, while HP packets have access to the whole buffer capacity. In the latter case the reservation is applied to the wavelengths. A subset of $K \leq W$ wavelengths on any output fiber is shared between HP and LP packets while the remaining $W - K$ wavelengths are reserved to HP packets. Generally speaking, wavelength reservation is more promising because of the larger amount of resources available that provide more flexibility to the algorithms. This is because WDM systems are continuously improving and the number of available wavelengths per fiber is getting larger and larger. On the other hand FDL buffers are bulky and should be kept as small as possible, therefore the number of delays that can be implemented is fairly limited and is probably not going to improve much in the future. The aforementioned approach provides QoS differentiation at the single network node, but does not tackle the problem at the whole network level. A further extension is to define QoS

routing algorithms to obtain even further service differentiation by combining QoS management at the routing level with QoS management in the WDS algorithms. This chapter assumes a meshed network topology and shortest path routing. Traffic is normally forwarded along the shortest path but alternate paths of equal or longer length are also identified and can be used. Two possible routing strategies defined in previous section are here reminded:

- *Single Link Choice (SL)*, that implements a conventional shortest path routing based on minimum hop count and do not use alternate paths;
- *Multiple Alternative (MA)*,besides the shortest path calculates alternate paths that are used by the network nodes when the link along the shortest path (also called default link) becomes congested.

QoS management is achievable by differentiating the concept of congestion and/or providing different alternatives to LP and HP traffic. The proposal analyzed here is as follows.

- The WDS algorithm works with wavelength reservation according to a partial sharing approach; H out of the W wavelengths available are reserved for HP traffic while the $W - H$ remaining are shared between HP and LP traffic. Two options are considered:

  1. The H reserved wavelength may be fixed, namely the W wavelengths available are ordered and the reserved wavelengths are λ_i with $i = 1, \ldots, H$ (FIX strategy).
  2. Any H wavelengths are reserved based on the actual occupancy, namely when at least $W - H$ wavelengths are available both LP and HP packets may be transmitted, otherwise when less than $W - H$ wavelengths are available only HP packets can be transmitted (RES strategy).

- In the routing algorithms congestion is defined according to the wavelength occupancy to determine wavelength availability, when at least T out of W wavelengths are busy the fiber (and the path to which the fiber belongs to) is considered congested. The value of T is different for different classes of service; for LP traffic

$T_{LP} = W - H < W$, while for HP traffic $T_{HP} = W$. This means that for the LP class congestion arises before and alternative path, if any, should be used more frequently.

- Alternate routing is used for LP traffic but not for HP traffic. Therefore HP traffic is always routed with a SL choice, while LP traffic is routed with a MA choice, and alternate paths are used when congestion is present.

The basic idea behind this approach is that the HP traffic stream should be preserved intact as much as possible. Congestion and alternate routing will modify the traffic stream, because of loss, delay, out of sequence delivery etc. Therefore resources to HP traffic are reserved to limit congestion phenomena. There is no rely on alternate routing to avoid as much as possible out of order packets.

4.4.2 Network performance analysis

In this section numerical results are provided to evaluate that the proposed techniques for QoS management may achieve service differentiation at the whole network level. Performance is evaluated in terms of packet loss probability (PLP). Numerical results were obtained by using an ad-hoc, event-driven simulator. The reference network topology is shown in figure 4.4 and consists of 5 nodes interconnected by 12 fiber links carrying 16 wavelengths each. Traffic enters the network at any node and is addressed to any other node according to a given traffic matrix.

The network adopts a connectionless transfer mode, with traffic generated by a Poisson process. The packet size distribution is exponential with average value equal to the buffer delay unit D measured in bytes. This choice minimizes the packet loss at the node level when adopting the Minimum Gap algorithm [CCC01]. The traffic matrix has been set up as follows, with two alternatives.

- Balanced traffic matrix (B): the traffic distribution in the network is uniform since each wavelength carries the same average load (80%). With this approach the input load at different ingress points of the network may clearly not be the same.

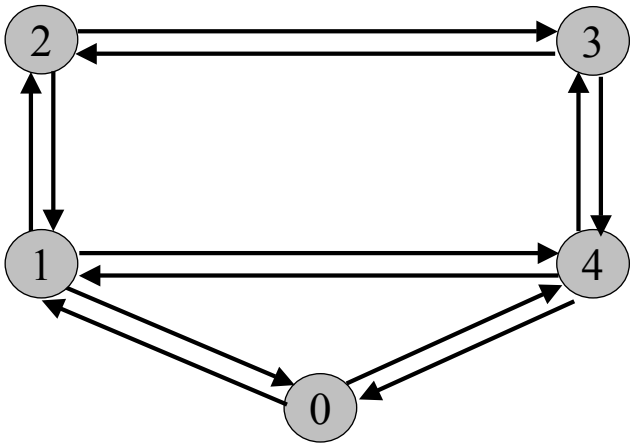

Figure 4.4: The reference network topology

- Unbalanced traffic matrix (U): in this case the traffic load at the ingresses of the network is assumed to be the same. By making this choice the links have a different average load per wavelength, with the only constraint that the maximum value cannot overtake a fixed value (80% in our simulations).

Since in the balanced case each link is loaded in the same way, the average loss probability of the whole network as an evaluation parameter is considered. On the other hand, in the unbalanced case this parameter may not be representative for performance evaluation, therefore the worst loss probability among all links will be taken into account.

At first figure 4.5 compares the FIX and RES strategies for wavelength reservation. The graph clearly shows that the RES strategy performs better for both traffic classes. This result was expected and it is due to the better exploitation of the network resources (the wavelengths in this specific case). In the RES case the reserved wavelength pool is dynamically adjusted to the present state of traffic requests, with a sort of call packing approach. Because of the clear advantage of this reservation strategy, in the following it will be assumed that RES will be used. In figure 4.6 the packet loss probability is shown for different routing algorithms (SL and MA) and different traffic matrices (B and U) considering undifferentiated traffic.

It is clear that MA performs better than SL even though the gain

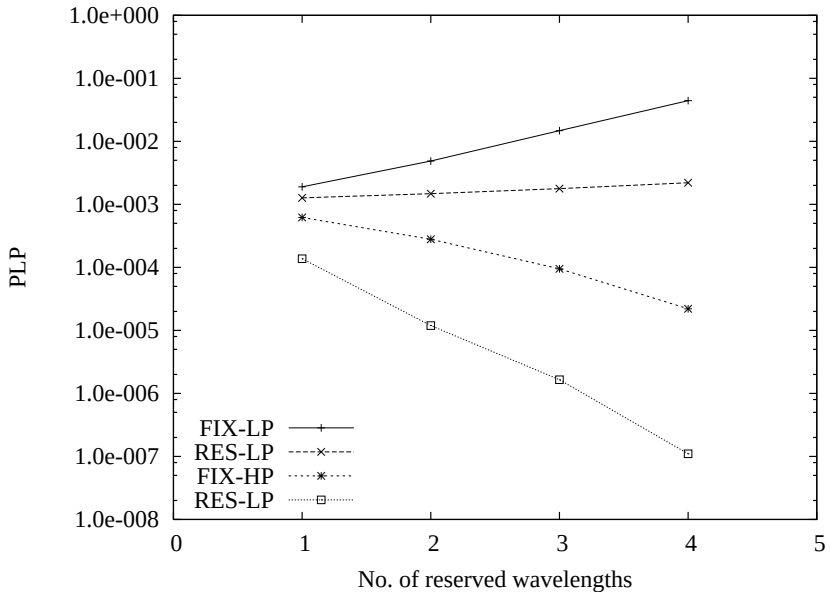

Figure 4.5: Comparison between FIX and RES.

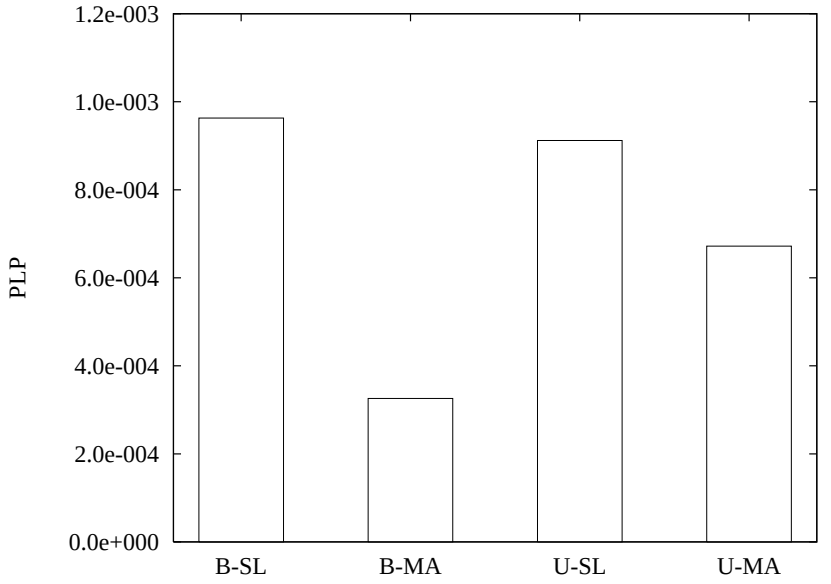

Figure 4.6: Packet loss probability for SL and MA algorithms for undifferentiated traffic and for

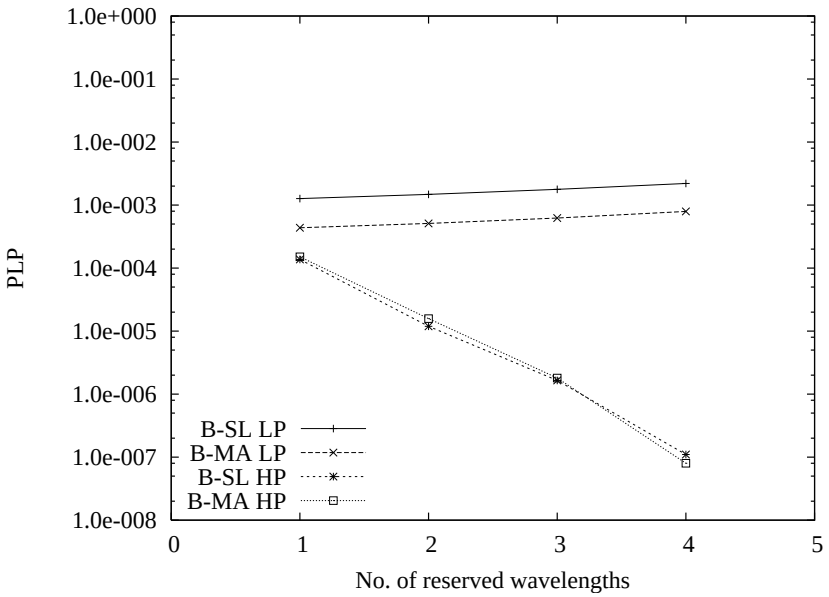

Figure 4.7: Packet loss probability for high and low priority classes with balanced traffic matrix varying the number of reserved wavelengths

is not that big. This can be explained by considering that the network topology is only composed by a limited number of hops and not so many routing alternatives may be actually exploited as, for example, shown in section 4.3 entirely dedicated to adaptive where it was proved that dynamic algorithms perform much better than the static ones in presence of a bigger and more complex network[CCM+04]. However, it is important to take into account that MA keeps packets within the network for longer and then the transmission delay becomes bigger than the SL case. This is why the routing of HP packets always adopts SL. Therefore the choice between SL and MA is relevant only to the routing of LP packets. Since results for the balanced and unbalanced traffic matrix are very similar, only the balanced case is shown in the following.

In order to understand how different choices affect the network behavior, figures 4.7 and 4.8 show the results obtained by different approaches. First performance assuming a fixed percentage of the HP class set to 20% are evaluated while the number of wavelengths reserved to HP class varies from 1 to 4 out of 16 on each link (figure 4.7). Then, the percent-

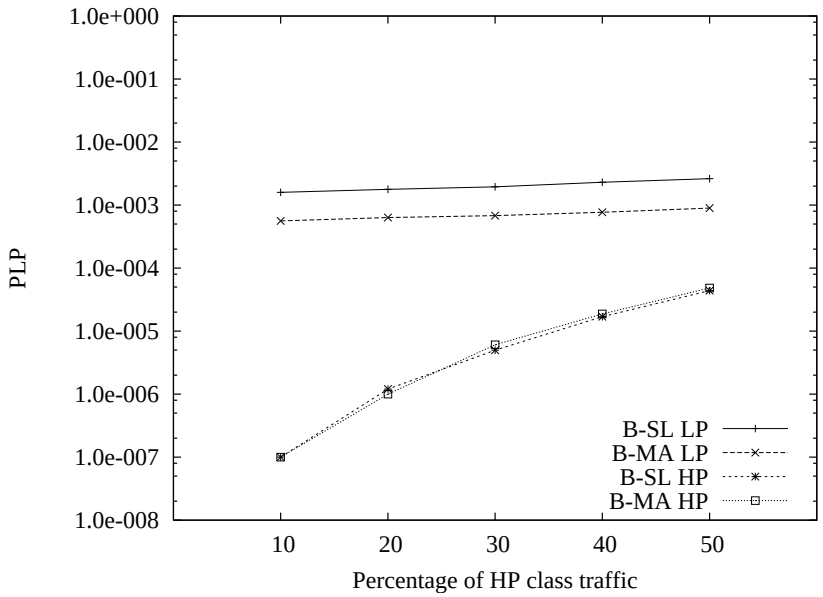

Figure 4.8: Packet loss probability for high and low priority classes with balanced traffic matrix varying the percentage of HP traffic

age of HP input traffic varies between 10% and 50% while the number of wavelengths reserved to HP class is fixed to 3 for each link (figure 4.8). As expected, the higher the number of dedicated wavelengths, the higher the gain in terms of loss probability that can be obtained for the HP class. When 1 to 4 wavelengths are reserved, the loss probability improves by three orders of magnitude, while the performance of the LP class is barely affected. Packet loss probability remains nearly constant at one order of magnitude worse than the undifferentiated case. HP class reaches very low packet loss probability (10-7) when resource reservation is equal to 25% (i.e. 4 wavelengths) of the whole set. Moreover, it is also not affected by the fact that LP class can be routed in different ways. On the other hand, for a very low percentage of HP traffic, good level of performance may be achieved. When HP traffic grows over the 10% performance starts getting worse quite rapidly, while the LP class again seems to be slightly affected. It follows that in case a given loss probability is required by HP traffic, the admission to the network has to be kept under control in order to avoid performance degradation due to the

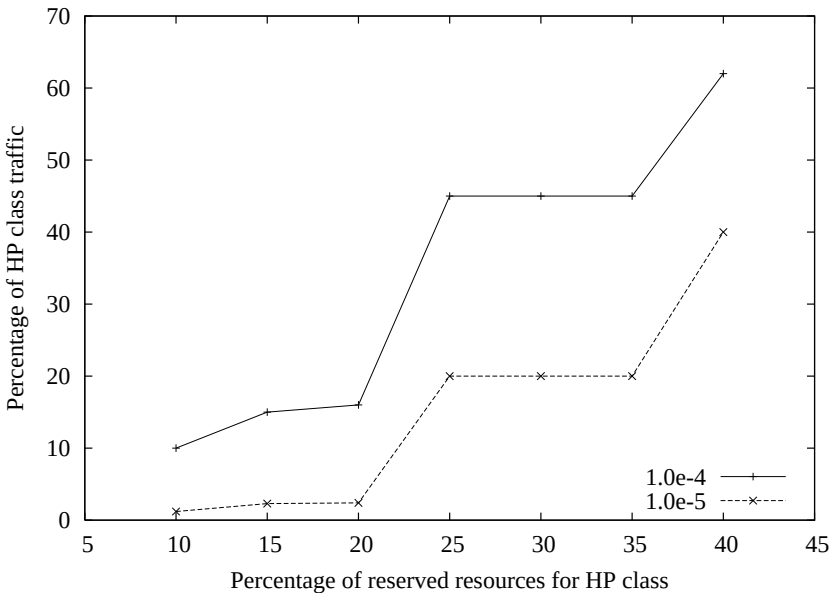

Figure 4.9: Relation between the HP traffic percentage and the percentage of resources reserved for given packet loss probability

limited resources reserved to HP class. A good degree of differentiation between the two classes may be obtained in both cases reaching up to four orders of magnitude, while the adaptive routing strategy for LP traffic allows a further performance improvement. The results presented in figures 4.9 and 4.10 let understand how the amount of reserved resources and the percentage of HP traffic are related when a given value of the HP class packet loss probability (PLP) is required. Only the SL algorithm is considered here. As expected, in case a given packet loss probability has to be guaranteed for an increasing percentage of HP traffic, more resources must be reserved. Furthermore, figure 4.10 shows the corresponding performance of the LP class.

