

# Assessment of packet loss for an optical packet router with recirculating buffer

Chris Develder, Mario Pickavet, Piet Demeester  
*Dept. of Information Technology (INTEC), Ghent University, Belgium*

**Abstract:** The design of an optical packet switched network to transport both data and voice, focussing on an IP client layer, is the subject of DAVID (Data And Voice Integration over DWDM), a research project funded by the European Community. The network is composed of a metropolitan part, where optical rings are connected through devices denoted as Hubs. Those Hubs also provide the connection to the backbone part of the network, made up by SOA based optical packet routers (OPRs) interconnected in a mesh. This paper focuses on the performance evaluation of the backbone network, in casu an OPR, in terms of packet loss rate (PLR): different optical buffer implementations and scheduling strategies are proposed, through simulation the resulting PLR is determined.

**Keywords:** DWDM, optical packet switching, simulation

## 1. INTRODUCTION

The research project Data And Voice Integration over DWDM (DAVID), is funded by the European Community as part of the Information Society Technology (IST) Program. DAVID aims at proposing a viable approach to optical packet switching [1]. The network is conceived as an optical packet switched transport infrastructure, interconnecting IP routers and/or switches collecting traffic from legacy networks. To this end, a traffic format independent of the client traffic type will be adopted.

The network architecture that will be adopted, and demonstrated in a testbed, consists of a ring-based metropolitan area network (MAN), and a wide area backbone network (WAN). In the MAN, ring nodes are interconnected through a Hub that also provides interconnection to the backbone part of the network. This wide area backbone network (WAN) will

consist of optical packet routers (OPRs), connected in a mesh via DWDM links.

This paper focuses on the WAN part of the DAVID network, more in particular the study of a single OPR: through simulation, we try to assess the performance of the OPR for different optical buffer configurations and scheduling strategies.

The rest of this paper is structured as follows: in Section 2, we briefly describe the DAVID network architecture. The description of the adopted simulation approach is presented in Section 3, the first results obtained with this simulation approach follow in Section 4. The final Section 5 concludes the paper.

## 2. THE DAVID NETWORK

Within the DAVID network, the optical packet switching approach is evaluated in both a metropolitan area and a backbone. This is reflected in the network architecture, which is based on a hierarchical interconnection of optical MAN rings to a mesh backbone WAN, as depicted in Figure 1. The network will deploy a mixed WDM/TDMA approach: a fibre will be carrying multiple wavelengths (up to 32 channels, at 2.5 or 10 Gbit/s) and time will be divided in fixed size slots carrying an optical packet made up of an optical header and payload. The payload part will be switched transparently through the network.

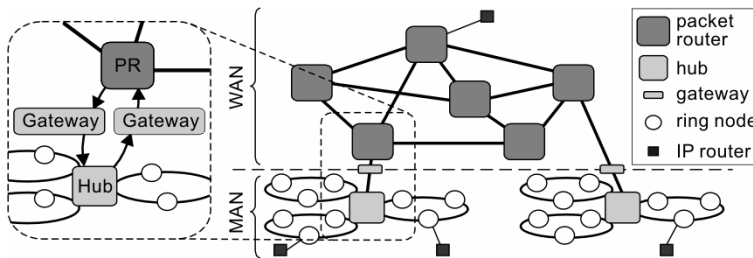


Figure 1. The DAVID network architecture.

The metro network comprises one or more uni-directional optical physical rings interconnected in a star topology by a Hub, collecting traffic from several nodes. Through a Gateway, the Hub will be connected to an OPR, thus providing the connection to the backbone WAN. The functionality of the Hub is interconnecting rings: also the connection towards the WAN (via the Gateway) logically can be seen as an extra ring to switch traffic to and from. The role of the Hub is then to switch traffic

between Metro rings and to/from the WAN: basically the Hub behaves as a space switch, and will be buffer-less. Indeed, the Hub will be an all-optical device, comprising a WDM synchronisation stage, a space switching stage, a wavelength conversion stage and possibly 3R regeneration. The absence of buffering in the Hub, and on any path between Ring Nodes not leaving the MAN, means that contention will need to be solved by means of a MAC protocol, dictating when what Ring Node can access the ring. How this MAC and the switch scheduling at the Hub may be implemented, is described and analysed in [2].

