Comparing fixed size and variable size packet in an optical ring networks: Algorithms and performances

D. Barth[1], J. Cohen[2], L. Gastal [2], T. Mautor[1], S. Rousseau[1]

1:PRiSM, UMR ????,Company1_address
2: LORIA, UMR 7503, B.P.239, 54506 Vandoeuvre Lès Nancy, France.

Abstract: In this paper, we compare the use of two packets models in slotted optical ring networks, i.e., a model where each message has to be routed in consecutive slots, and a model where the slots making a same message can be routed independently. We first focus on the algorithmic complexity of the related problems. Then, we give the results we obtain with a OMNET simulator in terms of messages delay and jitter and of communication resources really used at each step.

1. Introduction and description of the problem

In this paper, we compare the communication efficiencies of a simple optical ring network under two packet models, i.e., a fixed size and a variable size models. Consider a ring network *R* connecting *N* nodes with only one slotted wavelength (with *K* slots on each link). The ring is assumed to be in the OPADM DAVID model [?]. At each communication step, each node sees the slot of the ring located on him. If the data contained by this slot is intended for him, the node reads it and removes the data from the slot (it becomes empty). If the slot is empty, the node can use it to send a new message (or a part of the message) to its destination.
The purpose of this study is to compare variable size packet and fixed size packet from two points of view:

- What is the difficulty of determining an optimal scheduling in these two models?
- What are the performances of each model in terms of packet delay and jitter?
-

As we want the difficulty to be linked only to the models behaviours and not to the network control, to focus on these two questions we assume a very simple optical network, i.e., a slotted ring with only one wavelength. Each node could have some messages to send to another nodes. Each such message M is characterised by the origin node: **or(M)**, the destination node: **dest(M)**, the size in number of slots: **sz(M),** the time at which it becomes available in or(M): **dispo(M)** and the distance (number of slots=number of steps) from or(M) to dest(M): **dist(M)** = *K * ((dest(M) – or(M)) mod N).* We also consider the time **First(M)** (resp. **Last(M)**) at which its first slot (resp. last slot) is sent on the ring. Different measures can be defined and applied on each message M:

- The delay, defined by **Delay(M)=(dist(M)+Last(M)-dispo(M)),** represents the time between the arrival of the message on the origin node and the end of its transmission, i.e. the end of its reading on the destination node.
- The jitter, defined by **Jitter(M)=(Last(M)-First(M)),** represents the time between the beginning and the end of the emission of a message.
- The tardiness, defined by **Tard(M)=Last(M)-Dispo(M)-sz(M)+1,** represents the difference between the delay of M and its minimal possible delay (equal to dist(M)+sz(M)-1). This measure, inspired from some works on scheduling, is interesting to compare the delays of packets of different size.
-

The **variable size packet (VSP) model** consists here in saying that the Jitter is a constraint : for each message M, we want **Jitter(M)=sz(M)-1**. This implies that all the slots of a same message have to be **sent contiguously** on the ring. Let us remark that, in this model, the tardiness of a message M is equal to : **Tard(M) = First(M)-Dispo(M)**.
The **fixed size packet (FSP) model** says that all the slots of a same message can be sent independently the ones from the others. Thus, the Jitter is in this case an evaluation parameter of the behaviour of the network.
The paper is organised as follows. First, we present the theoretical problems we focus on in a static centralised model (i.e., where the (finite) set of messages to be sent is known in advance). Then, we give some simulation results under a (more realistic) distributed on line model.

2. Complexity

We focus on the following problems :

Problem MS-VSP-Scheduling
Given : A ring *R*, a set of messages in each node, an integer *S*.
Question : Could all the messages reach their destinations after at most *S* steps, with the constraint *Jitter(M)=1* for each message *M*?

Problem MD-VSP-Scheduling
Given : A ring R, a set of messages in each node, an integer K.
Question : Does exist a scheduling of S on *R* such that the maximal tardiness *Tard(M)* over all messages *M* is less

or equal to K, with the constraint $Jitter(M)=1$ for each message M?

We call makespan the date when all message are reached their destination. The problem **MS-VSP-Scheduling** and **MD-VSP-Scheduling** are NP-complete when the criterion to be minimized is the makespan or the tardiness (reduction from the k-partition problem [1]). So we will describe an approximation algorithm.