4.4.3 Network design

In this section a network design procedure is presented. The reference network is the same as above but the aim now is to calculate, with relation to the SL routing algorithm and to the traffic matrix adopted, the

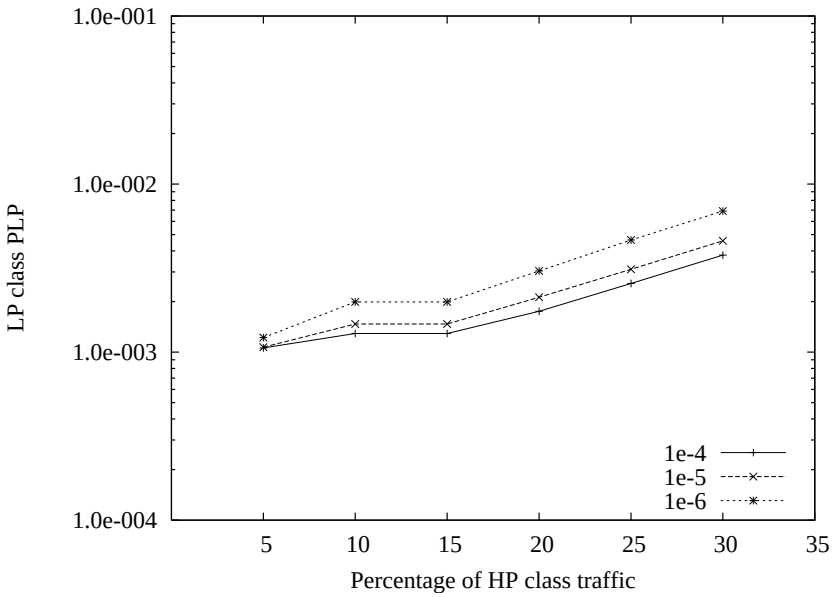

Figure 4.10: Relation between the LP traffic percentage and the percentage of resources reserved for given packet loss probability

number of wavelengths required per fiber so that a given average load per wavelength is obtained. The main assumption is that all nodes generate the same input traffic which is uniformly distributed to the other nodes. The input traffic value is chosen so that the total number of wavelengths is very close to $16 \times 12 = 192$ as in the previous case, each wavelength being loaded at 80%. This allows a better comparison with almost the same cost in terms of wavelengths. The resulting resource distribution varies from 7 to 28 wavelengths per link. In the design procedure it is important to adopt the MA approach, otherwise performance decreases. This is due to the fact that SL, not sharing the wavelength resources, does not achieve load balancing. With the MA approach performance is the same as the balanced case with the advantage that the traffic matrix is now imposed by user needs instead of network configuration as before. In figure 4.11 the performance of SL and MA is shown for both classes varying the percentage of wavelengths reserved to HP traffic (set to 20%). Obviously, when the percentage of reserved wavelengths is low there is a bad service differentiation between the two classes. Moreover the trend

PLP	It.	0	1	2	3	4	5	6	7	8	9	10	11
10^{-1}	0	14	28	21	21	14	14	21	14	14	14	14	7
10^{-2}	1	14	28	21	21	14	14	21	14	14	14	14	8
10^{-3}	2	15	28	21	21	14	14	21	15	14	14	15	9
10^{-4}	3	15	28	21	21	15	15	21	15	15	15	15	9
10^{-5}	4	16	28	22	22	16	16	22	16	16	16	16	10

Table 4.5: Number of wavelengths required to achieve different packet loss probabilities for each link (0-11); in the 2^{nd} column the number of iterations needed to get to the convergence is indicated.

of the curve for HP class is not as smooth as before. This is because losses are not uniformly distributed over the links, varying in a range between 10^{-2} and 10^{-6}. The reason why this happens is because different numbers of wavelength are available on different links due to the dimensioning procedure, providing different levels of wavelength multiplexing. In fact, links with less wavelengths experience worse performance. Thus the overall HP packet loss probability curve starts improving when more resources are added to these specific links. LP class seems to be less affected and its loss probability remains of the same order of magnitude with MA performing better than SL as usual. To improve the network design, a maximum acceptable packet loss probability per link may be fixed. Then simulation is iterated by adding wavelengths to those links that show higher losses until the loss constraint is satisfied. The drawback of this methodology is that the simulation time increases. The average wavelength load at the beginning is set to 80% but of course, when more resources are added, some links result less loaded. Moreover, at the end of the process links still do not have the same blocking probability, but at least all of them satisfy the loss requirement. The chart depicted in figure 4.12 shows the number of additional wavelengths (as a percentage of the starting number) that must be added to each link in order to meet different packet loss requirements. In table 4.5 the number of iterations and the corresponding number of wavelengths for each link required to achieve different loss constraints are shown.

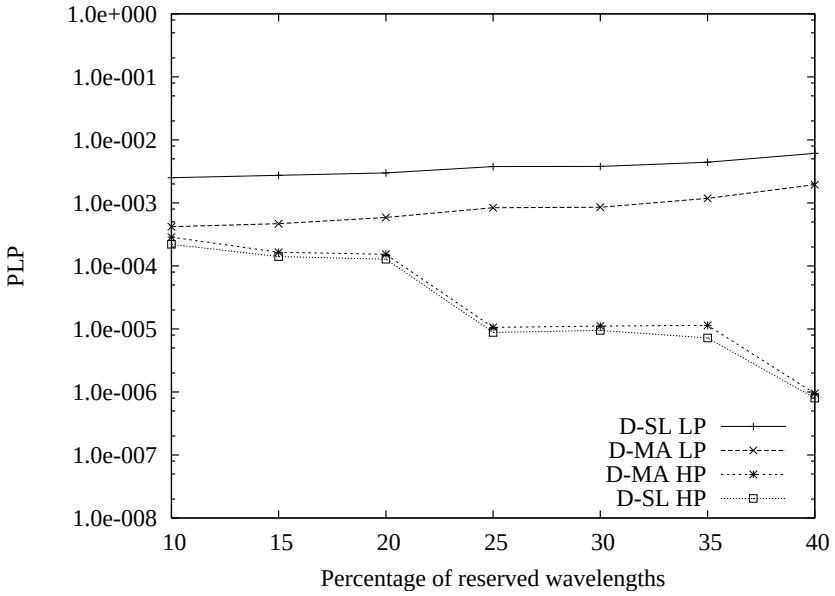

Figure 4.11: Packet loss probability for high and low priority classes resulting from the network design procedure as a function of the percentage of the resources dedicate to the HP class with % of HP class traffic

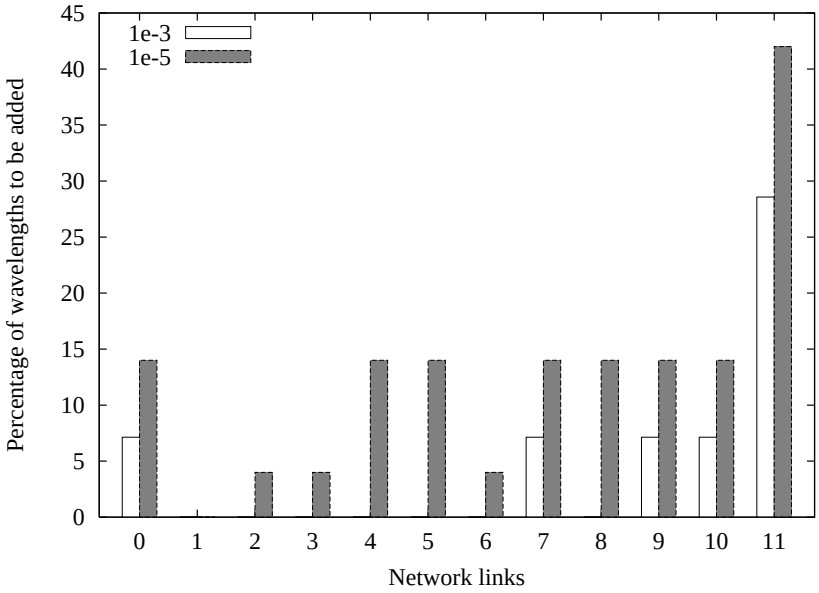

Figure 4.12: Percentage of additional wavelengths required by each link for different loss constraint

4.5 Resilience in Optical Packet Switching

Reliability in optical networks has been a widely discussed research topic, being a very important requirement for the next generation optical networks [VPD]. Different kinds of failure can happen even at the same time and the network must be capable to recover from them. This must be done in an efficient way by detecting the failure as quickly as possible, so that a recovery procedure can be called immediately. Information loss needs to be limited during the failure detection time, as well as when recovery is taking place. To this purpose a smart and efficient recovery algorithm is required. The general classification of possible network recovery choices is also suitable to optical networks. In particular, dynamic resilience proves to be a better choice with respect to static approaches, that reserve spare capacity in advance, before any connection is set up [FV00]. Dynamic resilience can be performed implementing either protection or restoration schemes. Protection techniques assume the availability of precomputed back-up paths, resulting in quick recovery

times but also requiring spare resource pre-allocation (either dedicated or shared). On the other hand, restoration schemes are capable to deal with failures as soon as they happen, resulting more efficient in terms of capacity utilization, but relatively slow in terms of recovery time. Additionally, the resilience degree may be one of the parameters on which service differentiation is based [AJVC04]. In this sub-section the problem of recovering from single link failures in an optical packet switched network is addressed. In a multi-layer network paradigm, such as IP over WDM, multi-layer recovery schemes may also be applied. However, it is not the case of this study, where only the optical layer is considered and thus only the optical recovery approach is assumed to be implemented. Two different failure recovery models are considered and compared in section 4.5.1, with the target to outline their effects on network performance.

4.5.1 Failure recovery in optical packet networks

In order to exploit the flexibility provided by packet switching for reliability purposes, a different approach is considered here, with respect to traditional resilience schemes used in optical network design, which are based on the concept of lightpath [RS]. In the following, a pure connectionless environment is assumed, where packets are independent from each other and are routed towards their destination following the shortest path. In case this one is congested, alternative paths (either shortest or not) are exploited. In this scenario the terms shortest path or recovery path refer to the single optical packet, according to its destination. Packets belonging to the same information stream or connection may travel on different wavelengths, due to a dynamic WDM management approach adopted [CC01], or even on different fibers, if they experience different congestion situations and a dynamic routing algorithm is applied [CCM+04]. As a consequence, in case of a link failure, packets will be re-routed on alternative paths, but always in a connectionless perspective. The reference network topology is shown in figure 4.13. Each vertex in the graph represents an OPS node, while each edge represents a pair of fiber links, transmitting packets in opposite directions. Nodes and fibers are identified by progressive numbers, while each edge is identified by the couple of indexes of the corresponding fibers (the former index

belongs to the fiber outgoing from the node with the lower identification number). Due to this assumption, the number of unidirectional links in the network is twice the number of edges in the corresponding graph. Packets are injected in the optical network at each node, with an header that contains full information about its destination, making the packet completely independent from the rest of the stream. Incoming traffic on ingress nodes is uniformly distributed towards every possible destination following the shortest path and the number of wavelengths per fiber link is dimensioned accordingly, in order to obtain an average load per lambda equal to 0.8 Erlangs. Packets belong to two traffic classes, namely high and low priority. Wavelength capacity is assumed at 10 Gbit/s. Each node is provided with a full set of wavelength converters as well as FDL buffers, which are the resources needed to solve packet contentions. The wavelength and delay selection policy adopted is again the Minimum Gap.

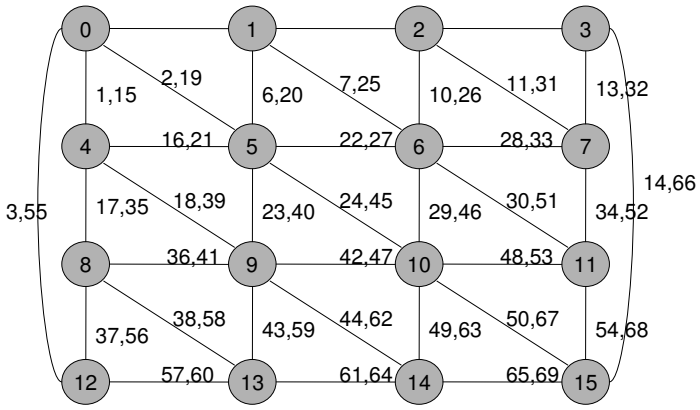

Figure 4.13: The reference network topology

From the routing point of view, the so called Partial Sharing with Multiple Alternative (PS-MA) algorithm is used, as defined before. Service differentiation is achieved by reserving a certain percentage of wavelengths on each fiber to the high priority class. In the following, two different approaches are adopted when failure occurs are considered, namely *link protection* and *dynamic routing*. In normal conditions, the routing algorithm at each node sends the packets on the shortest path between

source and destination. When the link on the shortest path fails, a recovery procedure must be called. If *link protection* is applied, an alternative path between the two nodes that were linked by the broken fiber is used. In case *dynamic routing* is implemented, an alternative path towards the final destination of the packet is calculated. Figure 4.14 shows the shortest path between node 1 and 14 and figure 4.15 explains the behavior of the two approaches using the reference topology when link 23 is down.

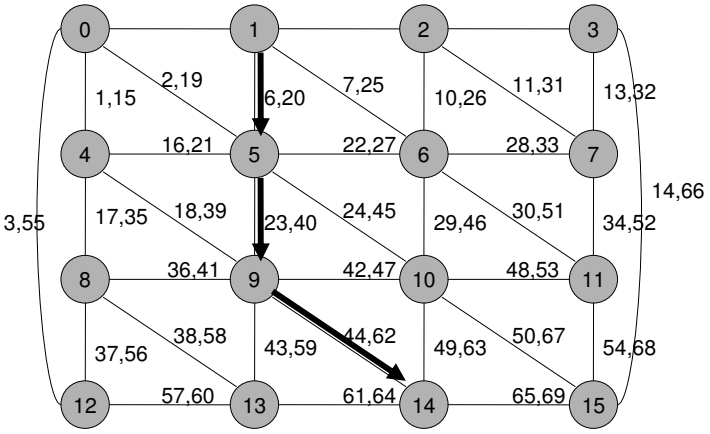

Figure 4.14: Shortest path between node 1 and 14.

4.5.2 Analysis of the link loss probability during the failure detection

In a packet-switched network subject to link failures, a key performance parameter is the time required to detect the failure and start re-routing the packets: obviously, the shorter the fault detection delay, the smaller is the number of packets lost due to link unavailability. This section is devoted to the analysis of the link loss probability behavior during the failure detection time. Link failure detection is assumed to be realized by each node with a receiver time-out set on each incoming fiber. If no information is received before one of such timers expires, the corresponding fiber is considered to be out of service and this event is notified to the involved node. By indicating with t_f the time when the failure occurs and with t_r the time when it is detected, the *detection time* is defined as

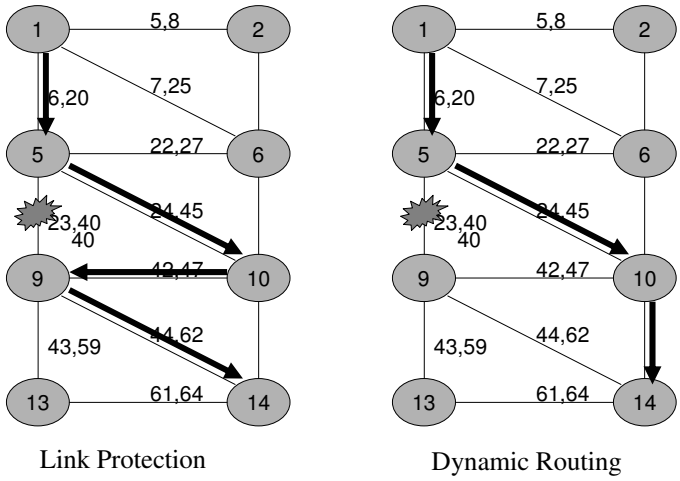

Figure 4.15: Comparison between *link protection* and *dynamic routing* algorithms.

