Algorithmic study of complexity of two QoS packet models in an optical slotted ring network

Dominique Barth[†], Johanne Cohen‡, Lynda Gastal[‡]†,
Thierry Mautor[†], Stéphane Rousseau[†]

† *PRiSM, Université de Versailles-S^t Quentin en Yvelines, 45 Bld des Etats Unis, F-78035 VERSAILLES*

‡ *LORIA, Campus Scientifique BP 239, F-54506 VANDOEUVRE LES NANCY*

‡† *LRI, Bt 490 Universit Paris-Sud 91405 Orsay Cedex France*

{*barth,mautor,ros*}*@prism.uvsq.fr,jcohen@loria.fr, gastal@lri.fr*

In this paper, we consider an optical slotted ring network. We distinguish two QoS packet models. In the first one, each sub-packet of a same packet can be routed independently, and in the second one each sub-packet has to be routed in consecutive slots (no jitter). We evaluate performances in terms of jitter and delay of routed packets. First, we study the algorithmic complexity of minimizing delay in the second model and show it is NP-complete. Then, we propose distributed and on-line algorithms for the two models. Finally, we compare these algorithms using an OMnet++ simulator.

OCIS codes: 060.4510

1. Introduction

The convergencebetween Internet and optical networks is also a major key-point in the conception of future telecommunication networks [11]. As mentioned by the ITU [8], architectural choices for the interaction between IP and optical network layers are keys to the successful deployment of the next generation networks (NGN). This is also a major topic of some working groups of the IETF [7].

This study concerns the DAVID project[A] we have been involved in. The DAVID optical network is a WAN interconnecting optical MANs, each one consisting in disjoint optical slotted rings [2]. Metropolitan optical rings have also been studied in the IST-1999-10402 European project **METEOR**.

One of the questions addressed in the DAVID project and in some other studies was the choice of the packets format and size transported by the slotted ring [3, 15, 14, 13, 12]. In this paper, we study and compare two communication models in a simple optical slotted ring network. At each communication step, each node accesses the slot located on it. If the data contained by this slot is intended for him, the node reads it and makes this slot free. If the slot is free, the node can use it to send a new sub-packet of a packet the node has to send. Let s the size of a slot, each packet is considered of a sequence of sub-packets of size s. We distinguish two QoS packet models:

the *Splittable Packet model (SP)*: All the sub-packets of a given packet can be sent independently on the ring.

[A] *Data and Voice Integration over DWDM, European project of the 5th PCRD*

the *Non Splittable Packet model (NSP)* The sequence of sub-packets of a given packet has to be sent in contiguous slots on the ring.

Contrary to the *Non Splittable Packet model*, in the *Splittable Packet model*, packets can be received with jitter. In this case, the whole packet needs to be reconstituted within the destination node whereas in the *Non Splittable Packet model* no memory is necessary in the destination node to reconstitute packets.

First, we define the considering model and the problem we focus on. In Section 3, we study the difficulty of this problem in a centralized and off-line context and prove the NP-completeness. In Section 4, we propose distributed and on-line algorithms and compare them by simulations with OMnet++. Finally, Section 5 summarizes our results and underlines the aspects that could be studied in some further works.

2. Models and problems

Within this framework, we consider a slotted optical ring network R connecting N nodes with only one wavelength. There are $(K-1)$ slots between each pair of consecutive nodes and one slot located on each node. Consequently, the total number of slots on the ring is equal to $K \times N$.

At each unit of time, all the slots are turning of one notch. Thus, a slot located on a node is located on the consecutive node after K units of time. Each node could have some packets to send. To be sent, a packet has to be divided in sub-packets of slot-size and emitted on the ring using the slots. Only one sub-packet can be emitted in a slot at the same time. We assume here that it is impossible to interrupt the transfer of a sub-packet. More precisely, once a sub-packet is emitted by its origin node and is put in a slot, it has to stay in this slot until the destination node is reached. It cannot be temporally stocked by an intermediate node.