In the WAN, a mesh network will be formed where links consisting of one or more fibres will interconnect Optical Packet Routers (OPRs). This WAN will be operated in a way similar to electrical packet networks, and will adopt a hierarchical networking approach [3]. Contrary to the MAN, buffers will be present to aid in solving contention in the optical domain by means of Fibre Delay Lines (FDLs). Another means to tackle contention is the exploitation of the wavelength dimension: when two optical packets destined for the same output fibre need to be switched, they will be placed on different wavelengths. Only when this proves not to be possible, the optical FDL buffer will be addressed. These FDLs will be deployed in a recirculating buffer scheme: in DAVID, the OPR will be made up of a broadcast-and-select switching fabric based on SOA technology [4,5], where some of the wavelength ports will be connected to a buffer block of FDLs. The performance, in terms of packet loss ratio, of such an OPR for various FDL schemes and buffer scheduling strategies is the subject of the simulations discussed in the rest of this paper.

### **3. SINGLE NODE PERFORMANCE: SIMULATOR**

Before trying to evaluate node architectures on a network wide level, we need to obtain information on the performance of a single node. Therefore we have developed a simulation program to investigate the performance of an OPR in DAVID. In this section, we discuss the properties of the developed simulator.

#### **3.1 Node architecture**

A high level overview of the simulated node is depicted in Figure 2, and reflects the node architecture deployed in DAVID. The core of the OPR is the switching fabric, being a broadcast-and-select switching fabric based on SOA technology (see e.g. [4]). The node has  $N$  input ports and  $M$  output

ports (normally with  $N=M$ ), where the number of wavelengths per fibre can be set for each port (independently of the other ports). These ports include both the add/drop ports of the OPR coming from/going to the Gateway (interfacing to the MAN) and the ports for transit WAN traffic, connecting the OPR with other nodes in the backbone. From the point of view of the switch, these ports are equivalent. Also, the wavelengths within a port are considered to be equivalent, as the wavelength domain can be used to solve contention.

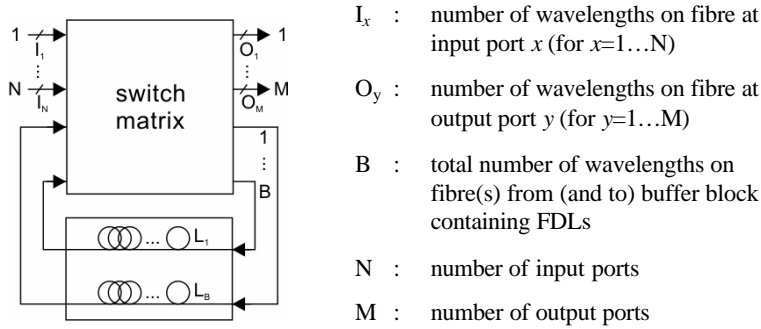


Figure 2. The simulated node and its parameters.

The number of wavelengths going to the buffer, denoted as  $B$ , and the lengths  $L_i$  ( $i=0..B$ ) of the fibre delay lines they are transmitted on, is another set of parameters. A packet entering the buffer will appear at the outputs after a certain number of timeslots, depending on the fibre length of the FDL. Indeed, the exact time is determined at the time the packet leaves the switch matrix and is put on one of the  $B$  wavelengths of the FDLs. More about the employed buffer scheduling strategy will follow in the sections to come.

### 3.2 Operation of the switch

The OPR switches optical packets of fixed length and operates in a slotted manner. Each timeslot when packets are present at the inputs, the switch matrix will forward the packets presented at the input ports (including the ports coming from the FDL buffer block) to the output ports and —if necessary— to the ports towards the buffer block of FDLs.

#### 3.2.1 Packet format

To determine which packets to switch to what port, the switch makes use of certain information associated with a packet. The parameters used in our

simulator are listed in Table 1. Note that these fields will not necessarily all have a direct counterpart in a real-world implementation's packet format: some of them are merely there for tracing purposes or because they make the simulation program implementation easier.

Table 1. Attributes associated with a packet in the simulator.

Attrib.	Meaning
fid	flow identifier: unique number associated with traffic source that generated the packet
src	source port: this is the port (fibre) along which the packet enters the DAVID node
dst	destination port: this is the outgoing port (fibre) along which the packet will have to leave the DAVID node (unless it is dropped of course)
pri	priority: packets with higher priority will get forwarded before packets of lower priority
tstamp	timestamp: packet will be marked with a timestamp (e.g. by means of a local clock maintained at DAVID node) when it enters the switch

The OPR will base its switching decisions on the following subset: *src*, *dst*, *pri* and *tstamp*. How these will be used is clarified below. The other fields are only used in the simulation program for the purpose of tracing and collecting statistics.