$\Delta t_D = t_r - t_f$. A set of simulations were run, monitoring the packet loss on link 52, which was subject to failure at a given instant. In figure 4.16 the trend of packet loss probability on link 52 is shown as a function of simulation time for different values of the detection time.

As long as there was no failure, performance was the same for all values of detection time. Obviously, when failure occurred things got worse for longer values of Δt_D. After an increasing phase, loss probability of the link established at a fixed value, because the link was not used anymore and the traffic was diverted towards other routes after the detection. It is possible to divide the whole simulation time in three main phases, as shown in figure 4.17:

1. *Pre-failure*: network works in a failure-free regime;

2. *Failure has occurred but not detected yet*: all packets across the failed link are lost;

3. *Failure is detected*: the node where the failed fiber starts from is finally informed about the failure and therefore it starts re-routing packets on alternatives paths according to the routing algorithm.

Let the simulation time be segmented into n very small intervals. The

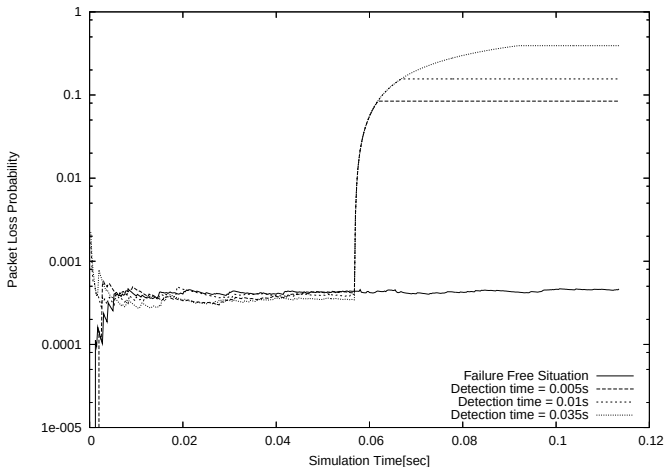

Figure 4.16: Percentage of searching operations for different values of load per wavelength for class 0 and class 1.

packet loss rate on a given fiber subject to failure in the ith interval is

$$P_{Li} = \frac{n_{Li}}{N_i} \qquad i = 1, ...n \tag{4.1}$$

where n_{Li} and N_i are the number of packets lost and generated in a generic interval i respectively. N_{Li} and N_i represent the total number of packets lost and generated up to interval i respectively and may be expressed as:

$$N_{Li} = \sum_{K=1}^{i} n_{lk} \qquad and \qquad N_i = \sum_{K=1}^{i} n_k \tag{4.2}$$

In the first phase ($i < F$ being F the interval when failure occurs) the number of packets lost in any interval is

$$n_{lk} = p \cdot n_k \tag{4.3}$$

being p the loss probability of the link in a failure free situation.Thus:

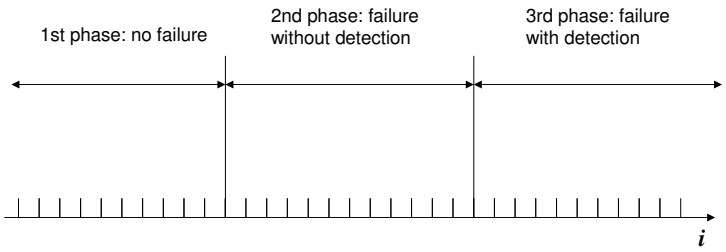

Figure 4.17: The three phases across the failure and its detection.

$$N_{Li} = \sum_{K=1}^{i} p \cdot n_k = p \cdot N_i \tag{4.4}$$

In the second phase ($F < i < R$, R = interval when the failure is detected) all packets sent to the broken fiber are lost and so $n_{Lk} = n_k$ with $k = F+1, F+2, \ldots, R$. The number of packets lost at the end of second phase is:

$$N_{Li} = \sum_{K=1}^{F} p \cdot n_k + \sum_{K=F+1}^{i} = p \cdot N_F + N_i - N_F = N_i - (1-p) \cdot N_F \tag{4.5}$$

In the third and last phase ($i > R$) $n_{Lk} = n_k = 0$ with $k = R+1, R+2, \ldots$ since no packet will be sent to the failed fiber anymore. Therefore:

$$N_{Li} = \sum_{K=1}^{i} n_{Lk} = \sum_{K=1}^{R} n_{Lk} = N_{LR} \quad with N_i = N_R \quad \forall i > R \tag{4.6}$$

In conclusion the total percentage of lost packets is:

$$P_L = \lim_{i \to \infty} \frac{N_{Li}}{N_i} = \frac{N_{LR}}{N_R} = 1 - (1-p) \cdot \frac{N_F}{N_R} \tag{4.7}$$

In the time domain it results:

$$P_L = p \quad when \quad t \leq t_f \quad (1^{st} phase) \tag{4.8}$$

$$P_L = 1 - (1-p) \cdot \frac{t_f}{t} \quad when \quad t_f \leq t \leq t_r \quad (2^{nd} phase) \tag{4.9}$$

$$P_L = 1 - (1-p) \cdot \frac{t_f}{t_r} \quad when \quad t \geq t_r \quad (3^{rd} phase) \tag{4.10}$$

This model requires a time origin t_0 to be fixed. Here t_0 is assumed to be the time when simulation starts. In figure 4.18 a comparison between theoretical and simulated results for packet loss probability on the single link is shown as a function of the detection time showing the expected matching.

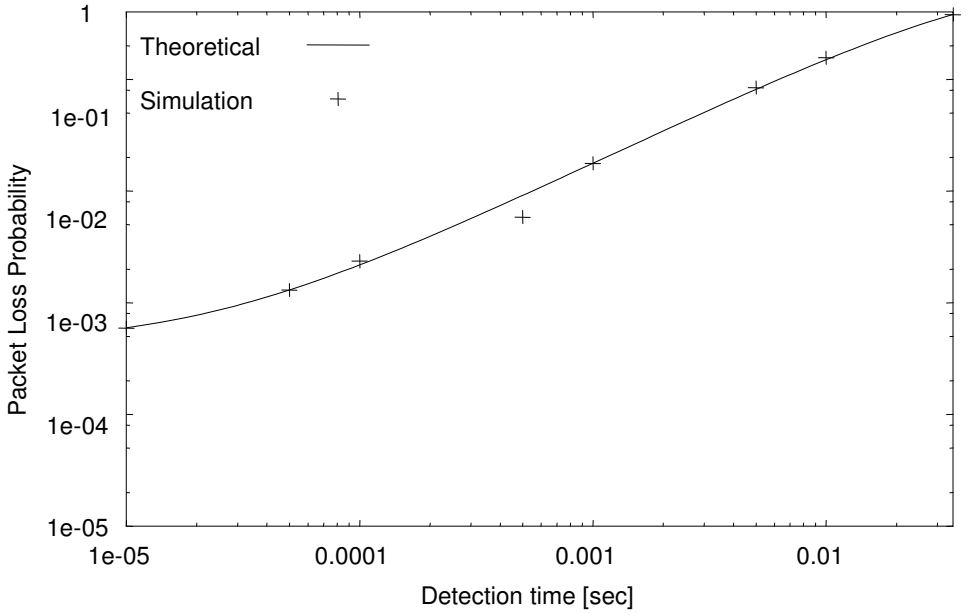

Figure 4.18: Comparison between theoretical and simulation results for packet loss probability.

Simulations follow very well the results obtained with the theoretical model. For high values of Δt_D the loss becomes very high and no longer acceptable. On the other hand, low values of Δt_D can keep loss probability very limited. However, this behaviour could be deviating. If Δt_D is

kept too short, i.e. in the order of a few microseconds, then a false alarm can happen, i.e. when a silence period is longer than the receiver time-out. The node assumes that a failure occurred and calls the recovery procedure, even though this is not necessary. This behavior obviously depends on the traffic load. Simulations have shown that for an average load of 0.8 Erlangs per wavelength $\Delta t_D = 10\mu s$ is a failure detection time-out long enough to let false alarms never happen in practice.

4.5.3 Network Performance with single fiber failure.

Comparison in terms of throughput between the two restoration techniques discussed in previous sub-section is plotted in figure 4.19, using the x axis in a decreasing scale.

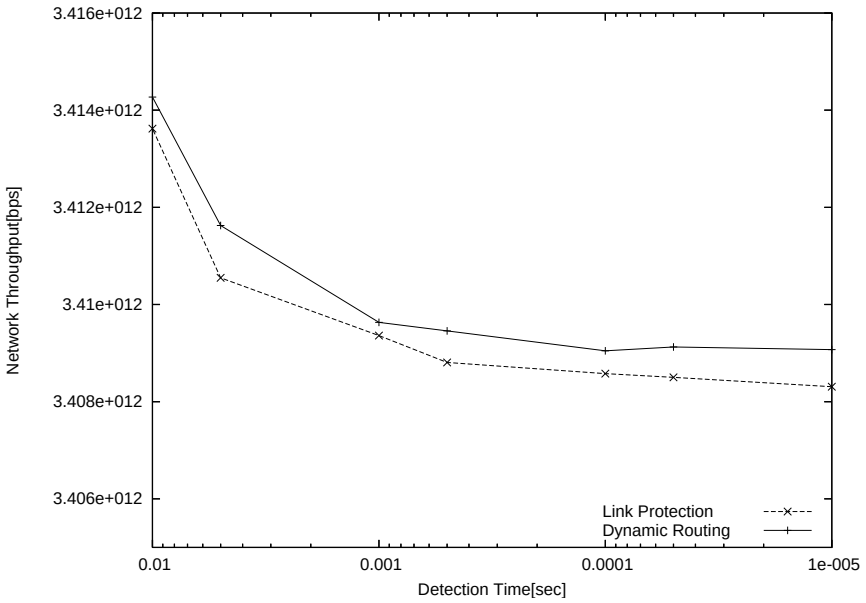

Figure 4.19: Throughput comparison for *Linkprotection* and *Dynamicrouting*.

The longer the detection time the higher the overall throughput. This is a consequence of the higher network traffic after the failure is detected and packets are delivered to links adjacent to the failure. Overall, the

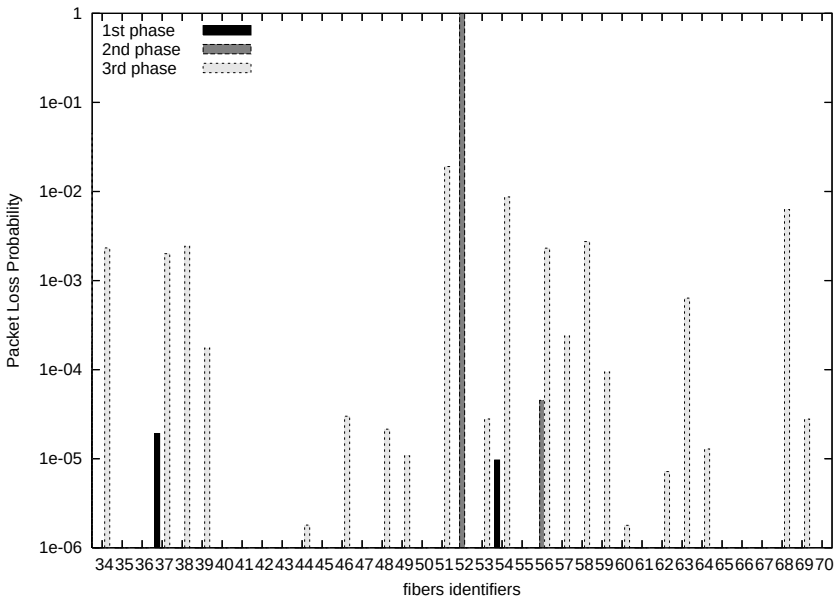

Figure 4.20: Loss probability for each fiber during the three phases.

original amount of traffic is spread over a network lacking one fiber. This causes the average load to be higher and some nodes to be overloaded. Figure 4.20 shows indeed that performance per single fiber gets worse during the third phase. This result was obtained with a relatively high initial load (0.7 Erlangs per wavelength). Therefore, when the network works with high load and a failure occurs, adaptive routing is no longer efficient and it cannot re-establish the situation without failure. A protection scheme is then needed.

4.5.4 Design guidelines

In this section a protection scheme with shared resources is described. This algorithm is applied both to *link protection* and *dynamic routing* techniques, with and without service differentiation. Single failures of three links differently located over the network are simulated. Protection from the failure is realized by dimensioning the network in two main steps:

1. first of all, the network is dimensioned to have an average load per

	NO QoS	QoS
Link Protection	$1.823 \cdot 10^{-02}$	*high* 0, *low* $2.328 \cdot 10^{-02}$
Dynamic Routing	$9.658 \cdot 10^{-03}$	*high* 0, *low* $1.211 \cdot 10^{-02}$

Table 4.6: Loss probability with failure on link 0

	NO QoS	QoS
Link Protection	$1.023 \cdot 10^{-02}$	*high* 0, *low* $1.378 \cdot 10^{-02}$
Dynamic Routing	$5.876 \cdot 10^{-03}$	*high* 0, *low* $7.345 \cdot 10^{-03}$

Table 4.7: Loss probability with failure on link 68

wavelength equal to 0.7 in relation to the traffic matrix;

2. then, further wavelengths are added to each fiber starting from the same node so that it will see all its output fibers with the same capacity.

With this implementation the additional cost due to protection results to be 73% of the initial cost in terms of number of wavelengths. Fiber 0 is the one with the highest number of wavelengths after step 1. The one with the lowest number of wavelengths is fiber 68. The failures of fibers 0 and 68 are simulated. An intermediate case (when fiber 23 fails) is also considered. Results with reference to phase three are shown in tables 4.6, 4.7 and 4.8 when two QoS classes (High and Low) are considered, as well as without service differentiation. When QoS is applied, the total traffic mix is 20% of high priority packets and 80% of low priority packets. The threshold of wavelengths reserved to high priority class is set at 25% of the total in each fiber.

Dynamic routing is shown to guarantee better performance with respect to *link protection*. When fiber 68 is down the network recovers nearly completely from the failure. When a more loaded fiber fails, the

	NO QoS	QoS
Link Protection	$5.5 \cdot 10^{-07}$	*high* 0, *low* $2.425 \cdot 10^{-06}$
Dynamic Routing	0	*high* 0, *low* 0

Table 4.8: Loss probability with failure on link 23

losses are more evident instead, even though links around it have the maximum capacity available. Figure 4.21 focuses on the failure of fiber 23 and shows the trend of the throughput. It can be seen that the situation of failure-free scenario is practically recovered. Considering the three phases discussed above, when failure occurs (second phase) performance drops drastically. However, without the protection scheme the throughput goes even worse after detection (third phase), whereas when protection is applied the original throughput is practically restored.