Any packet P is characterized by :

- its origin node : $or(P)$,
- its destination node : $dest(P)$,
- its size in number of sub-packets of size s (the size of a slot): $sz(P)$ (we suppose that $sz(P) < K \times N$),
- the time when it becomes available in or(P), called its release date and noted $release(P)$,
- the distance (number of slots = number of steps) from $or(P)$ to $dest(P)$: $dist(P) = K * ((dest(P) - or(P)) \mod N)$.

2.Packet transmission performances.

Different parameters can be considered to evaluate the performances of transmission of each packet on the ring.

We denote as $First(P)$ (resp. $Last(P)$) the time when the first sub-packet (resp. last sub-packet) of a packet P is sent on the ring.

The *delay*, defined by $Delay(P) = (dist(P) + Last(P) - release(P))$, represents the time spent between the release date of the packet and the arrival of the whole packet in the destination node.

The *jitter*, defined by $Jitter(P) = (Last(P) - First(P))$, represents the time spent between the sending of the first part of the packet and the sending of its last part.

The *over-delay*, defined by $OvDel(P) = Last(P) - release(P) - sz(P) + 1$, represents the difference between the delay of P and its minimal possible delay (equal to $dist(P) + sz(P) - 1$). This measure, inspired by some works on scheduling [4], is interesting to compare the delays of packets of different sizes.

With these parameters, the NSP model can also be defined by considering that the jitter is a constraint : for each packet P, we must have $Jitter(P) = sz(P) - 1$. In this model, the over-delay of a packet P is equal to: $OvDel(P) = First(P) - Release(P)$.

In the following Figure (see Figure 1), we propose a very simple example to illustrate some of our definitions. Let us consider a ring network connecting N nodes with 2 slots between each pair of nodes ($K = 3$). Here we focus on the link between nodes n_1 and n_2 and we suppose that two packets P_1 and P_2 have to be sent with n_1 as origin and n_2 as destination. These two packets are both available at time 2 and their size is 2. Let us suppose that the first packet is sent at time 2 and the second one just after P_1.

For the first packet, we have $First(P_1) = 2$, $Last(P_1) = 3$, $Delay(P_1) = 4 = 3 + 3 - 2$, $Jitter(P_1) = 1$ and $OvDel(P_1) = 0$.
For the second packet, we have $First(P_2) = 4$, $Last(P_2) = 5$, $Delay(P_2) = 6 = 3 + 5 - 2$, $Jitter(P_2) = 1$ and $OvDel(P_2) = 2$.
In this example, we do not compute the ratio of utilization previously defined as the whole network is not represented. Here on the four slots of Figure 1, the ratio of utilization between times 2 and 8 is equal to 16/28.

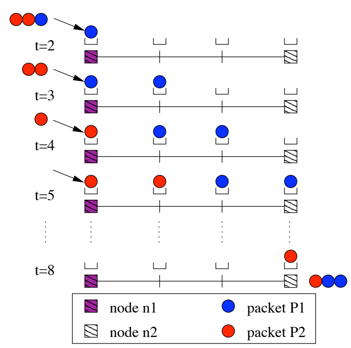

Fig. 1. Illustration of the emission of 2 packets from n_1 to n_2

2.Decision problem

In the NSP model, we ensure that all packets are received without any jitter. In this model, we evaluate the difficulty of finding an emission of all packets that minimize the over-delay of routed packets. In order to study the algorithmic complexity, we consider this model in an centralized and off-line context and we define this problem as a decision problem by:

Problem: MD-NSP-Emission

Given : A ring R, a set S of packets, an integer B.
Question : Does it exist an emission of S on R such that the maximal Over-Delay $OvDel(P)$ over all packets P is less than or equal to B, with the constraint $Jitter(P) = sz(P) - 1$ for each packet P?

Now, we prove the NP-Completeness of this problem.