### 3.2.2 Switching operation

For each input port, the packets presented are simply forwarded to the output port they are destined for, as marked in the *dst* field. When more than  $O_y$  packets, say  $P_y$ , are destined for a particular output port  $y$ , preference is given to packets with higher priority. Within a particular priority class, packets that have already gone through the buffer of FDLs will be taken first. For the latter purpose, packets are marked with a timestamp (the *tstamp* field of Table 1) the moment they enter the switch. In this way, we try to avoid recirculations of the same packets.

Thus, packets destined for a particular outgoing port will be ordered in descending priority, and within a particular priority class in ascending timestamp order. If for the highest priority class more than  $O_y$  packets remain with the same timestamp, then only  $O_y$  packets will be picked randomly (to ensure fairness).

The remaining ( $P_y - O_y$ ) packets that could not be sent, because of lack of wavelengths on output port  $y$ , will be buffered if possible. When the buffer is full, the remaining packets will be dropped (resulting in packet loss).

### 3.2.3 Scheduling algorithm

The strategy outlined above is implemented in our simulator using the algorithm illustrated in Figure 3. The algorithm, which is repeated each timeslot when packets are to be sent, consists of four phases:

1. The first phase is to *inspect new packets*. All the packets entering the node at the same input port (fibre) are put in a queue<sup>1</sup>. This means we have  $N$  input queues. The sole purpose of these input queues is to ensure we will not process more than  $I_x$  packets per timeslot at input port  $x$ . Indeed, as we use different sources per (source port, destination port, priority)-triple, it may happen —depending on the traffic model used— that at a particular timeslot the total number of generated packets at input port  $x$  accidentally exceeds the number of available wavelengths on that port (even if we ensure that the total average rate is lower than  $I_x$  packets per timeslot). In fact, these input priority queues simply model the previous parts of the network where the traffic has gone through. Indeed, the traffic will be shaped already by being transported through the previous nodes. This means that the input priority queues are not really part of the optical packet router model, but rather model the rest of the network.
2. In the second phase, the packets are *forwarded from input ports to output ports*. For each input port  $x$ , at most  $I_x$  packets are taken, marked with a timestamp (based on local clock of OPR), and put in the appropriate output queues (based on the *dst* field of the packet). Also, the packets at the output ports of the buffer at the current timeslot are put in the appropriate output queues.
3. In the third phase, the output queues are emptied. For output port  $y$ , the first  $O_y$  packets are *forwarded to the receivers*. (In our simulation program, that simply means we e.g. increase the receive-counter for the *fid* of the packet.) The remaining packets are put in the single buffer queue.
4. The fourth phase is the *buffering phase*: all packets that could not be sent (because of temporary overload) are now in the buffer queue. From this queue, as many as possible packets are taken and put in a FDL. As the buffer has only  $B$  wavelengths (see Figure 2), at most  $B$  packets can be

<sup>1</sup> Note that these queues are really nothing more than an aid in implementing the desired priority scheme: packets often have to be processed in order of descending priority, and within a same priority class in order of increasing timestamp; from the set of packets within the same priority class with the same timestamp, a certain number of packets has to be randomly picked. This is exactly what the queues are used for: all packets in a priority queue will be sorted as indicated before, and the “pop” operation will randomly pick a packet from the queue belonging to the set of packets with highest priority and lowest timestamp within that priority class.

put in the buffer. The remaining packets in the buffer queue are dropped. Which delay line to use for a particular packet, will be discussed in detail in Section 3.3.

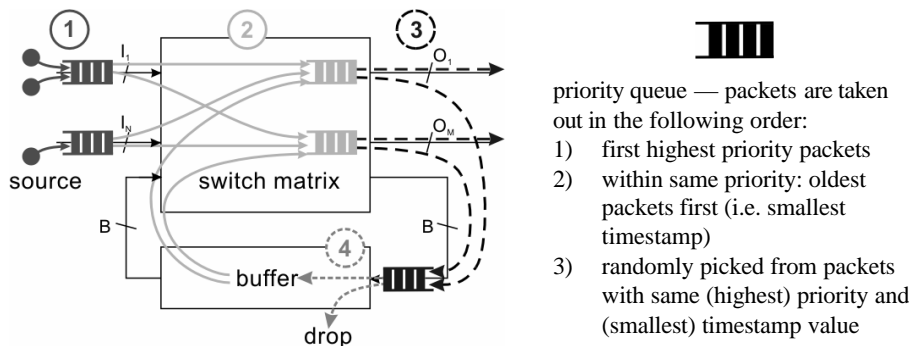


Figure 3. Implementation of the scheduling strategy in the simulation program.