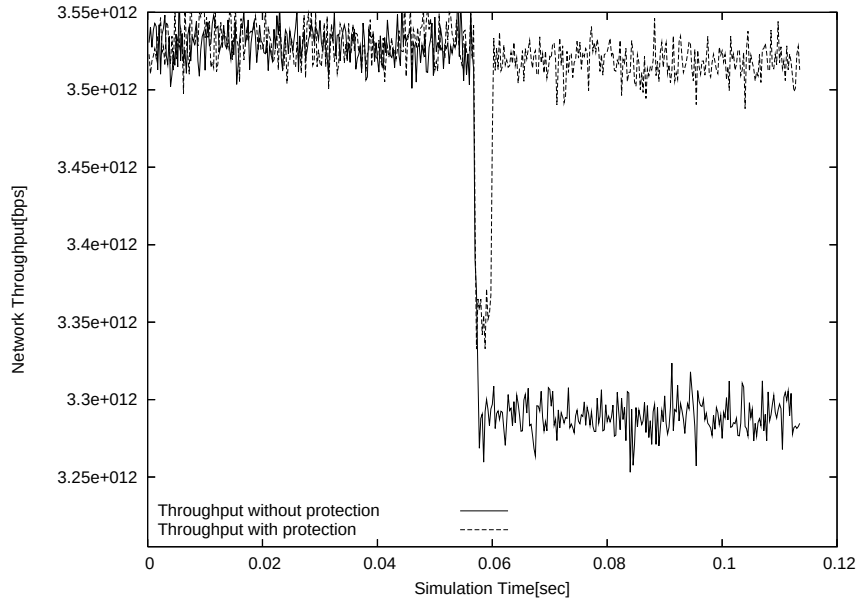

Figure 4.21: Throughput in cases with and without protection scheme.

4.6 Final remarks

In this chapter some routing algorithms were presented for an optical packet network that exploits the wavelength domain for statistical multiplexing of packet traveling on the same routes. Algorithms with different strategies for wavelength multiplexing have been considered. The more adaptive algorithms optimizes the choice of the output wavelength within a wider set of links which belongs to both shortest paths or higher cost

paths. These algorithms were tested on a particular topology of an optical core packet network. Different network configurations and different parameters have been taken in consideration for a better understanding of the behavior of the algorithms. The efficiency of an algorithm is basically a compromise between performance in terms of loss probability and the cost expressed in term of resources exploited. The results prove that the more the algorithm is adaptive the better it performs. The drawback is that even the network cost is higher.
Second part focused on the problem of quality of service differentiation in WDM packet-switched optical networks has been addressed. The effects of quality of service routing have been shown by applying dynamic wavelength management on each link jointly with static or dynamic routing strategies. Different quality of service algorithms have been analyzed and then applied to a network dimensioning procedure. The sharing effect produced by the dynamic routing algorithm proved to be particularly effective in this situation. An iterating procedure has then been applied to achieve loss balancing over network links with relation to design constraints. The main conclusion is that the use of assigned wavelength results optimized in relation to the performance target.
Finally the problem of resilience in a Optical Packet-Switched Network was investigated. In particular a single link failure has been studied and a way to recover such an inconvenient has been proposed based again on adaptive routing techniques. A design procedure has been described to exploit the proposed approaches and evaluated by simulation.

Chapter 5

Frameworks on problem related with DWDM optical networks

In this chapter two works related with optical packet switching networks will be presented. First the problem of effective implementation of void filling on OBS networks with service differentiation will be explored with some new proposals. The second work concerns with the packet assembly topic that has to be done at the edge node of optical networks.

5.1 Effective implementation of void filling on OBS networks with service differentiation

In this section the problem of channel reservation in Optical Burst Switching nodes is considered with the aim to reduce the scheduling processing time and, at the same time, to maintain burst loss probability as low as possible. A new implementation of the void filling problem related to a multi-class traffic environment is proposed based on the binary heap tree data structure. Fast search of the void intervals for burst scheduling is achieved by means of vector implementation of the tree that contains information about voids. The search is further optimized by reducing the search interval to obtain a good trade off between processing time and

burst loss probability in relation to traffic load. Results obtained by simulation show interesting improvements of the scheduling time compared with other implementations of void filling, while, at the same time, maintaining burst loss probability low, especially for high quality traffic.

5.1.1 Recalls on optical burst switching

In recent years Optical Burst Switching (OBS) has obtained growing attention in the research community and industry [QY00][Tur99]. It is claimed to combine the best of optical circuit and packet switching with the aim to dynamically exploiting the huge bandwidth made available by fibers [SGLG03][XPR01][TGCT99]. OBS networks can be viewed as a possible solution for the implementation of high-speed optical backbone that efficiently interconnects peripheral IP networks. To this end packets are assembled into burst at ingress edge nodes and disassembled into packets at network egress. The main property of the burst switching concept is to separate the transmission of data from control information to make control processing more feasible. More specifically, the OBS paradigm provides that a control packet is sent before the transmission of each data burst for resource reservation at each intermediate node along the edge-to-edge network path, giving rise to an offset time between data and control itself. The control packet is typically delivered out-of-band by using common channel signaling, e.g. using a separate wavelength, and can be transmitted with an extra-offset in advance to implement service differentiation techniques [YQD01]. The performed reservation is one-way and can be based on different approaches [XVC00], which differ for the length of the interval during which network resources are available for a given burst. An efficient exploitation of network resources is achieved by the so-called JET (Just Enough Time) algorithm [XVC00][YQ97][TR03]. In JET the reservation at each node is based on the expected burst arrival time, r and lasts for a period of time equal to the burst length, l. The arrival time is calculated taking the difference between the offset time and the processing time accumulated by the control packet at previous nodes. It is important to point out that by maintaining the second term low it is possible to limit the amount of the overall offset. This reservation in advance causes the rising of void intervals over different wavelengths, as shown in 5.1, that repre-

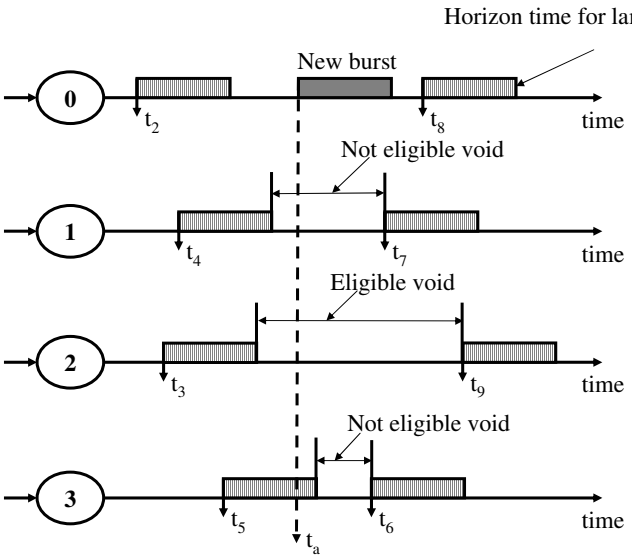

Figure 5.1: Wavelength usage fragmentation in time due to burst scheduling based on JET, with a new burst starting at T_a

sents a challenge to optimize bandwidth utilization. As a consequence a primary target in OBS networks is to implement efficient scheduling of burst transmissions in order to try to suitably fill voids with new arriving bursts, ensuring in this way an efficient use of the available bandwidth, possibly without penalizing network performance.

Moreover in this work a service differentiation is implemented through the usage of extra offsets assigned to different classes. In order to obtain a good class isolation long extra offset must be used. This leads to the creation of very long void intervals that must be exploited to keep loss probability as low as possible. The organization of these intervals in a proper structure is also crucial. In order to support the reservation function, tree data structures are typically built at each node to appropriately organize all the available void intervals for each channel. So, the processing time to find a suitable channel for a burst is strictly related to the searching method used for the investigation of the tree. Nonetheless, these structures grow with the number of available intervals and require to be updated whenever either a new interval is generated or an existing interval is no more available, being it already assigned to a burst or ex-

pired. Such operations contribute to the overall processing time needed for scheduling. Moreover, it is worthwhile remarking that a scheduling algorithm is as much efficient as it is able to process a control packet and find a suitable void quickly and smartly enough to guarantee the reservation for the related arriving burst. So both implementation and consequent management of the data structure used for organizing the void intervals play a significant role in satisfying the above-mentioned requirements. The tree structures presented in literature and used for solving the channel scheduling problem are typically implemented by means of linked lists [CLR90][McC85]. The crucial requirement is to resort to memory allocation functions and pointer management that are time consuming tasks. The aim of this work is to apply a new tree organization model for burst scheduling, based on the binary heap concept, and prove with several numerical evaluations that the proposed data structure and related major management operations permit to obtain efficient void filling scheduling. The key aspect of this approach is the implementation of the tree as a vector, whose elements are directly accessible through their related indexes. The processing time and the burst loss rate are calculated to prove the effectiveness of the proposed implementation and compare its performance with respect to other existing algorithms presented in literature.

5.1.2 Problem description

The main aspect of OBS networks that will be here considered is the presence of the offset time and of the related void interval that arises as a consequence of resource reservation, which resource reservation starts when the control packet arrives at the node and is kept for the whole time duration of the burst itself. Taking into account that bursts do not get to a node one right after another, void intervals in channel bandwidth usage arise. As already mentioned, one of the most-used reservation method is the JET protocol, which leads to efficient bandwidth exploitation providing that scheduling algorithms are implemented to utilize this unused time interval. In fact, the created void can be used by other bursts to achieve good bandwidth utilization and a consequent reduction of the burst loss rate. Offset time between control packets and burst is about some microseconds. Nonetheless, JET can be used for Quality of Service

(QoS) differentiation by assigning different extra offsets to diverse traffic classes of service [Tur99]. This extra amount of time is then added to the offset time, which is normally assigned to each burst. Larger extra offsets will be assigned to higher priority classes with respect to the lower priority ones. In this way, resources for highest-priority bursts can be reserved more in advance with a consequent reduction of the corresponding drop probability, which can be improved of several orders of magnitude in comparison with the undifferentiated traffic case [Tur99]. The problem to exploit the void intervals is usually called void filling. As regards, the introduction of QoS management in OBS networks leads to an increase in the number of idle time intervals (voids) available on different wavelengths of a given fiber and so, an efficient interplay between smart scheduling algorithms and the JET approach is required to properly exploit these intervals and improve the overall network performance. Different approaches have been proposed in literature to execute this task and try to achieve the best trade-off among performance, complexity and scheduling delay[XQLX03][DGSB01]. OBS can also take advantage of the usage of Fiber Delay Lines (FDLs) settled in each node, where bursts can be stored until a channel becomes available, but this solution undoubtedly complicates the design of scheduling algorithms. In fact, by using discrete delay unit as FDL more voids are created and hence the complexity of void filling problem becomes strictly related with the dimension of the optical buffer and depends on whether the transmission is synchronous (or not) or if the length of the bursts is variable (or not) as explained in [TGCT99]. So far, some scheduling algorithms have been already studied and discussed in terms of time scheduling and performance. The first one called HORIZON has been described in [Tur99]. It can be seen as an extreme case where no void filling is made basically. This algorithm considers the horizon time for a given channel as the time after which no reservation is applied and so the next arriving burst can only book this channel after the horizon time. This algorithm results very simple and fast but also bad performing. In [XVC00] a smarter algorithm called LAUC-VF (Latest Available Unused Channel with Void Filling) is presented. In LAUC-VF the key information about voids (i.e. starting and ending time), or at least those whose ending time is greater than the current time, are stored. Among all eligible void intervals for a specific burst the latest is chosen (i.e. the one with the latest starting time).

LAUC-VF is proved to perform better that Horizon but it also results a very slow scheduling solution, which may cause failed reservation. This is due to the impact of both storage and research operations on voids information on the computational time. In [LTyO02] the LGVF (Least Gap Void Filling) is presented. This algorithm only stores information about the least void interval (the one with largest starting time) improving LAUC-VF performance especially for scheduling time. In [XQLX03] more accurate algorithms are investigated especially from the scheduling time point of view. In particular, Min-SV (Minimum Starting Void) and Min-EV (Minimum Ending Void) are presented in a case without FDLs. Min-SV performs as good as LAUC-VF but provides a scheduling time comparable with the Horizon time and so much better than LAUC-VF. Min-SV performs a bit better than Min-EV but it takes longer to schedule a void interval for an arriving burst. Min-SV selects a void interval between those eligible for a given burst, minimising the time gap between the starting time of the void and the arrival time of the burst. Min-EV minimises the gap between the arrival time of the last bit of the burst and the ending time of the void. For both algorithms void interval information are kept in a balanced binary search tree implemented with pointers. Within this structure burst query, interval insertion and interval deletion operations are carried out. The last two algorithms, Min-SV and MIN-EV result to be the best performing in terms of trade-off between burst loss rate and scheduling time.

5.1.3 Implementation of the scheduling algorithm.

This section presents the proposed void filling implementation algorithm for channel scheduling in OBS networks. The idea on which this algorithm is based is here described together with the characterization of the data structure built to be exploited for a smart organization of the available void intervals, which represents one of the key aspects of the effectiveness in performance of the presented approach.

- **General assumptions.** Although it has been shown that FDLs can be used in OBS networks, in this study a pure loss OBS environment is considered and no storage capabilities exist for bursts at the switching. Hence, if a burst cannot complete the reservation

it is dropped. More specifically, a scenario with QoS management is considered. Three different classes of service are implemented. As mentioned above, different values of extra offset are assigned to bursts belonging to different classes. In particular, extra offset is zero for low priority class, 3*L for the intermediate class and 9*L for the high priority one, where L is the average burst length equally set for each class of service. This choice has been made in order to achieve a good level of class isolation as explained in [MQ98]. The scheduling algorithm here proposed is called HVF (Heap Void Filling) and works as follows. When a setup message is received all wavelengths belonging to the output link are investigated. If at least one wavelength whose horizon time is shorter than the arrival time of the burst is present, the burst is sent on that wavelength. If more of such wavelengths are available, the one that minimizes the time gap between two consecutive packets is chosen. If no wavelength satisfying that time constraint is present, the algorithm looks for one of the eligible void interval previously created and in particular the oldest one (that one with the shortest starting time) is chosen. Clearly, an eligible void means an interval whose starting time is shorter than the burst arrival time and with an ending time that is larger than the one of the tail of the burst itself. The burst is clearly dropped if a suitable interval is not available.

- **Binary heap as data structure.** The data structure used by HVF to treat information about void intervals is a kind of tree represented by a binary heap [FT84]. A binary heap can be intended as a balanced research tree, which allows a partial ordering among elements. The ordering imposes the key of a child to be lower than or equal to the father. The key used to sort the tree is the starting time of the interval. In the root there is the newest interval, in the leafs the oldest. The condition of balanced tree must be kept for each new insertion and so the node has always two sub-trees with the same weight. A binary heap stores information of a single node state an each element of it represents a void interval and carries the following key parameters:

  - the starting time of the void interval (T_a);

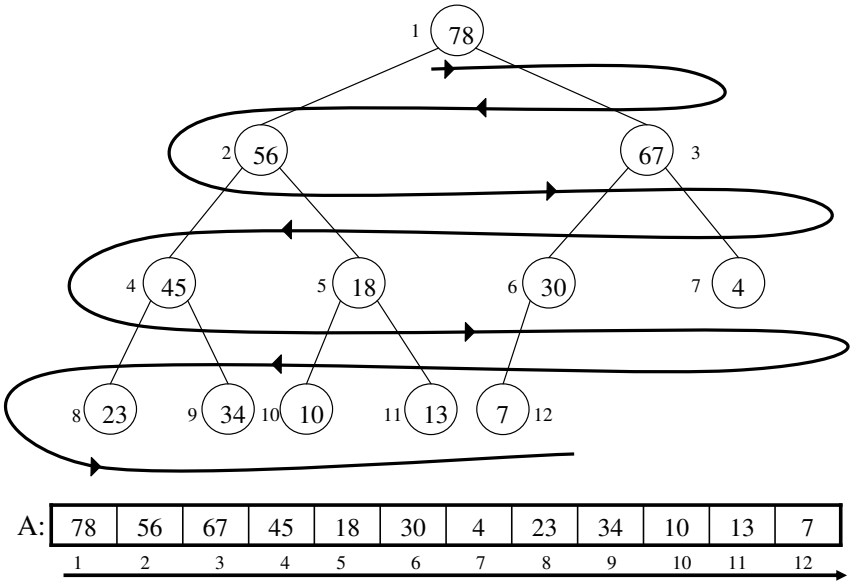

Figure 5.2: Example of a binary heap implemented with array data structure.