3. Study of one centralized ring

The MD-NSP-Emission problem can be considered as a scheduling problem [9, 1] in which we want to schedule the emission of the different packets on the ring. Herein, to prove the NP-Completeness of this problem, we reduce a partition problem to an specific instance of this problem.

Theorem 1 *MD-NSP Emission problem is NP-complete in the strong sense.*

Proof: First, it is easy to see that MD-NSP Emission problem belongs to NP since for a given scheduling S on R, it is possible to check in polynomial time that the over-delay of all packets in S is less than or equal to B .

Then, we reduce 3-PARTITION to MD-NSP Emission problem. 3-PARTITION is defined as follows [6]:

Problem: 3-PARTITION PROBLEM

Given : `A finite set` C `of` $3m$ `elements` $\{a_1, \ldots, a_{3m}\}$`, a bound` $\beta \in Z^+$ `and a size` $s(a_i) \in Z^+$ `for each element` a_i `of` C `such that` $\sum_{i=1}^{3m} s(a_i) = m\beta$.
Question : `Can` C `be partitioned into` m `disjoint sets` $C_1, C_2, \ldots, C_m$ `such that, for` $1 \leq i \leq m$, $\sum_{a \in C_i} s(a) = \beta$?

Let us recall that 3-PARTITION is NP-complete in the strong sense. Let the finite set C of $3m$ elements, the bound β and the sizes $s(a_i)$ (with $\sum s(a_i) = m\beta$) be an arbitrary instance of 3-PARTITION denoted by I. We transform instance I into an instance I' of MD-NSP Emission problem. The ring of instance I' is composed of 3 nodes called n_1, n_2, n_3. K is equal to $m(\beta + 1)$. The set S of packets has $3m$ elements $p_1, \ldots, p_3m$ such that for $1 \leq i \leq 3m$, $sz(p_i) = s(a_i)$, $release(p_i) = 0$, $or(p_i) = n_2$, $dest(p_i) = n_3$. Moreover, we consider that at time 0, ring R is not empty but contains $m + 1$ packets of size 1 in transit:

- the first one (z_0) is in the slot located on node n_1,
- the other ones (z_i) are in slots located between n_1 and n_2, separated by β empty slots. So, after the node 1, we have β empty slots, the second packet (z_1), β empty slots, the third packet (z_2), and so one...
- the last packet (z_{m+1}) is thus located $m(\beta + 1)$ slots after the node 1, that is in the slot located on node n_2.

Let us remark that it would be possible to add these packets in transit to the list of the packets to send and to force them to be sent separated by β slots (corresponding to packets of β slots, stopping in n_1). But, for reasons of simplicity, we prefer considering the ring not empty at time 0 with these packets in transit.

The construction of our instance I' is completed by setting $B = K = m(\beta + 1)$.

It is easy to see how this construction can be accomplished in polynomial time. All that remains to be shown is that the desired partition exists if and only if there exists a scheduling having all the stated properties.

First, if there exists a partition $C' = \{C_1, C_2, \ldots, C_m\}$ such that for $1 \leq i \leq m$, $\sum_{a \in C_i} s(a) = \beta$, it is easy to compose a scheduling where the tasks corresponding to the elements of the subset C_i are scheduled in the i^{th} sequence of β empty slots. More precisely, if $C_i = \{a_{i^1}, \ldots, a_{i^{l_i}}\}$ where l_i is the number of elements of the subset C_i, we construct a scheduling such that:

for each $j \in [1, \ldots, l_i]$, $First(p_{i^j}) = (\beta+1)(i-1) + \sum_{\alpha=1}^{j-1} sz(p_{i^\alpha})$

As $release(p_i^j)$ is equal to 0, the over-delay of packet p_{i^j} is equal to $First(p_{i^j})$, that is less than $B = m(\beta+1)$ and we have thus constructed a scheduling having all the stated properties.