Remark that at the beginning of each cycle in this algorithm (i.e. each timeslot), the output port queues (grey in the right of the figure above), and the buffer queue (black, bottom) will be empty.

### 3.3 Buffer scheduling strategy

In step 4 of the scheduling algorithm outlined above, all packets that could not be sent directly to the output ports because of contention are joined in a priority queue. Now, for each of these packets, it has to be decided what delay to use. So far, we have provided three buffering strategies, of increasing complexity, in our simulator:

- a) *FDL with smallest delay.* Each packet taken from the buffer queue is put in the FDL having the smallest delay that is not occupied already. When all FDLs are occupied, the remaining packets are dropped, resulting in packet loss.
- b) *FDL with smallest delay, avoid recirculation.* In this buffer strategy, it is avoided as much as possible to have more than  $O_y$  packets ( $O_y$  being the number of wavelengths on the output port  $y$  the particular packet is destined for) leaving the buffer at the same time and heading for the same output port. This results in a buffer strategy where for each packet the buffer contents are inspected: the packet is given the smallest possible delay  $D$ , such that a free FDL with delay  $D$  can be found and the entire buffer contains less than  $O_y$  packets with the same or higher priority. If no such delay  $D$  can be found, the packet is dropped.

- c) *FDL with smallest delay, avoid recirculation, but fully use FDLs.* This buffer strategy is the same as buffer strategy b), except that if no suitable delay  $D$  can be found to avoid recirculations, the packet is not dropped but put in the free FDL with the smallest delay. Only if all FDLs are fully occupied, the packet will be dropped.

Strategy (a) blindly tries to minimise the delay. The advantage of the strategy is its simplicity: it just takes the first  $B$  packets and fills all the available FDLs. No information on previously scheduled packets is needed, which implies that no state information on the contents of the buffer block has to be maintained.

The second strategy (b) is more intelligent and exploits the knowledge of the buffer contents in order to try to avoid recirculation. For this strategy, it is assumed that the control logic of the OPR maintains information on the complete contents of the fibre delay lines. It avoids recirculation of the highest priority packets completely. Indeed, if a packet with priority  $P$ , destined for output  $Y$  is taken out of the buffer queue, it will be scheduled in the fibre delay line of length  $D$  iff the number of packets already scheduled for  $t_{\text{now}}+D$ , destined for  $Y$  and with priority greater than or equal  $P$  is less than the number of output wavelengths at port  $Y$ . At the time  $P$  leaves the buffer, it will only be recirculated (or even lost) if at that time new packets arrive with priority higher than  $P$  (cf. packets of same priority have lower precedence because they have a lower timestamp).

The third strategy (c) extends the second, by trying to avoid losses as much as possible by fully using the FDLs. Indeed, strategy (b) suffers from the fact that packets may be dropped, even if the buffer is not completely filled, and even if some buffer ports are still free. This means that we might end up with losing more packets than when we blindly put packets in the free FDL with smallest delay. It is easy to conceive an artificial situation where this is possible: e.g. consider the situation where we have only FDLs of length 1, and we have multiple input ports with traffic destined for same output port. Now, suppose that at a certain time  $T$  more than  $O_y$  sources offer a packet (where  $O_y$  is the number of lambdas at the output port). Due to the limitation of entering only a maximum of  $O_y$  packets, we will lose some packets. However, if the buffer block has more than  $O_y$  ports, we could avoid some of the losses by entering all offered packets (and recirculate them at next timeslot).

It is clear that other strategies can be devised, e.g. strategies that not necessarily minimise the delay (by distributing packets for the same output port equally over the available FDLs). The impact of the scheduling strategy is clearly an issue that will need to be resolved. The simulation results in the next sections all use strategy (a). However, we will present results for different strategies at the conference.