- the ending time of the void interval (T_e);
- the wavelength of the considered channel (w);
- the number of void intervals in the whole set of wavelengths belonging to the link (N_i).

As far as the implementation is concerned, the binary heap consists of a vector (or array) whose access is proved to be less time consuming if compared to a pointers-based structure. The drawback of using a vector is that its maximum dimension has to be decided a priori in a static way. In figure 5.2 an example of a binary tree and its implementation on an array is shown. If A is assumed to be the used array, root occupies position A[0] and a generic element i that occupies position A[i] has its left child in position A[2i] whereas the right one is in A[2i+1].

- **Operations within the binary heap and their complexity.** The data structure is assumed to be supported by the three main

operations described in the following:

- *Insertion:* a new void interval is inserted in the last position and its starting time is compared to that of his father: if it is largest the two elements are swapped. Complexity results O(logn) being n the number of elements of the tree.
- *Refresh:* this operation is made when n reaches the maximum array dimension. All intervals whose starting time is shorter than the current time are removed from the tree. After that, a specific function is called in order to update the structure and respect the aforementioned binary heap features. O(nlogn) is the related complexity
- *Search:* among all eligible void intervals, the one with the largest starting time is chosen. As a consequence, the whole tree must be inspected since the used hierarchy only assures a partial ordering. If the interval exists then it is selected and removed from the binary heap, which is updated immediately after. In figure 5.2 the arrow indicates how the search would be done in that specific case. The resulting complexity is also O(nlogn).

The pseudo-code of the HVF algorithm is presented below and describes its behavior when a burst gets to a given node at the current_time instant.

begin{HVF algorithm}
step 1:
for($i = 0; i \leq W; i++$){ /*explore all the W wavelengths of the output link */
search(i); /* search for the wavelength w that minimizes the gap */
}
if(w is found){
if(class!=0){
insertion(new_void); /* Insert the new void */
if(N_i == Array_Max_Dimension)
refresh(current_time);

}
new_time_horizon(w)=Ta+extra_offset+burst_length; /*update time horizon*/
return channel; /* report the selected channel */
}
else goto step 2/* channel not found */

- **Optimization of the search procedure.** The search operation takes longer time than the other two and mostly affects the scheduling time when the ingress load is high because it is the most frequently called procedure with highest complexity. In fact, when the traffic is high (i.e. 80%) bursts belonging to the low priority class especially need to exploit a void interval to make a reservation and so the binary heap requires to be explored very often. An attempt to reduce the overall scheduling time by reducing the search time has been done afterward. This is possible thanks to the implementation of the binary heap on a vector, where a particular position is directly accessed knowing its index. The procedure consists in calculating off-line the sample mean μ and the sample variance S^2 of the position of the array containing the useful void interval by referring to a Gaussian distribution as a direct consequence of the Central Limit Theorem. This assumption is a quite good approximation taking into account both the very high number of bursts typically involved and searches on the vector access the scheduling algorithm has to perform. The search interval [L1, L2] is calculated for each class and each input load per wavelength in a way such that $P(L1 < \mu < L2)$=(1-α). The parameter α that belongs to [0,1], determines the probability to have μ falling in the search interval and it can be seen as an index of approximation. The scheduling algorithm refers then to an interval centered in that position, which results shorter than the one already existing and in any case it contains most of the void values of interest. At that time, the search operation is executed with respect to this shorter interval and the last eligible void interval (if any) is chosen. Reducing the search range of available intervals allows a significant improvement in performance of the scheduling algorithm as exposed in the next section

where simulations results for different values of a are presented.

5.1.4 Numerical evaluations

In this section performance of the scheduling algorithm presented in 5.1.3 are shown. Results have been obtained by means of an ad-hoc, event-driven simulator of the OBS node and simulations have been conducted on a DELL PC with a Pentium 4 CPU (1.3 GHz). The numerical evaluations concern the different contributions to scheduling time that are compared with those of other approaches already known in literature [Tur99][XVC00][LTyO02]. Performance results in terms of burst loss probabilities are also presented. A single switching node characterized by 4 input and output fibers with 32 wavelengths each has been considered. The input traffic is generated according to a Poisson distribution. Burst size is exponentially distributed with average value L equal to 5 Mbit, which corresponds to a burst duration in the range of milliseconds at Gbit/s link speeds. As previously mentioned, three different classes are considered. More specifically, the low priority class, namely class 0, has an extra offset equal to zero, while class 1 and class 2 have an extra offset equal to $3 \cdot L$ and of $9 \cdot L$, respectively. In this work it has been assumed that only voids created with a positive extra offset are considered. The reason for this statement is that class 0 creates short voids compared to those created by class 1 and 2 and their possible exploitation results very modest. It follows that class 0 doesn't create any void and consequently this class makes no insertion operations. Actually, it has been shown that this technique achieves good service differentiation in terms of burst loss probability [XVC00]. The whole traffic is assumed to be equally divided among the three classes. The number of simulated burst considered for testing performance of the proposed algorithm is twenty millions.

5.1.5 Scheduling time for HVF

Let's now focus the attention on the scheduling time of HVF, whose average value is here indicate as T_{sch} and can expressed as follows

$$T_{sch}^{i} = S_i + I_i \tag{5.1}$$

where S_i and I_i represent the average searching time and the average insertion time for the corresponding class of service, which is distinguished

with the apex "i". Figure 5.3 shows the average time taken by the single procedure call. For the void insertion time, only class 1 and 2 are considered since they are the only classes that create voids. Refresh time is included in insertion time given that they are strictly related and, in any case, the contribution of the first one is negligible. The evaluations of the average time have been obtained through the function clock() provided by the Programming Language C - ANSI. In particular this function has been called at the starting time, $Tckstart$ and at the ending time, $Tckend$ of the procedure to be evaluated. The variable events stores the number of times the procedure has been called and so the average duration $Tckaver$ of the procedure is then given by:

$$Tckaver = Tcktotal/events \tag{5.2}$$

Or, if expressed in time units, by:

$$Taverage = tckaver/CLOCK\_PER\_SEC \tag{5.3}$$

where *CLOCK_PER_SEC* is a constant value, which contains the clock period expressed in time units (e.g. seconds). The procedure eval used to perform these evaluation is shown below.

```
/* procedure eval */

tckstart = clock();
procedure();
tckend = clock();
tcktotal += (tckend - tckstart);
events++;
```

By observing figure 5.3, it is quite clear that the search time (15-16μs) is dominant if compared to the insertion time (2-3μs) regardless of the class is considered. This result can be explained taking into account that when the searching procedure is called, the whole binary heap has to be explored independently by which class is considered. As regards the insertion, class 2 is slower than class 1 because its requests are made more in advance and so more steps are needed in order to find the correct position in the binary heap for the new void to be inserted. It is also interesting to note that the trend of the curves is independent of the average traffic load and in any case the overall search time depends on how

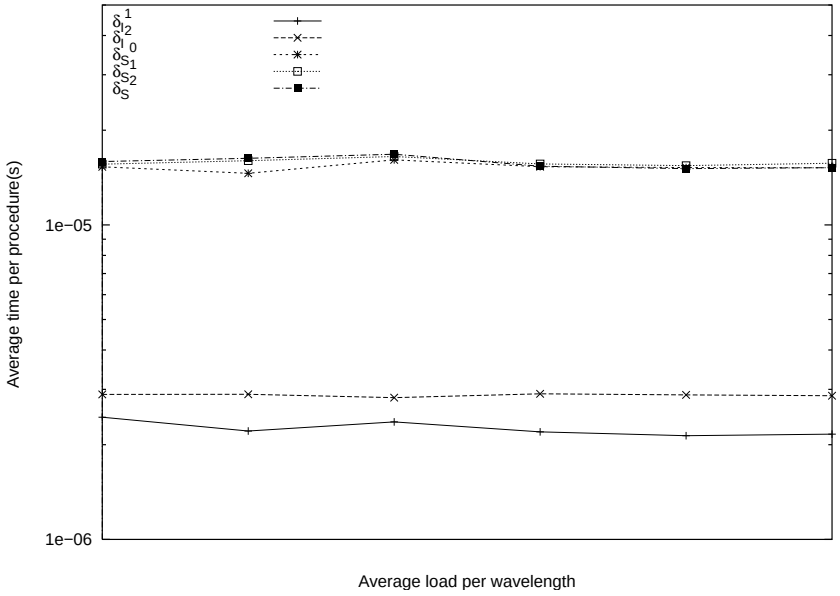

Figure 5.3: Average time in seconds for single insertion and single searching procedures executed by HVF for different classes as a function of the average load per wavelength.

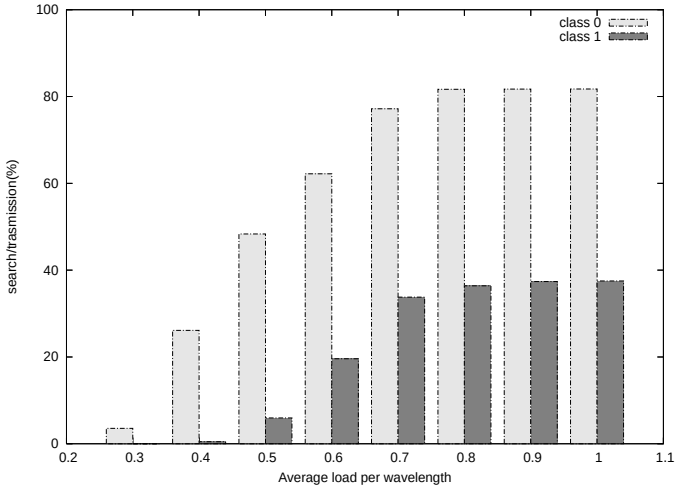

Figure 5.4: Percentage of searching operations for different values of load per wavelength for class 0 and class 1.

many times each class calls the procedure. In figure 5.4 the percentage of search requests as a function of the amount of transmitted data is shown.

As expected, class 0 exploits the search of a suitable void interval to make a reservation much more often than the other two classes and this happens at any load value. Class 2 calls the search procedure a number of times nearly equal to zero and for this reason it has not been considered in the figure. This result is due to the fact that the large extra offset of this class allows to find a wavelength much more in advance and without the need of a void interval. In figure 5.5, T^i_{sch} experienced by HVF is shown for i=1,2,3.

The scheduling time for class 0 coincides with the search time since this class makes no insertions, as could be obtained by multiplying corresponding values of curves depicted in figure 5.3 and 5.4. The graph shows that for class 0 and for low values of traffic load, the average scheduling time is driven by the insertion time, whereas for higher traffic load values the contribution given by the searching operation becomes dominant. This suggests that for low traffic load the usage of the search procedure is limited. Nonetheless, the decreasing trend of the curve for very high loads (e.g.: 0.8 and 0.9) is due to the fact that in this interval losses for class 1 and class 2 become relevant. As a consequence a smaller number

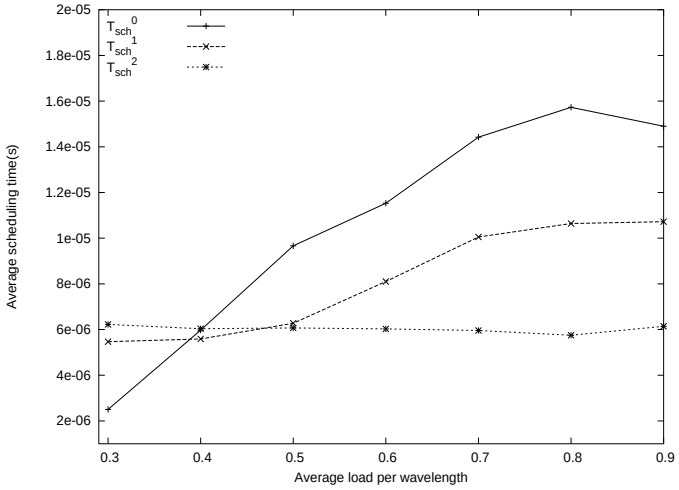

Figure 5.5: Average total scheduling time in seconds for the different classes as a function of the average load per wavelength.

of voids is created and thus a smaller binary heap has to be explored. Moreover, the searching time for class 1 becomes the most important contribution to the average scheduling time when load gets higher. Class 2 experiences an average scheduling time, which remains basically constant, proving again that this class needs to call the searching procedure very rarely.

5.1.6 Reducing scheduling time for HVF

This results confirm that to further reduce the overall scheduling time it can be useful to limit the time for the running of the search procedure, as discussed in 5.1.3. This target can be met by limiting the searching interval thus reducing the portion of the binary heap that is explored to find a suitable void. Figure 5.6 presents the evaluation of the width of the search interval obtained with (1-α) = 68% and the related sample mean of the index containing the void to be used, calculated for different values of the average traffic load per wavelength, with a maximum dimension of the array set at 200.

As expected the more the load grows, the higher is the position of the sample mean (central value) and the wider is the search interval. For

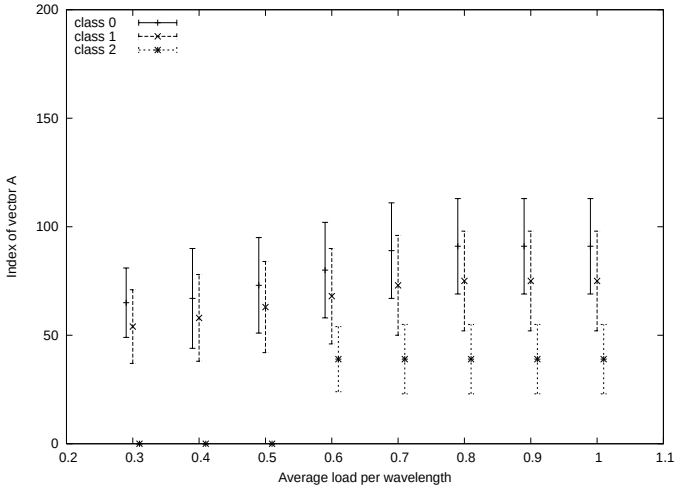

Figure 5.6: Sample mean values and related search interval widths for the three classes of service.

instance, when load is equal to 0.7, class 0 has a search interval centered at the 89th position with width equal to 44. So, the search would start at the (89 - 22) = 67th position of the array and would take the time to explore 44 positions of the array. Simulation results prove that class 2 has always narrowed search intervals and thus shorter searching time if compared to the other classes of service. It is important to remark that the searching time was previously the same for each class since the whole tree had to be explored independently by the class as shown in figure 5.4. Numerical evaluations presented in the following have been obtained with experiments conducted by applying these new concepts in order to test how burst loss probability and scheduling time may change. It is foreseeable that the loss rate would increase whereas scheduling time would clearly improve, since the searching procedure does not explore the whole tree (and so all available information),. Different values of (1-α) are considered (50%, 68%, 90% and 95%) and the resulting performance is compared with the previous case, called complete search (cs) in the following figures. The average load for wavelength is set to 0.8. In figure 5.7 the trend of the overall searching time δ_S^i as a function of a is depicted.