Conversely, let us suppose that there exists a scheduling S on R such that the over-delay over all packets of S is less or equal to B. The fact that for any packet p_i ($1 \leq i \leq 3m$), we have $OvDel(p_i) \leq B$ implies that all the packets are inserted before that the slot containing packet z_0 arrives to node n_2. Moreover, as the sum of the sizes of the packets is $m\beta$, all the slots available between z_0 and z_{m+1} have to be filled.

Now, we construct set C_i with the elements corresponding to the tasks scheduled in the i^{th} sequence of β empty slots: a_j belongs to C_i if: $(i-1)(\beta+1) < First(p_j) < i(\beta+1)$.

As the slots in this interval are filled by contiguous packets, the sum of the sizes of these packets is equal to β. So, we obtain a partition of C=$\{C_1, C_2, \ldots, C_m\}$ such that for $1 \leq i \leq m$, $\sum_{a \in C_i} s(a) = \beta$.

So, we have shown that the desired partition exists if and only if there exists a scheduling having all the stated properties. This concludes the proof.

Thus, we prove the NP-Completeness of this problem. We propose, in the next Section, polynomial distributed algorithms for the two models. Let us underline that no algorithm gives an optimal solution for this problem.

4. Study of the distributed model

We consider the ring network R as presented in Section 2. We consider the problem an a distributed and on-line context. In this section, we propose distributed and on-line algorithms and compare them in terms of delay and jitter.

We assume that for each packet P, $sz(P) < K \times N$ and that packets are sent using the FIFO strategy within each node.

4.Description of the distributed and on-line algorithms

4..1. A distributed and on-line algorithm for the SP model

In the SP model, packets can be considered as several packets of size 1 (see Section 2). A very simple distributed algorithm consists in sending a sub-packet as soon as a free slot crosses the node.

In each node, packets are considered using the FIFO strategy.

It is trivial that in an off-line SP distributed model, this distributed management algorithm ensures that all packets reach their destination in a finite time.

4..2. A distributed and on-line algorithm for the NSP model

For the NSP model, it is not possible anymore to use the simple distributed algorithm described above for the SP model. Indeed, when a node sends a sub-packet, it needs to be sure that there are enough consecutive free slots for the whole packet. So, at the beginning of each series of consecutive free slots, the size of this zone has to be indicated so that a node can verify that a packet of a given size may be emitted.

Thus, the ring is partitioned in successive zones of free or full slots. The first slot of each zone contains the heading -i.e. the length of the zone and its status. Let us underline that such a model implies that nodes emit packets only in the beginning of free zones.

So, when a node has a packet p to send and when the slot located on this node corresponds to the beginning of a free zone, the node checks that the length $sz(p)$ of the packet is less than the size sz of the free zone. In such a case, the packet is emitted in the first $sz(p)$ slots of the zone. If $sz(p) < sz$, a new free zone will be created after the emission of the packet: its heading is on the $(sz(p)+1)^{th}$ slot and its length $sz - sz(p)$. When a node receives a packet, the corresponding zone becomes free but its size remains unchanged (the status of the heading is modified but not the length).

The consequence of such a distributed control is that the ring can quickly become made of a sequence of very little zones, each one too short to be used by any node.

In order to limit that, the FIFO strategy is applied (as in the distributed algorithm for the SP model). In each node, only the first packet can be emitted. If the size of a free zone is not large enough for this packet, no packet is emitted by the node. This strategy avoids also that all the little packets are emitted first. But, besides this access control strategy, a distributed algorithm that merges contiguous free zones has also to be proposed.

The Merging-Algorithm

The purpose of this algorithm is to merge, at least, two contiguous free zones in one. It consists in updating the heading of the first free zone and in removing the heading of the following ones. After being merged, the length of the new free zone is equal to the sum of the lengths of the merged zones. We introduce the notion of reserving a free slot. When a slot is reserved, none of the node can write ino it. Only the node that reserves this slot can use it. As soon as a node manages to reserve adjacent free zones, it merges them in one in the next turn. Indeed, when a node reserves several adjacent free zones, the total length of these zones is stored and the zones are merged when the first zone comes back on the node that has reserved these zones. Thus, the status of the heading of a zone is free, full or reserved.