We focus on the **MS-VSP-Scheduling** problem and begin our analysis by the static centralised model. Let $C_{max}(S)$ be the makespan of the schedule S and $C_{max}(OPT)$ be the makespan of the optimal schedule. We will describe the scheduling S: if a node has a message of length l to send and there are at least l consecutive slots free, the node puts his message on. Let L_{max} be the maximal length of all messages. The quality of this scheduling S is:

$$C_{max}(S) \leq (L_{max} + 1)\, C_{max}(OPT)$$

It gives an indication of how well the heuristic is guaranteed to perform in the worst case. Thus, if L_{max} is equal to 1, this schedule is a 2-approximation.

Now, we focus our analysis on the distributed model with assuming that the maximal length of all messages is less than a constant L_{max}. Each node only knows its own set of messages to be sent. We define a frame as L ($L_{max} \leq L$) consecutives slots on the ring. Thus, ring R is cut into $(NK)/L$ frames (recall that N is the number of nodes and that K is the number of slots on each link). Now, when a frame crosses over on a node, if this node has a message of length l to send and if the frame has at least l consecutive slots free, the node puts his message on this frame. The quality of this scheduling S is:

$$C_{max}(S) \leq (2\,L + 1)\, C_{max}(OPT)$$

Moreover, if L is a multiple of L_{max}, we have $C_{max}(S) \leq (2\,L_{max} + 1)\, C_{max}(OPT)$.

Let us summarize all theses results in this table (recall that L is the size of a frame and L_{max} is the maximal length of all messages): we have $C_{max}(S) \leq \beta \times C_{max}(OPT)$ and β take the values:

	Static centralised		Static distributed
	$L_{max} = 1$ and L = 1	L = L_{max}	L = $\alpha\, L_{max}$
β	2	$L_{max} + 1$	$2\, L_{max}$

3. Algorithmic and Simulation studies

A FAIRE : SR

4.1 Vision distribuée

4.2 Résultat de simulation .

underlined and

Equations: Equations must be numbered sequentially. Equation number should be placed in the right-hand margin as follows:

$$a = b + c \qquad [1]$$

Illustrations: Illustrations may be in colour provided that they appear clearly when printing in grey shades.

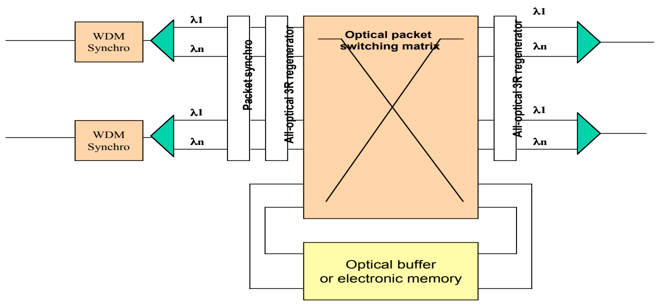

Figure 3 : Structure of an all-optical packet router

5. Conclusion

This paper has provided guidelines to submit final papers to PS 2003 Photonics in Switching. Please make an effort.

6. Acknowledgment

The authors acknowledge the contribution of their colleagues to this work.

6. References

[1] J. Carlier: " *One machine problem*". European Journal of Operational research 11(42-47), 1982.

[2] M. Garey and D. Johnson: " *Computers and Intractability: A Guide to the Theory of NP-Completeness* ", W.H. Freeman, 1979.

[3] M. Pinedo: "*Scheduling: theory, algorithms and systems*". Prentice Hall ed., 2002.

[4] G. Tell. "*Introduction to distributed algorithms*". CamBridge Int. series on Parallel Computation Vol.1, Cambridge University Press, 1984.

[5] V.V. Vazirani : " *Approximation algorithms* " . Springer Verlag ed., 2001.

7. Glossary

PDF: Portable Document Format

RTF: Rich Text Format

[CA82] J. Carlier. One machine problem. European Journal of Operational research 11(42-47), 1982.

[GJ79] M.R. Garey and D.S. Jhonson. Computer and intractability: a guide to the theory of NP-completeness. Freeman ed., 1979.

[PI02] M. Pinedo. Scheduling: theory, algorithms and systems. Prentice Hall ed., 2002.

[TE84] G. Tell. Introduction to distributed algorithms. CamBridge Int. series on Parallel Computation Vol.1, Cambridge University Press, 1984.

[VA01] V.V. Vazirani. Approximation algorithms. Springer Verlag ed., 2001.