The improvement of the average search time involves all classes significantly. Class 2 experiences a further improvement compared to the

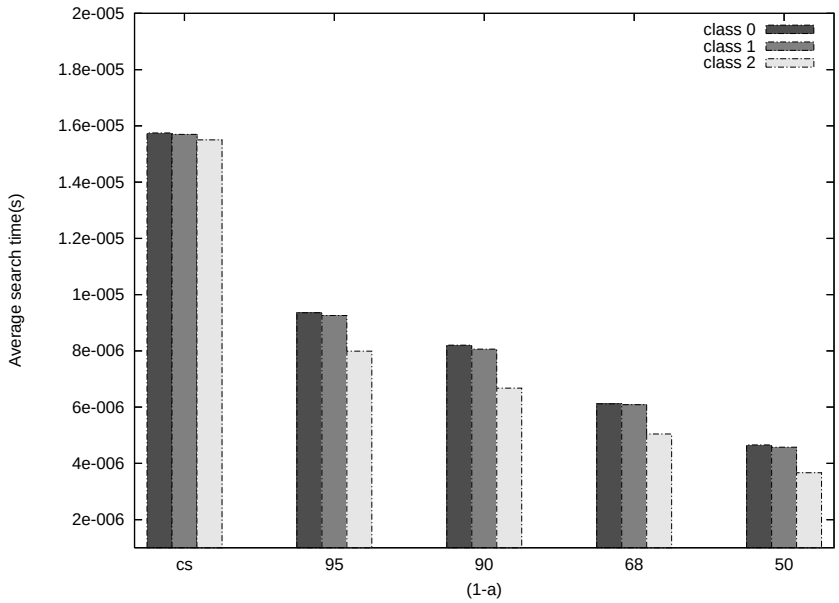

Figure 5.7: Average search time in seconds as a function of a for load per wavelength equal to 0.8.

other two classes because it benefits of a narrower search interval. By denoting with $\Pi^i_{1-\alpha}$ the burst loss probability for class i at (1-α), the effects of different values of (1-α) on this probability are presented in figure 5.8 as a function of the average load per wavelength, being Π^i_{cs} the notation for the complete search performed throughout the array.

As expected the curves of the complete search represents the lower bound. It is possible to observe how loss probability increases very slightly maintaining the same order of magnitude independently of α. Actually, this represents an appreciable result together with the gain obtained for the scheduling time by considering that the only price to pay is in terms of a limited worsening for the loss probability. Figure 5.9 shows the overall average scheduling time for the new search time for different values of (1-α). It is possible to note the improvement that can be obtained by reducing the running time for the search procedure. This improvement involves basically only class 0 and class1 since, as already said, class 2 nearly does not use search procedure at all.

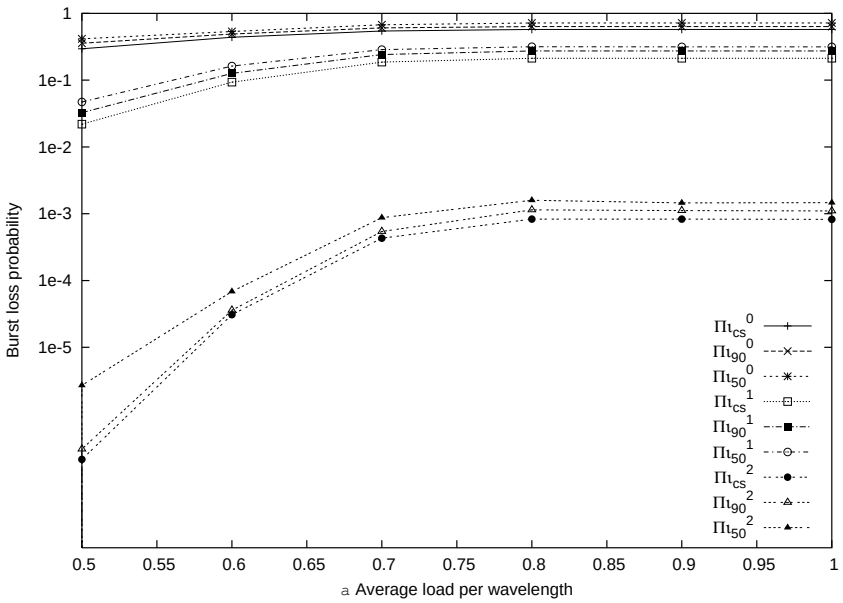

Figure 5.8: Burst loss probability as a function of load varying a as a parameter.

5.1.7 HVF compared to others solutions

Last results make the comparison of HVF with other algorithms already presented in literature and mentioned in section 5.1.2. All these algorithms have been tested on the node configuration taken in consideration in this study. In figure 5.10, burst loss probability of HVF compared with Horizon and LGVF algorithms is shown. For a matter of clarity LAUC has not been included in the figure and however it is outperformed by LGVF. HVF actually outperforms the others for each class in the range of the load value considered for the evaluation. In figure 5.11 the scheduling time T_{sch} averaged over all classes is plotted for different algorithms confirming the good achievement of HVF in terms of trade off between processing time and burst loss performance. As it can be seen HVF gets closer to the fastest scheduling algorithms LGVF and Horizon (especially HVF with (1-α) =50%) but performing much better in terms of burst loss probability.

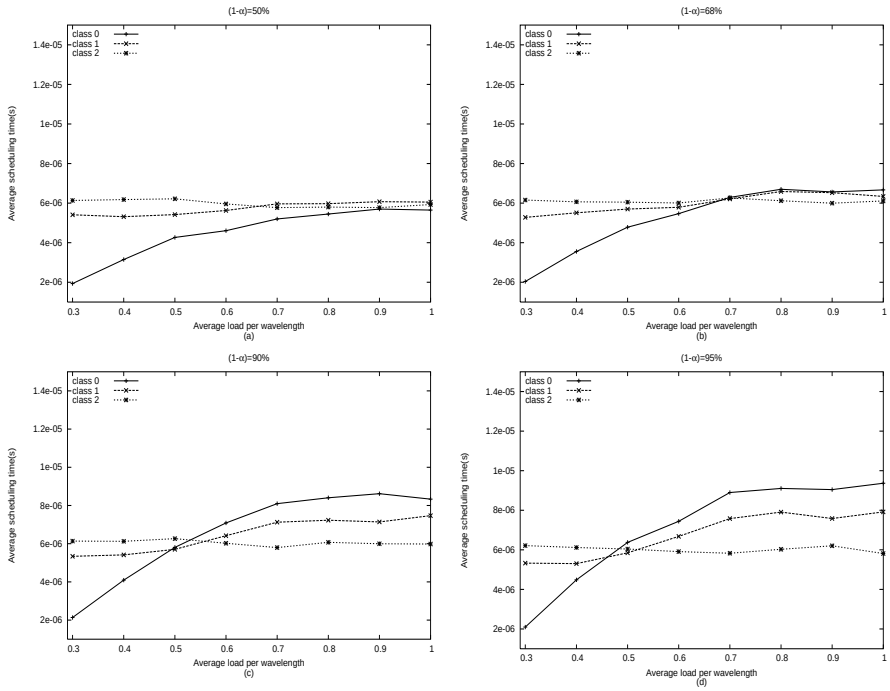

Figure 5.9: Average scheduling time in seconds as a function of the load per wavelength and different search interval widths (a,b,c,d).

5.1.8 Comments

In this work the problem of channel reservation in an OBS node has been investigated. A new implementation of the void filling scheduling algorithm, HVF, that tries to optimize performance in terms of both burst loss probability and scheduling time has been presented. To achieve the target a binary heap implemented by means of an array data structure is exploited to store the information related to void intervals created in the presence of QoS differentiation. Several numerical results presented in the work evidence the improvement obtained with respect to other solutions. In particular the proposed implementation allows void filling to be performed within the temporal target of typical offset values.

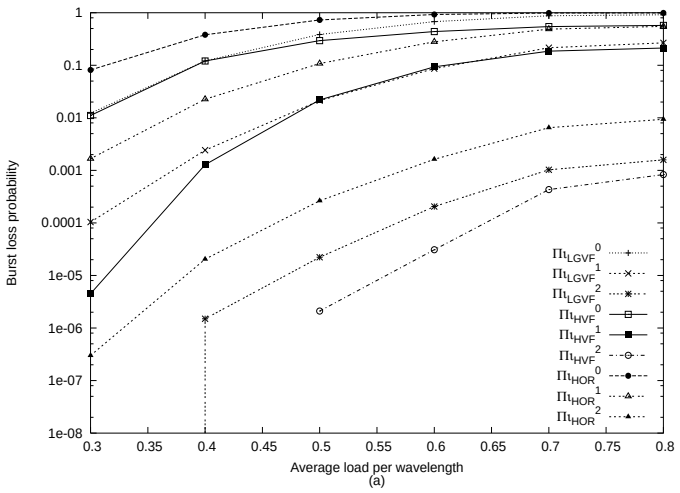

Figure 5.10: Burst loss probability for HVF, LGVF and Horizon algorithms as function of the average load per wavelength (a).

5.2 Traffic and performance analysis of optical packet/burst assembly with self similar traffic

In this chapter analytical and simulative study of optical packet/burst assembly in the presence of self similar input traffic is presented. The influence of the main assembly parameters is studied by simulation for timer and size-based aggregation strategies. Analytical model is proposed to represent the average traffic on optical link with the aim to evaluate system performance. Comparisons with simulation prove that the model is well suited to catch the loss system behavior. Optical packet-switched (OPS) and optical burst-switched (OBS) networks have been considered with growing interest in the last decade as a long and medium-term solution for core networks to carry the expected increased traffic generated by high capacity local and metropolitan area networks. Many papers presented technological issues and discussed the potential benefits of the adoption of optical burst and packet switching using the huge bandwidth of the DWDM transport with fine granularity and considerable flexibility [HA00][OSHT01][DDC+03][QY99][CCXV99][GBPS03]. In order to alle-

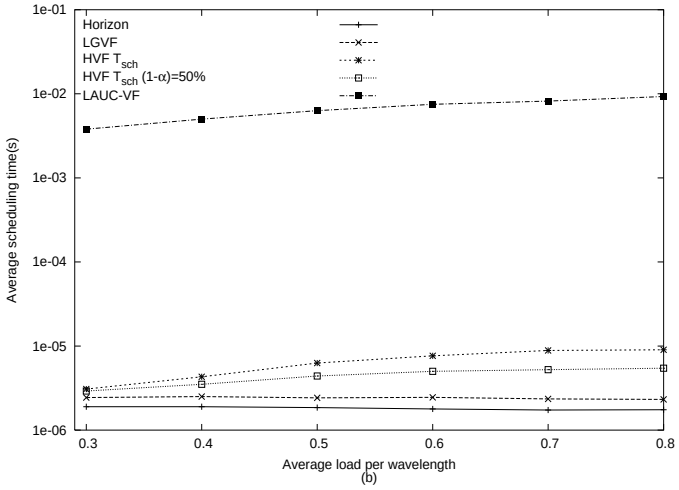

Figure 5.11: Scheduling time in seconds for HVF, HVF $(1-\alpha)$=50, LAUC-VF, LGVF and Horizon algorithms (b).

viate the switching overhead in the high-speed optical switch, both OPS and OBS apply a large data frame for data transmission. Correspondingly, the edge node has to first classify the data traffic coming from the client networks (Ethernet, IP,) into different forward equivalent classes (FEC), and then assemble data of the same FEC in optical data frames. The assembly procedures can substantially change the traffic characteristics so and have an significant impact on the network performance, which will be closely looked at in this paper. Since the assembly function means the same thing for OPS and OBS, for brevity we do not distinguish them in the following context unless otherwise indicated explicitly. The optical data frame of OPS/OBS will be referred to as optical burst uniformly. A number of publications have been contributed to the traffic characterization and performance impact of the burst assembly. The statistics for the size and interarrival time of optical bursts from the assembly are studied in [Lea02][dVRG04]. The impact of the assembler on the self-similarity of the data traffic is inspected in [XY02][HDG03]. [IA02] discusses performance issues with respect to blocking probability, and discovers that the Poisson approximation of the optical burst traffic provides an upper bound for blocking probability. In this work, through extensive simulation and analysis, the performance impact of different

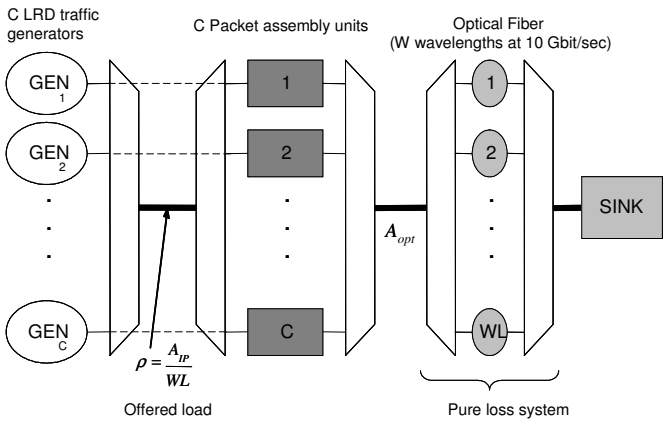

Figure 5.12: The burst assembly system

assembly parameters is inspected. We find the performance behavior in the edge node can be well captured by an on-off model of the optical burst traffic in the typical network operation scenarios.

5.2.1 System and traffic model

The system under study is the assembly process at the optical network edge which is sketched in figure 5.12. It is composed of three main blocks: traffic generators, optical burst assembly units and DWDM optical network link. Totally C traffic generators are used, with each of them generating aggregate self-similar IP traffic for one FEC class according to M/Pareto model as applied in [HDG03]. With M/Pareto model, IP packets are segmented from data transmission sessions that arrive following a Poisson process and have the session size distribution conforming to a heavy-tailed Pareto distribution. The main parameters characterizing an M/Pareto traffic model include:

- IP_{max}: the maximum length of the IP packets used to fragment each data session, for our purpose 1000 Bytes;
- PpF: the mean session size normalized by the maximum IP packet length;
- H: the Hurst parameter that represents the degree of self-similarity

and from which the shaping parameter of the Pareto distribution can be derived. In the simulations, H is set to 0.7.

- Ba: the access link speed (100Mbps), which determines the interval between back-to-back packets of the same session.
- A_{IP}: the total offered traffic of IP traffic in the unit of Erlang.

At the stage of burst assembly, IP packets are classified according to their destination address and QoS class and distributed into correspondent assembly queues. With respect to the assembly schemes, we suppose there is always a timer bounded to an FEC assembly queue to constrain the assembly delay. As for the burst size, two cases are distinguished: unbounded size and fixed size. With unbounded size there is no padding overhead, while fixed burst length can bring some efficiency in performance and implementation. The following parameters are defined for the burst assembly:

- $TOF = \frac{t_{out}}{IAT}$ time out factor, given by the ratio between the assembly time out tout and the mean packet inter-arrival time of each $FECIP$ flow (IAT). In our simulations, each $FECIP$ flow has the same IAT.
- $BSF = \frac{BS}{IP_{max}}$ burst size factor, given by the ratio between the fixed burst size (BS) and the maximal IP packet size.

The third stage is a model of a WDM transmission link with w wavelengths and 10 Gbps per wavelength.

5.2.2 Traffic characterization and performance analysis

To analyze the loss probability in the edge node, the third stage can be modeled by a pure loss system where multiple servers represent the wavelength channel bundle. Correspondingly, the number of servers equals the number of wavelengths w. The incoming traffic to the loss system is multiplexed by departure flows from the C assembly queues. The offered load to the system A_0 equals to A_{IP} in the case of unbounded burst size. If the fixed burst size is used, A_0 is greater than or equal to A_{IP}

since padding can be added by the assembly. Here, the filling of burst depends on the relation between TOF and BSF. Following asymptotic operational regions can be investigated:

- $BSF >> TOF$ where the assembly time out dominates the assembly of burst;
- $BSF << TOF$ where all bursts leave because they are full.