In order to distribute the reservation process, we have decided to apply the following rules :

- a node cannot reserve several non-adjacent free zones; it can only reserve one zone and the eventual free zones that follow this first reserved zone,

- any node can reserve a zone; as soon as a node has a packet to send and the length of this packet is greater than the length of the free zone, the node reserves the zone (if it has not reserved yet another non-adjacent zone).

However, with this algorithm, dead-lock situations are possible. Successive free zones may be reserved by different nodes and so never merged. A very basic illustration is given in Figure 2 with only two nodes, two free slots and two zones. We suppose that nodes A and B both have a packet of size 2 to emit. The first zone is reserved by node A while the other one is reserved by node B. These zones can never be merged and the packets are never sent.

This leads us to introduce the notion of prioritar node. A node can be prioritar if and only if it has at least one packet to send. This node has a priority on the other ones and can reserve a zone already reserved by another node. More precisely, this node reserves a zone already reserved by another node either when it has not yet reserved any zone or when the zone follows directly the zone it has reserved. However to ensure some fairness between the nodes, the node that is prioritar is periodically changed among the nodes that have packets to send. Let us remark that when the prioritar node reserves a zone already reserved, the first node that

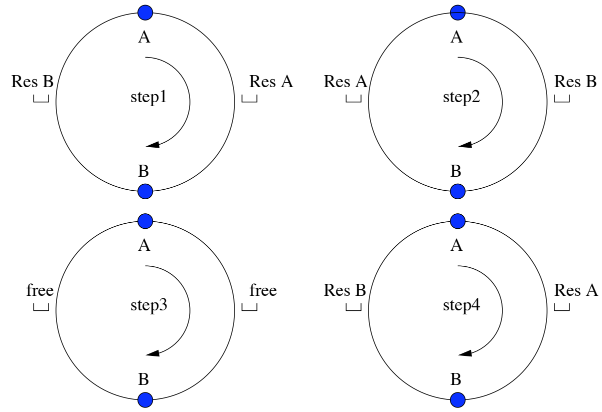

Fig. 2. Dead-lock in a one ring network

made the reservation sees the reservation by the prioritar node when the zone comes back on it and becomes then free to reserve another zone.

Lemma 1 *In an off-line NSP distributed model, the Merging algorithm ensures all packets are emitted and reach their destination in a finite time.*

Proof: Consider this lemma as false. Suppose that some nodes have still some packets to send but that these packets can not be sent. As all the packets emitted reach their destination in a finite number of step, all the zones finish by becoming free. Then, the only reason why none of the node can send its packets is because the free zones are not large enough to emit them. But, in such a case, the prioritar node reserves one of these zones and all the following ones and merges them all during the next turn. As the size of any packet is less than the number of slots ($K \times N$), at least one packet can be emitted that leads to a contradiction.

Finally, the distributed algorithm is the following one in which the variable *sizemerg* is a local variable located in each node that represents the global size of the zone that could be merged :