### 3.4 Traffic models

As the DAVID node operates in a slotted manner, we use a discrete event simulator. For each time slot we need to know how many packets are offered for forwarding at the input ports. The packets will be generated by traffic sources. Each source will produce packets for a particular (input port, output port)-pair, with a given priority, at a given rate (i.e. average number of packets per time unit). We have implemented different traffic models, including:

- POISSON: Poisson inter-arrival times between consecutive packets;
- BURST\_GEO\_POISS: bursty traffic, generating bursts with geometrically distributed number of packets per burst and Poisson distributed inter-arrival times between bursts (packets within a burst arrive at same time).
- PARETO: On/Off traffic using Pareto distribution for both on- and off-periods. This model was added because it is well-known that an aggregate of these sources will produce a packet stream that exhibits self-similar behaviour [6,7]. The Pareto distribution's probability mass function is given by eq. 1, thus we have four free parameters for both the on- and off-periods: the shape parameters  $\mathbf{a}_{\text{on}}$  resp.  $\mathbf{a}_{\text{off}}$  and the minimum burstsizes  $b_{\text{on}}$  resp.  $b_{\text{off}}$ .

$$p(x) = \frac{\mathbf{a} \cdot b^{\mathbf{a}}}{x^{\mathbf{a}+1}}, \text{ for } x \geq b, \text{ with } \mathbf{a}, k > 0 \quad (\text{Eq.1})$$

## 4. SINGLE NODE PERFORMANCE: SIMULATION RESULTS

In this section we briefly discuss the first simulation results obtained with the node simulator as discussed in the previous section.

### 4.1 Simulation set-up

The simulations presented in this section are based on the parameter values that are envisaged in the demo testbed under development within the DAVID framework. The switching fabric to be used in DAVID will have a dimension of 256×256, which means that it will be a matrix interconnecting 256 input wavelengths and 256 output wavelengths. There will be 6 input/output ports in the OPR, each carrying 32 wavelengths. That leaves 64

wavelengths providing connection to the fibre delay lines in the recirculating buffer.

The traffic pattern offered at the inputs was considered to be uniform, meaning that for each input, the load was equally distributed over the 6 output ports. Also, 3 priorities were considered, where 50% of the traffic had the highest priority (2), 25% was of middle priority (1), and the other 25% was of lowest priority (0). To create this set-up, we used a single traffic source per (input port, output port, priority)-triple, thus a total of  $6 \times 6 \times 3 = 108$  traffic sources. The total offered load at the inputs by these sources was set at 0.8 (meaning that per input port, on average  $0.8 \times 32 = 25.6$  packets were offered per timeslot). These parameter values are summarised in Table 2.

Table 2. Simulation parameters

Parameter	Value
nr. of input/output ports	6
nr. of $\lambda$ s per input/output port	32
priorities	50% highest (2), 25% middle (1), 25% lowest (0)
traffic load	0.8

The traffic source models used are the Poisson and Bursty models mentioned in 3.4. The Poisson model uses Poisson distributed inter-arrival times between successive packets generated by the same source. The Bursty model uses Poisson inter-arrival times between bursts of geometrically distributed size with an average of  $n$  packets per burst. We have set  $n$  to 2 and to 4. This means that for those cases, the traffic arrives in bursts of average size 2 (or 4) packets per timeslot.

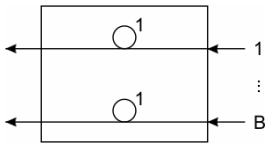


Figure 4. Fixed loop size.

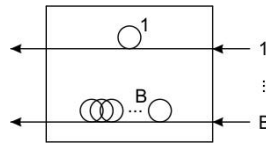


Figure 5. Increasing loop size.

With the simulations presented below, we have explored two buffer configurations. The “Fixed Loop Size” case of Section 4.2 investigates the effect of adding extra wavelengths on FDLs of the same size (Figure 4), whereas a second set of simulations (Section 4.3) explored the advantage of using the extra wavelengths for FDLs of increasing length (Figure 5).

## 4.2 Fixed loop size

The simulations presented in this section considered the buffer configuration sketched in Figure 4. We consider a buffer block consisting of

fibre delay lines of length 1 (timeslot). The graphs show the loss rates for an offered load of 0.8 for an increasing number of wavelengths (parameter B on the figure). The plots marked as BURST-4 are for Bursty traffic (see 3.4) with average burst size 4, BURST-2 for burst size 2. The curves for POISSON are for Poisson traffic. The curves for PARETO-1 (resp. PARETO-2) are for Pareto On/Off sources using  $a_{on}=1.3$ ,  $b_{on}=1$  (resp.  $b_{on}=2$ ) and  $a_{off}=1.5$ . The remaining parameter  $b_{off}$  follows from the constraint that the load has to be a given value. The values were inspired by [7], which reports on-times being heavy tailed with  $a_{on} \approx 1.0$  to 1.3, and  $a_{off} \approx 1.5$ .