In the first case, the average number of IP packets in a burst can be derived as $n = \lambda_{IP,FEC} \cdot t_{out} + 1$ where $\lambda_{IP,FEC} = \frac{1}{IAT}$ [Gau03]. This leads to $n = TOF + 1$ directly. Therefore, it holds that:

$$A_0 = C \cdot \frac{\lambda_{IP,FEC}}{TOF + 1} \cdot \frac{BSF \cdot IP_{max}}{B} \quad for \quad TOF << BSF \tag{5.4}$$

Here, B is the service capacity of one server and equal to 10 Gbps. In the second case, since the bursts are mostly full, there is $A_0 \approx A_{IP}$ and $n \approx BSF$. Then approximately

$$A_0 = C \cdot \frac{\lambda_{IP,FEC}}{BSF} \cdot \frac{BSF \cdot IP_{max}}{B} \quad for \quad TOF >> BSF \tag{5.5}$$

It is now useful to find the intersection point of the two operational regions by equating expression 5.2.2 and 5.2.2. It results in $BSF = TOF + 1$ that can be approximately considered as the delimiting operational point where most assembled bursts turn to be completely filled. As A_0 and number of servers w are available, Erlang-B formula can be applied to calculate the burst loss probability as an upper bound [IA02]. However, this is generally too conservative for small and medium number of FEC classes. Actually, as long as the load contribution from each FEC flow, i.e., A_0/C is less than 1 and the aggregation degree of the assembly is large (i.e., with large TOF or BSF), it becomes unlikely that a burst inter-departure time from an assembly queue is smaller than the burst transmission time of the foregoing burst. So, the optical burst traffic of each FEC can be modeled by a fluid on-off flow with the ON period corresponding to the transmission time of one burst on the wavelength channel. Traffic rate in each ON period is constant and equal to

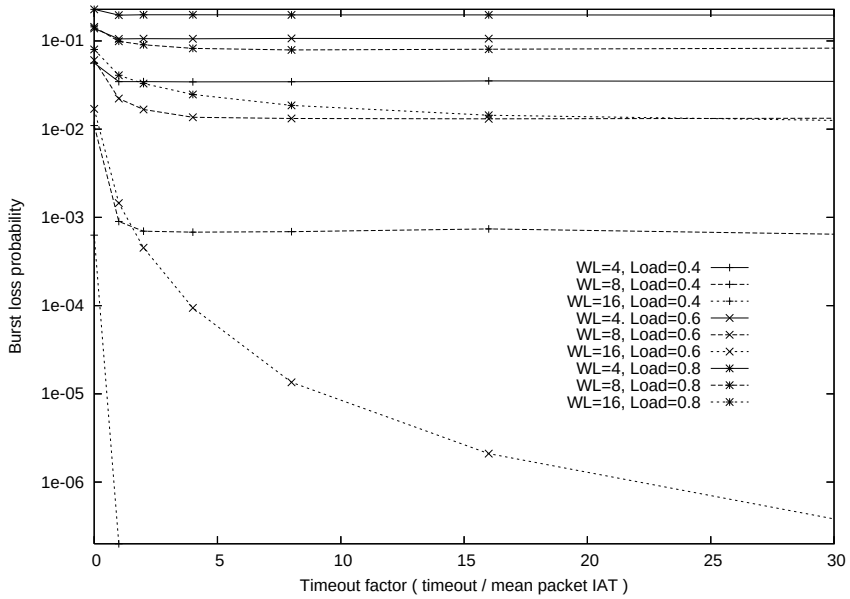

Figure 5.13: Burst loss probability as a function of TOF for different values of the offered load and number of wavelength channels, $C = 10$

the transmission rate of a wavelength channel. The interarrival time of ON-periods corresponds to the burst inter-departure time from an assembly unit. For self-similar IP traffic, packets tend to arrive in clusters (Joseph effect) [LWTW93]. As a result, the departure burst traffic is likely to have large burst size (pure time-out assembly) or clusters of fixed burst size with small interarrival time (assembly with fixed burst size). Therefore, a general on-off traffic model [Kel96] can be applied, which is parameterised by two parameters p and r. p is the proportion of time spent in the ON period, which is equal to A_0/C. r is the constant traffic rate in the ON period. The loss probability of the aggregated traffic multiplexed by C such on-off flows can be calculated according to the method of effective bandwidth (Equation 2.9 and 3.11 in [Kel96]).

5.2.3 Performance evaluation

The results of performance evaluation will be given in this section.

In figure 5.13 performance of burst traffic assembled by algorithm

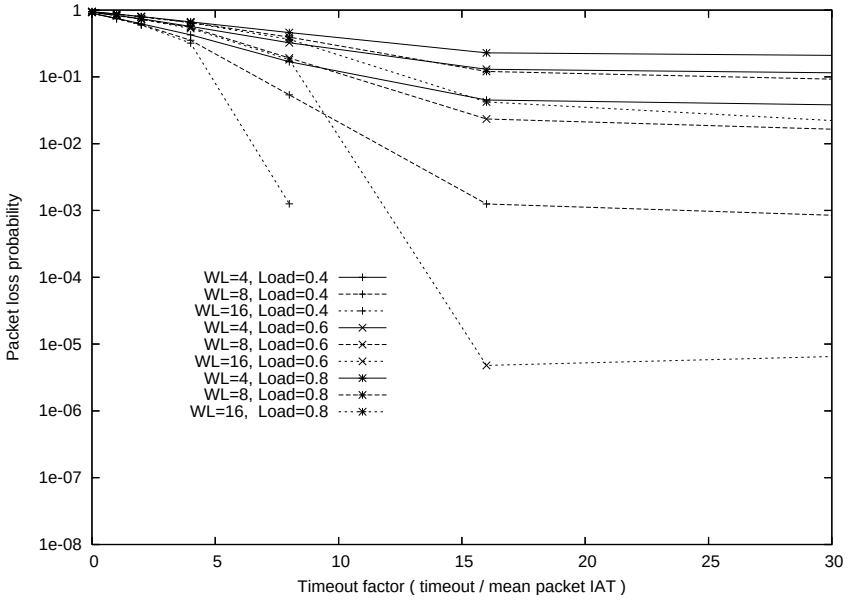

Figure 5.14: Burst loss probability as a function of the TOF for different values of the offered load and number of wavelength channels, $C = 10$, $BSF = 16$

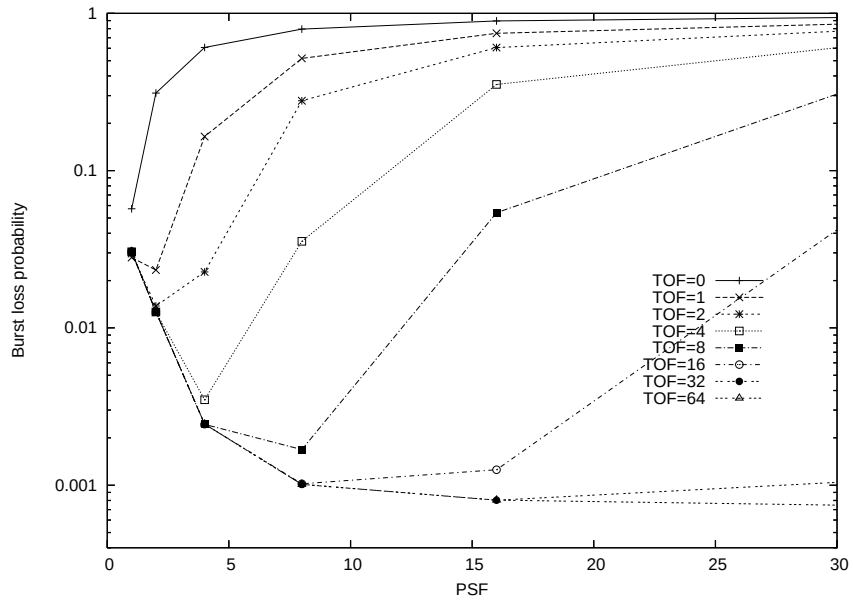

Figure 5.15: Burst loss probability as a function of the BSF varying the TOF as a parameter; $C = 10$, $w = 8$, $\rho = 0.4$

based on the pure time out strategy is presented. $C = 10$ here. It can be seen how the aggregation process can improve the performance in terms of burst loss probability. In most cases, the loss probability decreases at the beginning fast with the increase of timeout and then becomes stable. For the cases of 16 w (load=0.4 and load=0.6), the number of wavelengths are greater than the number of burst flows. At the same time, the traffic load contributed by each flow is less than 1 (0.64 and 0.96 respectively). In case the number of servers is larger than the number of on-off flows and the peak rate of the on-off traffic is equal to the service rate, the loss probability turns out to be 0. Therefore, with increasing timeout each burst flow asymptotically degrades to an on-off flow and the loss probability goes down continuously to zero. Also note that in the case of $w = 16$ and $load = 0.8$, the load on each flow is 1.28 and the burst flow cannot be modeled as on-off flow any more. In figure 5.14 the burst size is fixed. Loss probability is plotted as a function of the TOF, and when $TOF < BSF$ performance gets better with TOF due to increasing aggregation and a better filling efficiency, whereas when

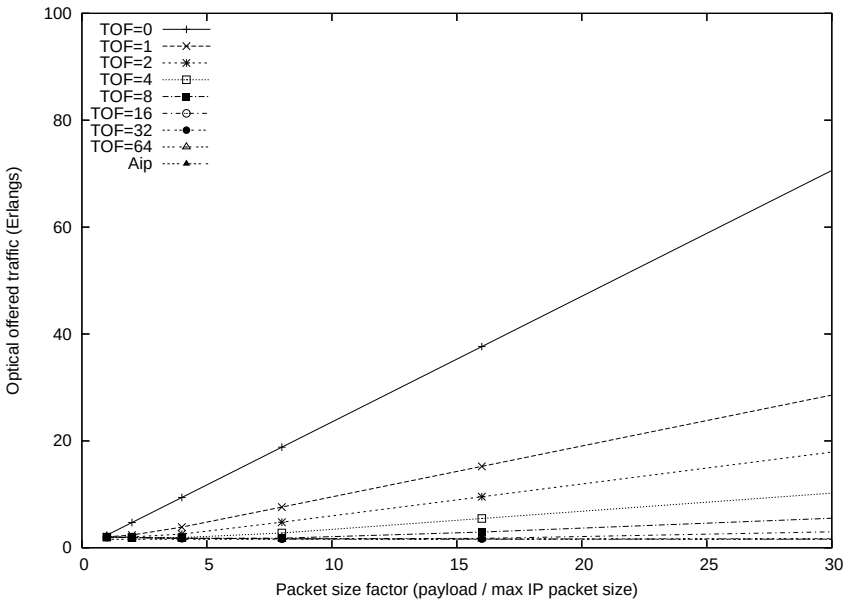

Figure 5.16: Optical offered load A_0 as a function of BSF varying TOF. A_{IP} is also shown as a reference.

$TOF > BSF$ the assembly algorithm works almost always by payload filling and we could assume that time out never expires. In the latter case the assembly procedure is not affected by TOF and performance saturates at values depending on BSF (lower values for higher BSF). This comparison shows that there is an optimum design choice for TOF and BSF, i.e. when TOF is about equal to BSF loss probability has a minimum. This is very close to the delimiting operational point derived in section 5.1.3. To better explain this we show the burst loss probability as a function of TOF and BSF with BSF and TOF as parameters, respectively. Figure 5.15 shows the burst loss behavior as a function of the BSF. It is clear that the optimum of each curve falls around the point of $BSF = TOF + 1$. The curve $TOF = 64$ has the minimal loss probability among all due to resulting largest aggregation level. The minimum loss probability in figure 5.15 is very close to that of the correspondent curve ($w = 8, Load = 0.4$) in figure 5.13. This indicates that same performance model can be applied for both unbounded burst size and fixed size schemes as long as the aggregation level is large enough.

Similarly, for the reason of increasing aggregation level, most curves decrease first with the increasing BSF. Beyond the optimum point, the impact of the padding overhead begins to be serious. This is confirmed by figure 5.16 that represents the offered traffic A_0 as a function of the BFS varying the TOF. For given TOF, A_0 increases with BSF, which can lead to higher loss probabilities as shown in figure 5.15. In figure 5.17, the analytical results of the loss probability are compared with the simulation results with respect to different offered load. Here $C = 10$ and $w = 8$. For simulation only the unbounded burst size case is considered and $TOF = 8$. From figure 5.13 it can be seen that at $TOF = 8$ the loss probability already converges, so it is a large enough aggregation level for the application of on-off source model. For analysis, we consider the Erlang-B formula and the effective bandwidth method based on the multiplexing of on-off model as described in section 5.1.3. It is seen that for small loss probability, the effective bandwidth provides quite good estimations. However, for large loss probability it works not well. This is because that effective bandwidth method is developed on the basis of large deviation theory which is oriented for the estimation of rare events. In the real application where the goal loss probability is in the order of 10-4, effective bandwidth method can play an important role.

5.2.4 Final remarks

In this work the burst traffic characteristic and burst loss probability in the edge node of OBS/OPS networks with self-similar IP traffic were studied. The basic system behaviour of the assembly is analyzed and simulated. The relative relation between timeout and fixed burst size is discovered which can be used as a reference for the optimal system design. We propose the multiplexing of on-off sources to model the optical burst traffic. This model can not only explain the simulative system behaviour very well, but also lead to tight estimation of loss probability in the practical interested area.

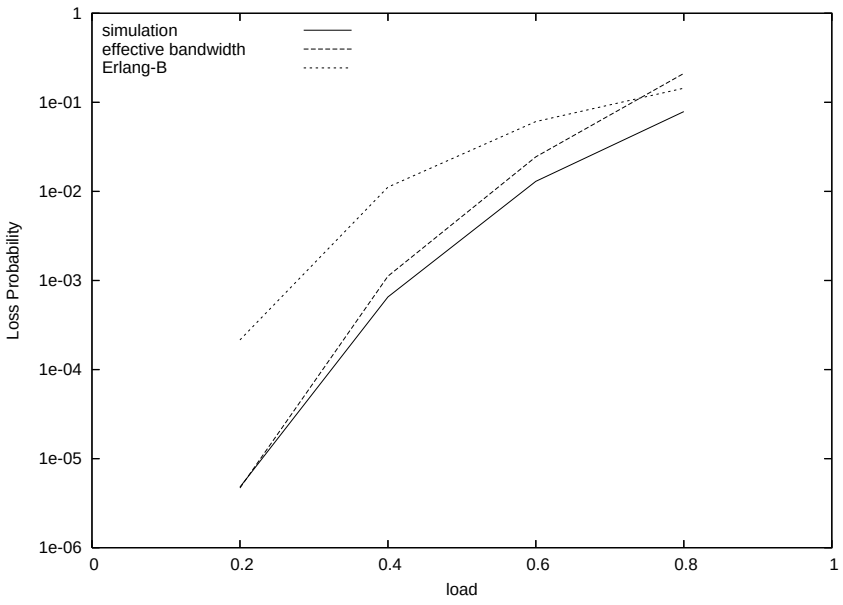

Figure 5.17: Comparison of analysis results as a function of the offered load for $C = 10$, $w = 8$, $TOF = 8$

Chapter 6

Conclusions

Photonic is the most promising technology to meet the ever increasing demand of bandwidth of the last years. The research on this field aims to find efficient solutions in order to exploit the huge capacity offered by optical fibers. From the network point of view Optical Packet Switching is the most relevant one in terms of performance even though it requires high technology at component level. Although many good results have been achieved recently, the technology for optical packet switching is not ready yet and this makes OPS a long-medium term solution for the next-generation backbone networks. OPS is the topic studied in this thesis. In particular packets here are assumed variable length and working asynchronously. The DWDM technique is also assumed since it allows to multiply the available capacity by multiplexing all the channels within a fiber, each one characterized by a different wavelength. OPS needs to be optimized both at node and network level. A single node approach permits to study algorithms to solve contention when packets are directed to the same resources at the same time. Such algorithms could not preserve the right sequence of the packet stream. Keeping packet sequence would be very much appreciated instead since out of order packets can impact harmfully on transport level performance. Some algorithms can be designed with the intent of keeping the sequence and simulations results showed that this can be done basically maintaining the same performance. It is also important to evaluate possible solution for switch architecture in order to design feasible switches that don't require too complex schemes from the technology point of view. The

multi-fiber scheme was taken into account in a cost and conversion saving perspective. By repeating the same set of wavelengths within an interface it is possible to work and maintain the same performance with less tunable wavelength converters. From the network perspective the routing problem is fundamental to achieve flexibility, efficiency and reliability. In this thesis the adaptive routing as alternative to static routing has been discussed. Such dynamic routing take its forwarding decision with a knowledge of the current state of the network and thus the node has the opportunity to avoid temporary congestion situations. Moreover adaptive routing can come in help to achieve quality of service and protection from failures which are fundamental requirements in a network. Another issue to take care of is the packet assembly. This operation is done at the edge node of the network where incoming electronic IP packets from local and metropolitan networks are first converted to optical domain and then organized in optical units, called bursts, that will be sent to the core node of the network. Constraints on time and on the burst length play a crucial role in this sense and they must be designed carefully.