-

Merging-Algorithm

```
Si the slot located on the node N is the beginning of the zone Z Alors
|   Si Z − status = Full Alors
|   |   Si the destination of the packet inside is N  Alors
|   |   |   Receipt the packet - Z − status = free
|   |   FinSi
|   FinSi
|   Si Z − status = reserved Alors
|   |   Si Z has been reserved by N Alors
|   |   |   Si Z is the first zone reserved by N Alors
|   |   |   |   Z − status = free
|   |   |   |   Z − length = sizemerg(N)
|   |   |   Sinon
|   |   |   |   Remove the heading
|   |   |   FinSi
|   |   Sinon
|   |   |   Si N is the prioritar node Alors
|   |   |   |   Si N has not reserved any zone Alors
|   |   |   |   |   N reserves this zone
|   |   |   |   |   sizemerg(N) = Z − length
|   |   |   |   Sinon
|   |   |   |   |   Si Z is just after the zone reserved by N Alors
|   |   |   |   |   |   N reserves this zone
|   |   |   |   |   |   sizemerg(N) = sizemerg(N) + Z − length
|   |   |   |   |   FinSi
|   |   |   |   FinSi
|   |   |   FinSi
|   |   FinSi
|   FinSi
|   Si Z − status = free Alors
|   |   Si N has a packet to send Alors
|   |   |   Si the packet fits in Z Alors
|   |   |   |   N emits the packet
|   |   |   |   Z − status = full
|   |   |   Sinon
|   |   |   |   Si N has not reserved any zone Alors
|   |   |   |   |   N reserves this zone
|   |   |   |   |   Z − status = reserved
|   |   |   |   |   sizemerg(N) = Z − length
|   |   |   |   Sinon
|   |   |   |   |   Si Z is just after the zone reserved by N Alors
|   |   |   |   |   |   N reserves this zone
|   |   |   |   |   |   Z − status = reserved
|   |   |   |   |   |   sizemerg(N) = sizemerg(N) + Z − length
|   |   |   |   |   FinSi
|   |   |   |   FinSi
|   |   |   FinSi
|   |   FinSi
|   FinSi
FinSi
```

An additional point has been finally added to the algorithm. When a node emits a packet that does not fill completely the free zone, we search among all the packets in the buffer of the node if one packet with the same destination than the first packet could increase the completion of the zone. In such a case, the size of the full zone is considered as the sum of the lengths of the packets. This contradicts a little bit the FIFO strategy but slows down the division of the ring in small zones.

4.Simulation results

In this section, we present some simulation results. The purpose of this work is to compare SP model and NSP model.

We have considered a ring of 10 nodes and 9 slots between each pair of nodes. Thus, the nodes can use 90 slots to send their packets. We choose a network with few resources to send packets in order to study the network behavior with our distributed and on-line algorithm. We focus on three kinds of traffic. The first one, called *traffic A*, is characterized by a long period of small packets and short period of large packets. The second one, called *traffic B*, is characterized by a short period of small packets and a long period of large packets. The third one, called *traffic C*, is characterized by a short period of small packets and a short period of large packets. To be accurate, the packets are defined by different parameters (origin, destination, size, release date). The generation of new packets is simultaneous on all the nodes and occurs with a given and constant periodicity called inter-arrival-time. During one generation, one packet is created on each node. The destination of each packet is chosen uniformly among all the nodes. The length may be small (2 slots), or large (7 slots). This corresponds to the fact that in the DAVID project [10], two classes of packets has been identified: the short ones and the long ones (video data). Thus, in this model, the length follows a $MMUP2$ distribution (Markov Modulated Uniform Process with two states). Keeping the same average length of packets, we have realized simulations for different ratios of small and large packets in order to compare different uses of the network.

Let us notice that for different data traffics we keep the same data flows. Indeed, to compare our results it is important to have the same amount of packets per time unit generated.

The acceptable data flow : Let us present here our analysis of the data flow acceptable by the network. As we noticed before, the network is composed of 90 slots and 10 nodes. Then it is important to know the maximum acceptable amount of data per time unit under a uniform traffic. Consider that the network is uniformly used by the nodes. As long as destinations are randomly chosen, a packet sent stays in the ring on average one half turn. Then, each node can generate a packet of length 18 every 90 time unit. In other words, the maximum acceptable data flow by the network is 0.2 amount of data per time unit. This bound is reached in the SP model but not in the majority cases of the NSP model. During our simulations, we do stop to increase the data flow when the network is saturated.

Even if many measures can be investigated with such a simulation model, we have focused on two parameters: the over-delay and the jitter.

In all of the following figures the $X-coordinate$ represents the amount of packet-unit per time-unit within each node. For example, when $X - coordinate = 0.2$, during in period of 100 time-slot, each node generates a set of packets S which the sum of their size equals about 20. It can be associated to the flow rate entering in the network. The over-delay and the jitter average is counted in time-unit (see Section 1).