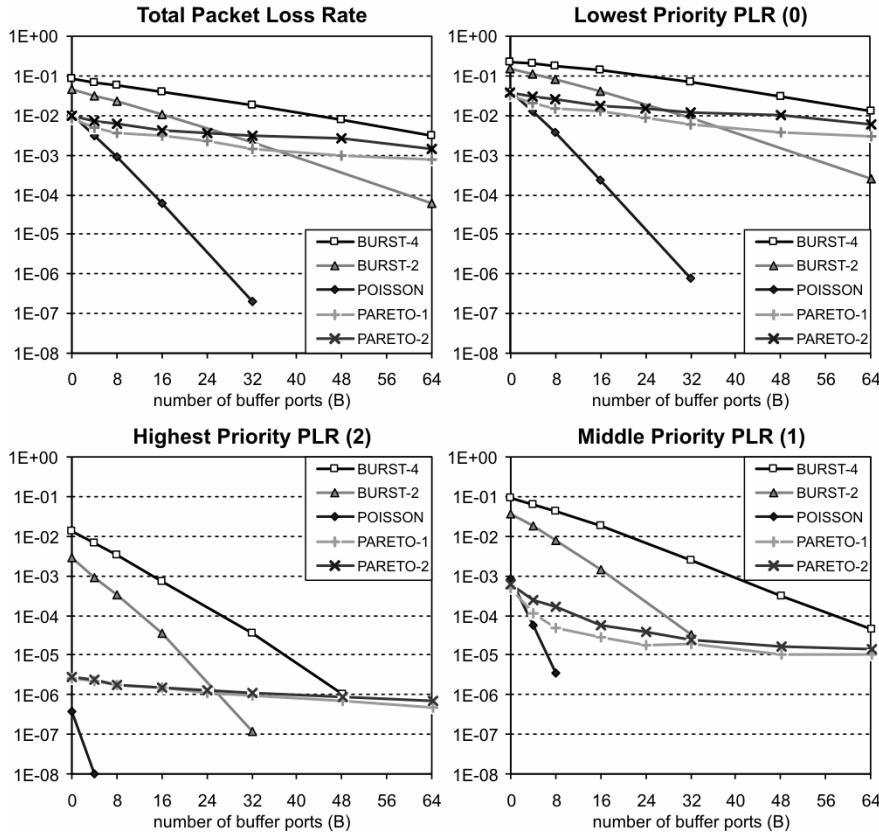


Figure 6. PLR for increasing number of loops of length 1 (i.e. delay of single slottime).

From these graphs, several observations can be made. In the first place, we notice that the loss probability is the highest for low priority packets. Also, the effect of increasing the number of FDLs on the PLR is the biggest for the high priority packets: the slope of the declining curves (for increasing number of wavelengths) is steeper for higher priority packets. This confirms

that the priority scheme is working: only if the highest priority packets are satisfied other packets can use the FDLs.

With regard to the traffic source models, the differences are striking. Increasing the buffer size for Bursty traffic is far less effective than for Poisson traffic: for Bursty traffic, the slopes of the declining curves are far less steep, and the PLRs are significantly higher, especially for the high priority packets, where the difference is about a factor  $10^5$ : Whereas for Poisson traffic the high priority traffic could possibly do without buffering (loss rate less than  $10^{-6}$ ), this is most certainly not the case for Bursty traffic. From this, we may conclude that —if possible— we should avoid creating bursts of packets all destined for the same output port. The results seem to indicate that shaping the traffic at the inputs, in order to spread packets for same destination over different timeslots can help a lot. This means also that we should avoid any scheduling strategy that would create such bursts. Surely, use of randomness when choosing between packets of the same priority from different flows, is important here. This randomness will smoothen burstiness.

### 4.3 Increasing loop size

The results presented below again show packet loss rates for an increasing number of wavelengths used. However, in this case we use a FDL of a different length for each additional wavelength: if we use  $B$  wavelengths, we use FDLs of lengths 1, 2, 3, ...,  $B$  timeslots as illustrated in Figure 5. The traffic sources used are again of Poisson, Bursty, and Pareto types, and all results are for load 0.8.

When we compare the PLRs for different priorities, we notice that again the highest priority packets lose less. However, if we look at the rate of PLR decrease with increasing number of buffer ports (wavelengths), we see a striking difference with the graphs for the fixed length delay line case (see Figure 7).