Bibliography

[AJVC04] N. Andriolli, T. Jakab, L. Valcarenghi, and P. Castoldi. Separate wavelength pools for multiple-class optical channel provisioning, telecommunications network strategy and planning symposium. page 379 384, June 2004. Networks, IEEE.

[aN06] B. Mukherjee andF. Neri. Report of us/eu workshop on key issues and grand challenges in optical networking, June 2006. available at http://networks.cs.ucdavis.edu/mukherje/US-EU-wksp-June05.html.

[BBC+98] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and Z. W. Weiss. An architecture for differentiated services, 1998. RFC 2475.

[BPS99] J. C. R. Bennett, C. Patridge, and N. Shectman. Packet reordering is not a pathological network behavior. *IEEE/ACM Transactions on Networking*, 7(6):789–798, December 1999.

[BSS95] K. Bala, T. Stern, and K. Simchi. Routing in linear lightwave networks. *IEEE/ACM Transection on Networking*, 3:459–469, August 1995.

[C+01] D. Chiaroni et al. First demonstration of an asynchronous optical packet switching matrix prototype for MultiTerabit-class Routers/Switches. October 2001. Proceedings of ECOC 2001, Amsterdam, The Nederlands.

[Cal00] F. Callegati. Optical buffers for variable length packets. *IEEE Communications Letters*, 4(9):292–294, September 2000.

[CC01] F. Callegati and W. Cerroni. Wavelength selection algorithms in optical buffers. June 2001. Proceedings of IEEE ICC 2001, Helsinki, Finland.

[CCC01] F. Callegati, W. Cerroni, and G. Corazza. Optimization of wavelength allocation in WDM optical buffers. *Optical Networks Magazine*, 2(6):66–72, November 2001.

[CCC+04] F. Callegati, D. Careglio, W. Cerroni, J. Sole-Pareta, C. Raffaelli, and P. Zaffoni. Keeping the packet sequence in optical packet- switched networks. June 2004. Proceedings of 9th European Conference on Networks and Optical Communications, Eindhoven, The Netherlands.

[CCM+04] F. Callegati, W. Cerroni, G. Muretto, C. Raffaelli, and P. Zaffoni. Adaptive routing in DWDM optical packet switched network. pages 71–86, February 2004. in Proc.eedings of ONDM 2004, Gent, Belgium.

[CCR02] F. Callegati, G. Corazza, and C. Raffaelli. Exploitation of DWDM for optical packet switching with quality of service guarantees. *IEEE Journal on Selected Areas in Communications*, 20(1):190–201, January 2002.

[CCRZ03] F. Callegati, W. Cerroni, C. Raffaelli, and P. Zaffoni. Dynamic wavelength assignment in MPLS optical packet switches. *Optical Network Magazine*, 4(5):41–51, September–October 2003.

[CCRZ04] F. Callegati, W. Cerroni, C. Raffaelli, and P. Zaffoni. Wavelength and time domains exploitation for QoS management in optical packet switches. *Computer Networks, QoS in Multiservice IP Networks*, 44(4):569–582, March 2004.

[CCXV99] F. Callegati, H.C. Cankaya, Y. Xiong, and M. Vandenhoute. Design issues of optical ip routers for internet backbone applications. *IEEE Communications Magazine*, 37(12):124–128, December 1999.

[CGK99] I. Chlamtac, A. Ganz, and G. Karmi. Lightpath communications: An approach to high bandwidth optical WAN's. *IEEE/ACM Transection on communication*, 40:1171–1182, July 1999.

[CLR90] T.H. Cormen, C.E. Leiserson, and R.L.Rivest. *Introduction to algorithms, McGraw-Hill, MIT Press*, 1990.

[CM05] M. Casoni and M. L. Merani. On the performance of tcp over optical burst switched networks with different qos classes. February 2005. Proceedings QoSIP 2005, Catania, Italy.

[CMR+04] F. Callegati, G. Muretto, C. Raffaelli, P. Zaffoni, and W. Cerroni. A framework for the analysis of delay jitter in optical packet switched networks. October 2004. Proceedings of the IFIP 1st Optical Networks and TEchnologies Conference OpNeTec, Pisa, Italy.

[Cos03] *Advanced Infrastructure for Photonic Networks - Extended final report of COST action 266*, 2003.

[CTT99] G. Castanon, L. Tancevsky, , and L. Tamil. Routing in all-optical packet switched irregular mesh networks. pages 1017–1022. in Proceedings of IEEE Globecom 1999, December 1999.

[DDC+03] L. Dittmann, C. Develder, D. Chiaroni, F. Neri, F. Callegati, W. Koerber, A. Stavdas, M. Renaud, A. Rafel, J. Sole-Pareta, W. Cerroni, N. Leligou, L. Dembeck, B. Mortensen, M. Pickavet, N. Le Sauze, M. Mahony, B. Berde, and G. Eilenberger. The european IST project DAVID: A viable approach towards optical packet switching. *IEEE Journal on Selected Areas in Communications*, 21(7):1026–1040, September 2003.

[DGSB01] K. Dolzer, C. Gauger, J. Spath, and S. Bodamer. Evaluation of reservation mechanisms for optical burst switching. *AEU Int. Journal of Electronics and Communications*, 55(1), 2001.

[DHS98] S. L. Danielsen, P. B. Hansen, and K. E. Stubkjaer. Wavelength conversion in optical packet switching. *Journal Lightwave Technology*, 16(9):2095–2108, September 1998.

[dVRG04] M. de Vega Rodrigo and J. Gtz. An analytical study of optical burst switching aggregation strategies. October 2004. The Third International Workshop on Optical Burst Switching (WOBS), San Jose, CA, USA.

[ELS06] V. Eramo, M. Listanti, and M. Spaziani. Resources sharing in optical packet switches with limited-range wavelength converters. *Journal of lightwave technology*, 23(2):671–687, February 2006.

[Fre80] A.A. Fredericks. Congestion in blocking system, a simple approximation technique. *Bell System Technical Journal*, 59:805–826, 1980.

[FT84] M.L. Fredman and R.E. Tarjan. Fibonacci heaps and their uses in improved network optimization algorithms. *Foundations of Computer Science, 25th Annual Symposium on*, pages 338–3461, October 1984.

[FV00] A. Fumagalli and L. Valcarenghi. Ip restoration vs. wdm protection: Is there an optimal choice? *Network, IEEE, Texas University, Dallas, TX, USA*, 14(6):43–41, November - December 2000.

[G+98] C. Guillemot et al. Transparent optical packet switching: The european ACTS KEOPS project approach. *IEEE/OSA Journal of Lightwave Technology*, 16(12):2117–2134, December 1998.

[Gau03] C.M. Gauger. Trends in optical burst switching. *Proceedings of SPIE ITCOM, Orlando, USA*, 2003.

[GBPS03] C.M. Gauger, H. Buchta, E. Patzak, and J. Saniter. Performance meets technology - anintegrated evaluation of obs nodes with fdl buffers. *Proceedings of International Workshop on Optical Burst Switching WOBS, Dallas, TX*, October 2003.

[GRG+98] P. Gambini, M. Renaud, C. Guillemot, F. Callegati, I. Andonovic andB. Bostica andD. Chiaroni, G. Corazza andS.L. Danielsen, P. Gravey andP. B. Hansen, M. Henry, C. Janz, A. Kloch, R. Krahenbuhl, C. Raffaelli, M. Schilling, A. Talneau, and L. Zucchelli. Transparent optical packet switching: Network architecture and demonstrators in the KEOPS project. *IEEE Journal on Selected Areas in Communications*, 16(7):1245–1259, September 1998.

[HA00] D. K. Hunter and I. Andonovic. Approaches to optical internet packet switching. *IEEE Communications Magazine*, 38(9):116–122, September 2000.

[HCA98] D. K. Hunter, M. C. Chia, and I. Andonovic. Buffering in optical packet switches. *IEEE/OSA Journal of Lightwave Technology*, 16(12):2081–2094, December 1998.

[HDG03] Guoqiang Hu, Klaus Dolzer, and Christoph M. Gauger. Does burst assembly really reduce the selfsimilarity? April-May 2003. OFC 2003, Atlanta, USA.

[IA02] M. Izal and J. Aracil. On the influence of self-similarity on optical burst switching traffic. *Proceedings of IEEE Globecom, Taipei, Taiwan*, 3:2308–2312, Novemeber 2002.

[JID+03] S. Jaiswal, G. Iannacone, C. Diot, J. Kurose, and D. Towsley. Measurement and classification of out-of-sequence packets in a tier-1 ip backbone. *Proc. of INFOCOM 2003, San Francisco, USA*, 2:1199–1209, March 2003.

[Kel96] F. Kelly. Notes on effective bandwidths. stochastic networks: Theory and applications. pages 141–168, 1996. Oxford University Press.

[KR02] K. Kompella and Y. Rekhter. LSP hierarchy with generalized MPLS TE. Internet Draft draft-ietf-mpls-lsp-hierarchy-08.txt, September 2002.

[Lea02] Leavens. Traffic characteristics inside optical burst switched networks, 2002. Opticomm, Boston.

[LG02] M. Laor and L. Gendel. The effect of packet reordering in a backbone link on application throughput. *IEEE Network*, 16(5):28–36, September 2002.

[LS00] L. Li and A.K. Somani. A new analytical model for multifiber wdm networks. *Selected Areas in Communications*, 18(10):2138–2145, October 2000.

[LTyO02] K. Long, R. S. Tucker, and Se yoon Oh. Fairness scheduling algorithms for supporting qos in optical burst switching networks. *Proceedings of the SPIE's International Symposium: Aia-Pacific Optical and Wireless Communications APOC, Shanghai, CHINA*, 2002.

[LWTW93] W.E. Leland, W. Willinger, M.S. Taqqu, and D.V. Wilson. *On the self-similar of Ethernet traffic*, 1993. ACM SIGComm.

[Man03] E. Mannie. Generalized multi-protocol label switching GMPLS architecture. Internet Draft draft-ietf-ccamp-gmpls-architecture-04.txt, February 2003.

[McC85] E. McCreight. Priority search trees. *SIAM J. Computing*, 14(2):257–276, October 1985.

[MQ98] M.Yoo and C. Qiao. A new optical burst switching protocol for supporting quality of service. *in SPIE Proceedings, All Optical Networking: Architecture, Control and Management Issue*, 3531:396–405, November 1998.

[MR06a] G. Muretto and C. Raffaelli. Contention resolution in multi-fibre optical packet switches. February 2006.

COST291/GBOU ONNA Workshop on Design of Next Generation Optical Networks: from the Physical up to the Network Level Perspective, Gent, Belgium.

[MR06b] G. Muretto and C. Raffaelli. Performance evaluation of asynchronous multi-fibre optical packet switches. *Proceeding of Optical Networks Design and Modeling ONDM, Copenhagen, Denmark*, May 2006.

[NW96] D. Norte and E. Willner. All-optical data format conversions and reconversions between the wavelength and time domains for dynamically reconfigurable wdm networks. *Journal of lightwave technology*, 14(6), June 1996.

[Odl00] A. Odlyzko. Internet growth: myth and reality, use and abuse. *Information Impacts Magazine*, November 2000.

[OMSA98] H. Obara, H. Masuda, K. Suzuki, and K Aida. Multifiber wavelength-division multiplexed ring network architecture for tera-bit/s throughput. *Proceedings ICC*, 2:921–925, June 1998.

[OSHT01] M.J. O'Mahony, D. Simeonidou, D.K. Hunter, and A. Tzanakaki. The application of optical packet switching in future communication networks. *IEEE Communications Magazine*, 39(3):128–135, March 2001.

[QY99] C. Qiao and M. Yoo. Optical burst switching: A new paradigm for an optical internet. *Journal of High Speed Networks*, 8(1):69–84, January 1999.

[QY00] C. Qiao and M. Yoo. Choices features and issues in optical burst switching. *Optical Networks Magazine, 12*, 12:36–44, May 2000.

[RM02] R. Ramamurthy and B. Mukherjee. Fixed-alternate routing and wavelength conversion in wavelength-routed. *IEEE/ACMTransaction on Networking*, pages 351–367, June 2002.

[RS] R. Ramaswani and K. N. Sivarajan. *Optical Networks, a pratical prespective*. Morgan Kaufmann Publishers. ISBN 1-55860-445-6.

[RVC01] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol label switching architecture. IETF RFC 3031, January 2001.

[SGLG03] K. Sriram, D.W. Griffith, SuKyoung Lee, and N.T. Golmie. Optical burst switching: Benefits and challenge. October 2003. Proceedings of the First International Workshop on Optical Burst Switching WOBS, Dallas/TX.

[SML04] A. K. Somani, M. Mina, and L. Li. On trading wavelengths with fibers: A cost-performance based study. *IEEE/ACM Transaction on Networking*, 12(5), October 2004.

[TGCT99] L. Tancevski, A. Ge, G. Castanon, and L. S. Tamil. A new scheduling algorithm for asynchronous, variable length IP traffic with void filling. February 1999. Proceedings of Optical Fiber Communication OFC.

[TR03] J. Teng and G.N. Rouskas. A comparison of the jit, jet and horizon wavelength reservation schemes on a single obs node. *Proceedings of the First International Workshop on Optical Burst Switching WOBS, Dallas/TX*, October 2003.

[Tur99] J. Turner. Terabit burst switching. *Journal of High Speed Networks*, 8(1):3–16, 1999.

[TYC+00] L. Tancevski, S. Yegnanarayanan, G. Castanon, L. Tamil, F. Masetti, and T. McDermott. Optical routing of asynchronous, variable length packets. *IEEE Journal on Selected Areas in Communications*, 18(10):2084–2093, October 2000.

[VPD] Jean-Philippe Vasseur, Mario Pickavet, and Piet Demeester. *Network Recovery, protection and restoration of Optical, SONET-SDH, IP and MPLS*. Morgan Kaufmann. ISBN: 0-12-715051.

[WD97] N. Wauters and P. Demeester. Wavelength conversion in optical multi-wavelength multifiber transport networks. *International Journal of Optoelectronics*, 11(5):53–70, October 1997.

[Wil56] R.I. Wilkinson. Theories for toll traffic engineering in the usa. *Bell System Technical Journal*, 352:569–582, 1956.

[XPR01] L. Xu, H. G. Perros, and G. Rouskas. Techniques for optical packet switching and optical burst switching. *IEEE Communications Magazine*, 39(1):136–142, January 2001.

[XQLX03] J. Xu, C. Qiao, J. Li, and G. Xu. Efficient channel scheduling algorithm in optical burst switched networks. *Proc. of INFOCOM 2003 - 22nd Annual Joint Conference of the IEEE Computer and Communications Societies*, 3(1):2268 – 2278, 30 March-3 April 2003 2003.

[XVC00] Y. Xiong, M. Vandenhoute, and H. Cankaya. Control architecture in optical burst-switched wdm networks. *IEEE Journal on Selected Areas in Communications*, 18:1838–1851, September 2000.

[XY02] F. Xue and S. J. B. Yoo. Self-similar traffic shaping at the edge router in optical packet switched networks. April-May 2002. Proceedings of IEEE International Conference on Communications ICC 2002, 2002.

[YMD00] S. Yao, B. Mukherjee, and S. Dixit. Advances in photonic packet switching: An overview. *IEEE Communications Magazine*, 38(2):84–94, February 2000.

[YQ97] M. Yoo and C. Qiao. Just-enough-time JET: a high speed protocol for bursty traffic in optical networks. *Digest of the IEEE/LEOS Summer Topical Meetings*, pages 26–27, August 1997.

[YQD01] M. Yoo, C. Qiao, and S. Dixit. Optical burst switching for service differentiation in the next generation optical internet. *IEEE Communications Magazine*, 39(2):98–104, February 2001.