Over-Delay average of packets: In this part, we compare the over-delay average for the two models. In the SP model, the over-delay average can be more than 1000 times better than in the NSP one (see Figure 3). Indeed, the merging algorithm seems to cost a lot in terms of resources (number of slots). Let us notice we do this comparison for all the traffics A, B, C and we get the same results. So the average over-delay does not seem to be influenced by the data traffic so far. Of course, in the case where we have only packets of length 1 in NSP model, the over-delay would be better.

The over-delay seems not depend on the data traffic but only on the quality of the algorithm to merge free zones. The over-delay average obtained by using the Frame-algorithm described in [5], is much better than the one using the Merging-Algorithm in terms of over-delay average.

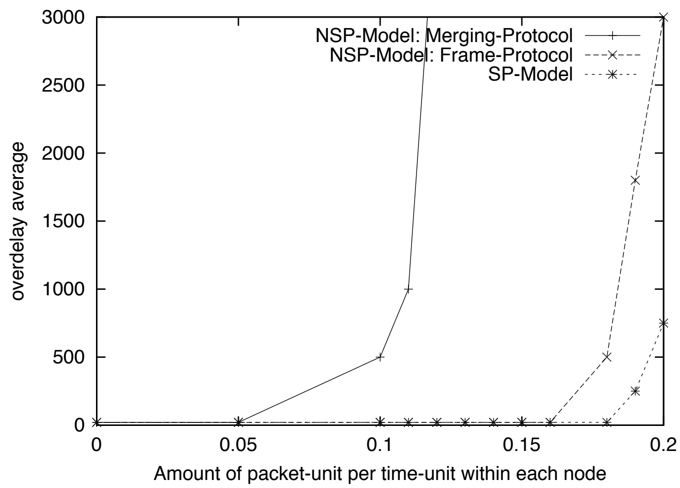

Fig. 3. Comparison of the Over-Delay Average of packets in different models

Behavior of the network in terms of average jitter in SP model : Previously, we saw that considering over-delay average, the SP model has a better performance than the NSP model. Indeed, in the SP model the constraint of jitter is removed, and packets can be sent using non contiguous free zones. Thus, it is easier for nodes to send their packets. But when a node receives the first sub-packet of a packet, it has no guaranty to receive the rest of this one in a bounded time. Then, the destination node has to store all uncompleted packets waiting for the rest of them. Remove the jitter constraint appears as a good solution when packets are inserted in the network but can be cost full for destination nodes.

In Figure 4, we present the average jitter of packets for three different traffics. The jitter is at worst about 35 time units, so about 1/3 of one turn. It is not that much considering the average over-delay value is more than 6000 time unit when the jitter is 35 in the SP model. To compare the two models, in the NSP one the jitter equals to $size(P) - 1$ and the average Over-Delay is greater than in the SP model.

As we can see in Figure 4, the jitter average increases according to the data traffic. In fact, the jitter average is better in the traffic C because there is a lot of small packets. Moreover, the performance decreases when the number of large packets increases. For example, traffic B has a more important number of large packets than the other two traffics.