Adding additional buffer ports with increasing FDL lengths (instead of more of the same) has a major impact on the PLR. The highest priority traffic gains about as much as in the case of fixed FDLs, but for FDLs with increasing lengths, the lower priorities gain also. From the individual graphs per priority (that are omitted due to space limitations), we see that the decreasing slope is now more of the same order for the different priorities.

These results seem to suggest that it is advantageous to add buffer ports of increasing FDL length instead of more of the same.

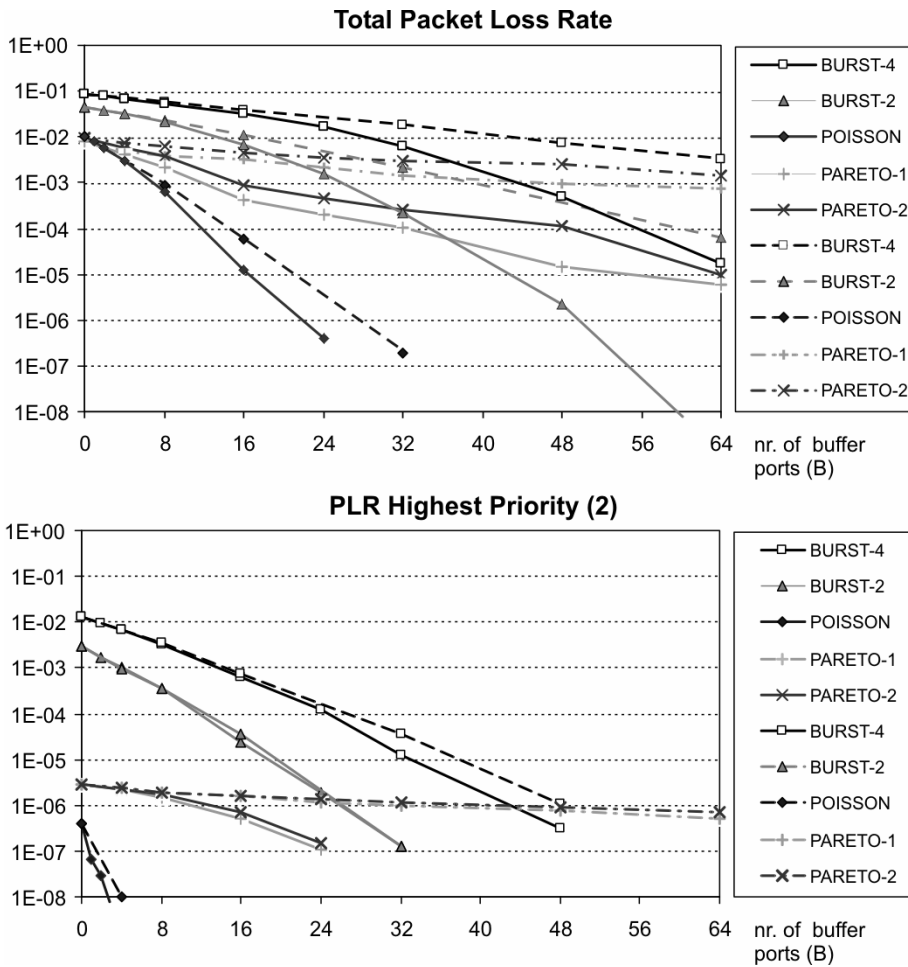


Figure 7. Comparison of PLR for FDLs of fixed size (dashed lines) versus increasing size (full lines) for an increasing number of buffer wavelengths.

#### 4.4 A note on fairness

To verify the operation of our simulator, we have also investigated the fairness among different flows of the same priority. In the graphs below, the result of that analysis is illustrated for a single simulation: the case where there is a buffer of 8 FDLs of length 1, and the load is 0.8 generated by traffic sources of the Burst type, with mean burst size of 4. Similar graphs are obtained for other cases.

According to our scheduling strategy, we expect that losses for packets going to the same destination are equally distributed over the flows coming

from the different input ports. In Figure 8, we see the loss ratios for all the individual sources generating traffic with output port 1 as destination. As expected, we see that the losses within a priority class are neatly distributed over all the flows coming from the different input ports.

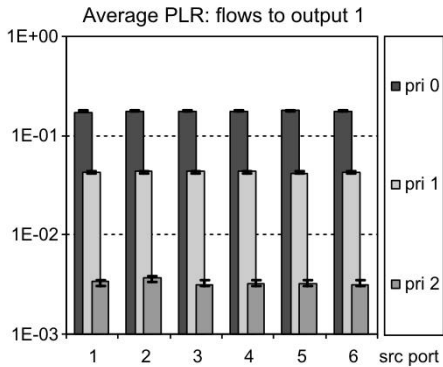


Figure 8. PLR for individual sources generating traffic for output port 1; the deviation markers indicate the interval ( $avg-stdev$ ,  $avg+stdev$ ) taken over the six traffic sources per priority class.