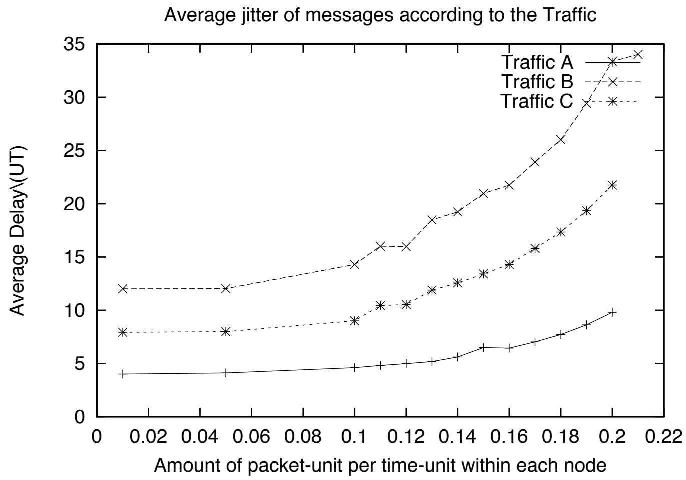

Fig. 4. Jitter Average of packets in SP model

5. Conclusion and Perspectives

Our study focus on the all-optical packet network. The topology of the network is simple : it is a single ring. Proving that finding optimal packets emission is NP-Complete in the NSP model considering the over-delay is our main contribution in this paper. Indeed, the problem is proved NP-Complete in a one ring network implies that the same problem in a multi-rings network is also NP-Complete.
Moreover, we have proposed and studied one algorithm for the SP model and one algorithm for the NSP one. Simulations show that greedy protocol in SP model gives better results than the two protocols in NSP model, in terms of over-delay.

A study of a multi-rings topology network under the same assumptions is given in [5]. In this research report, we present a comparison of the SP model and NSP model considering severals rings linked by one switch.

References and Links

[1] Henri Amet, Johanne Cohen, Freddy Deppner, Marie-Claude Portmann, and Stéphane Rousseau. Un problme d'ordonnancement de messages : Partie 1 modélisations ; partie 2 approches de résolution. *6me congrs de la Société Franaise de Recherche Opérationnelle et d'Aide la Décision - ROADEF'05*, 2005.

[2] D. Barth, J. Cohen, P. Fragopoulou, and G. Hébuterne. Wavelengths assignment on a ring all-optical metropolitan area network. *Proc. of 3rd Workshop on Approximation and Randomization Algorithms in Communication Networks*, 2002.

[3] A. Bianco, M. Bonsignori, E. Leonardi, and F. Neri. Variable-size packets in slotted wdm ring networks. *ONDM (Optical Network Design and Modelling)*, 2002.

[4] Peter Brucker. *Scheduling Algorithms.* Verlag, 2001.

[5] D.Barth, J.Cohen, L. Gastal, T. Mautor, and S. Rousseau. A comparison of splittable and non-splittable packet models in an optical ring network. Research Report 2005/83, Universite Versailles Saint Quentin France(78), 2005.

[6] Michael Garey and David Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. A series of books in the Mathematical sciences. W.H. Freeman and Co., 1979.

[7] IETF. Ip over optical networks: A summary of issues. 2000.

[8] ITU. Convergence between internet and optical networks. *ITU News Magazine n. 10*, 2003.

[9] Michael Pinedo. *Scheduling: Theory, Algorithms, and System*. Prentice-Hall, 2002.

[10] European project of the 5th PCRD. Data and voice integration over dwdm (david).

[11] B. Rajagopalan, D. Pendarakis, D. Saha, R.S. Ramamoorthy, and K. Bala. Ip over optical network: Architectural aspects. *IEEE Communications Magazine*, 2000.

[12] Yao, S.J. Ben Yoo, and B. Mukerjee. A comparison study between slotted and unslotted all-optical packet-switched network with priority-based routing. *networks.cs.ucdavis.edu/ syao/ofc01.pdf*, 2002.

[13] S. Yao, B. Mukherjee, S.J. Ben Yoo, and S. Dixcit. All-optical packet-switched networks: a study of contention-resolution schemes in irregular mesh network with variable packet size. *Proc. of Opticomm 2000*, 2000.

[14] X. Yu, Y. Chen, and C. Qiao. A study of traffic statistics of assembled burst traffic in optical burst switched networks. *www.cse.buffalo.edu/ qiao/wobs/ obs/papers/Yu_opti02.pdf*, 2003.

[15] A. Zapata, J. Spencer, I. de Miguel, M. Dser, P. Bayvel, D. Breuer, N. Hanik, and A. Gladisch. Investigation of future optical metro ring networks based on 100-gigabit ethernet. *ITG-Fachtagung "Photonische Netze"*, 2003.