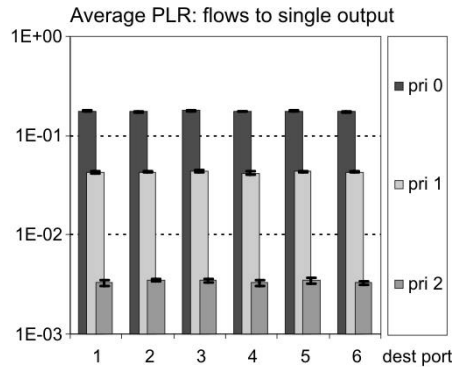


Figure 9. Average PLR over all sources generating traffic for a particular output port; the deviation markers indicate the  $stdev$  on those averages.

In Figure 9, we see the averages for the different output ports. (This means that the three bars at “dest port” 1 are actually the averages of the corresponding bars in Figure 8. The graph shows that the loss rates are the same for every output port. This indicates that the losses are fairly distributed over all output ports, which is what we wanted (and thus expected).

For completeness, we have also compared sources coming from a single source port (going to the different output ports). The resulting graphs (which are not shown due to space limitations) again indicate that the losses are fairly distributed among the different flows.

## 5. CONCLUSION

We have presented part of the work carried out within the DAVID project on assessing the performance of the proposed OPR. The presented results show the packet loss rate for streams of fixed size packets (length is equal to DAVID slot length) with service differentiation: we implement a strict priority scheme with three priorities. It is shown that through introducing randomness in the decision procedure, the losses are equally spread among flows of the same priority.

Through our simulation results, we demonstrate that it is advantageous to add extra FDLs to the optical buffer structure, instead of providing more wavelengths on the same FDL (i.e. same buffer depth). However, the traffic source model has a strong impact on how effective the reduction in loss rate, compared to no buffering, is.

We have also proposed different buffer scheduling strategies to decide upon what packets to put in which delay line. The results covered in this paper only reflect one strategy, but results for other strategies will be presented at the ONDM conference.

## ACKNOWLEDGEMENTS

This work has been supported by the European Commission through the IST-project DAVID, and by the Flemish Government through the IWT-project ITA/980272/INTEC. The first author is a Research Assistant of the Fund for Scientific Research – Flanders (F.W.O.-Vl., Belgium).

## REFERENCES

- [1] L. Dittman, H. Christiansen, D. Chiaroni, "DAVID – An approach towards MPLS-based optical packet switching with QoS support", in Proceedings of Photonics In Switching 2001 (PIS 2001), Monterey, CA, June 2001.  
*available online: <http://david.com.dtu.dk>*
- [2] A. Bianco, G. Galante, El. Leonardi, F. Neri, "Access control protocols for interconnected WDM rings in the DAVID metro network", in Proceedings of Thyrrhenian International Workshop on Digital Communications, Taormina, Italy, September 2001.
- [3] L. Dittman, H. Christiansen, M. Berger, "Hierarchical MPLS – An approach for efficient resource administration in multi-technology networks", in Proceedings of the 6<sup>th</sup> European Conference on Networks and Optical Communications (NOC 2001), Ipswich, England, June 2001.
- [4] D. Chiaroni, "Status and applications of optical packet switching", Tutorial at 27<sup>th</sup> European Conference on Optical Communication (ECOC 2001), Amsterdam, The Netherlands, October 2001.
- [5] D. Chiaroni, N. Le Sauze, T. Zami, J.-Y. Emery, "Semi-conductor optical amplifiers: a key technology to control the packet power variation", in Proceedings of 27<sup>th</sup> European Conference on Optical Communication (ECOC 2001), Amsterdam, The Netherlands, October 2001.
- [6] W. Willinger, M.S. Taqqu, R. Sherman, D.V. Wilson, "Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level", IEEE/ACM Transactions on Networking, Vol. 5, No. 1, January 1997, pp. 71-86.
- [7] M.E. Crovella, A. Bestavros, "Self-similarity in World Wide Web traffic: evidence and possible causes", IEEE/ACM Transactions on Networking, Vol. 5, No. 6, December 1997, pp. 835-846.