

# Modelling and Performance Evaluation of a National Scale Switchless Based Network

Josep Solé-Pareta, Davide Careglio, Salvatore Spadaro, Jaume Masip,  
Juanjo Noguera and Gabriel Junyent

Universitat Politècnica de Catalunya - UPC,  
Advanced Broadband Communications Lab. - CCABA<sup>1</sup>  
Jordi Girona, 1-3, Mòdul D6 (Campus Nord),  
08034 Barcelona, Catalunya, Spain  
{pareta, careglio, sspadaro, jmasip, jnoguera}@ac.upc.es and  
junyent@tsc.upc.es

**Abstract.** In a packet switching based scenario, optical technologies can not completely overcome the problem of insufficient network capacity, due to limitations produced by the presence of electronic switches and regenerators in the fibre path. As a new alternative approach, the SONATA project proposed a switch-less based network with the triple objectives of avoiding the need for large and fast switching nodes and drastically simplifying both the network structure and the layer architecture within the network. This paper describes the simulation model and presents essential results from the performance evaluation of a Network Controller for a large-scale SONATA network. The simulations were carried out on a Silicon Graphics Origin-2000 using TeD (Telecommunication Description language) and the main objective was to assess the feasibility of switchless based networks to be used as large IP backbones.

## 1 Introduction

The SONANA network is based on the switchless network concept, which was defined and demonstrated in the SONATA (Switchless Optical Network for Advanced Transport Architecture) AC351 ACTS project. The "switchless" network concept is based on a mixture of WDMA (Wavelength Division Multiple Access based on wavelength tuneable transmitters and receivers) and TDMA (Time Division Multiple Access) methods to allow point-to-point and multi-point-to-multi-point interconnections between passive optical networks without the need for electrical switch nodes within the network. The general structure of this network is shown in Figure 1, and is described in detail in [1].

The key element of SONATA is the Network Controller. Figure 1 shows that a Network Controller is attached to the PWRN (Passive Wavelength Routing Node). This device is responsible for the resolution of access conflicts, for the allocation of

---

<sup>1</sup> Centre de Comunicacions Avançades de Banda Ampla



design). While scheduling is a simple task, which must be done slot-by-slot, the logical topology design is a much harder task, which can be performed at a lower rate.

This paper focuses on the performance evaluation of the algorithm proposed within SONATA for *scheduling the user requests*. The evaluation was done in terms of the *Request Loss Rate* and the achieved *Resource Occupancy*. Since the objective of the project was to evaluate the performance and global behaviour of the Network Controller for a large-scale (national and metropolitan scale) SONATA network, a theoretical study was not sufficient, but a simulation study was needed. The main advantage of the simulation process is that once the simulation elements are incorporated in the simulation layer, then easier comparison and optimization processes can be accomplished. Due to the adoption of a TDMA-AC protocol, simple traffic assumptions were sufficient.

The simulation environment used for this work was based on parallel computing simulation techniques. We used TeD [2], which stands for Telecommunications Description language. TeD is a network modelling language that is transparently mapped on top of a high-performance parallel computer. As a high-performance parallel computer we used a Silicon Graphics 2000.

This paper is organised as follows. In Section 2 the scheduling algorithm for the Network Controller proposed within SONATA is described. Section 3 is devoted to the simulation scenario we modelled to carry out our performance evaluation study. In Section 4 some samples of the simulation results are discussed. Finally Section 5 concludes the paper.

## 2 Scheduling Algorithm

This section is devoted to explain the basic concepts of the concrete scheduling algorithm we used within the SONATA Network Controller. The details of this scheduling algorithm can be found in [3].

The scheduling algorithm deals with *requests* and *frames of requests*. Each SONATA terminal generates requests for connection, and using a TDMA based approach all requests from SONATA terminals connected to the same control wavelength enter to the Network Controller as a frame. The Network Controller, using the scheduling algorithm, will decide which of the incoming requests are served or not.

The implementation of this scheduling algorithm is based on a set of matrices, which represents the network available resources. These matrices are (see Figure 2):

- **Matrix W:** Each entry of this matrix points out to a list of records describing the wavelength channels that provide connectivity between the corresponding PON-to-PON pair. The number of rows and columns is equal to the number of PONs.
- **Matrix S:** This matrix indicates, for each wavelength channel ( $\lambda_i$ ), which slots of the frame are already assigned.
- **Matrix Utx:** This matrix indicates for each transmitter (Tx) of  $\lambda_i$ , which slots of the frame are already assigned.
- **Matrix Urx:** This matrix indicates for each receiver (Rx), which slots of the frame are already assigned.

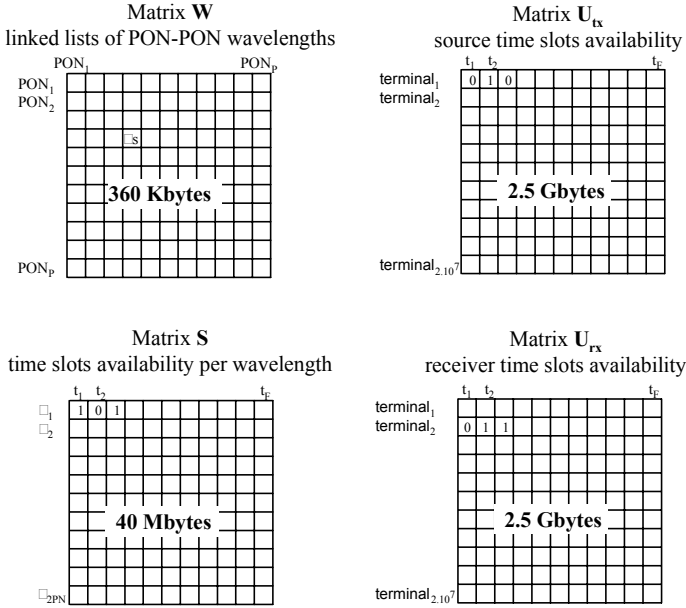


Fig. 2. The several matrices (data structures) used by the scheduling algorithm.

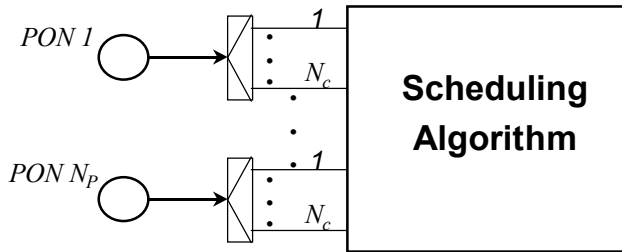
Based on the contents of these matrices, the scheduling algorithm is specified in four steps, which will be executed for each incoming frame. These four steps are the following:

1. Select the wavelength channel on which request is attempted; scan the linked list of wavelength channels associated with the pair (source PON, destination PON) in matrix W.
2. Select the time slot for use within the frame; logically AND the row of matrix S corresponding to the selected wavelength channel with the complement of the rows of source and destination matrices U<sub>tx</sub> and U<sub>rx</sub>.
3. Assign the number of requested slots to the requesting terminal, using a round-robin scan on the vector resulting from the previous step of the algorithm.
4. Update matrices S, U<sub>tx</sub> and U<sub>rx</sub> and descriptor in matrix W.

### 3 Simulation Scenario

Here we completely decouple the two problems of scheduling and logical topology design, assuming that the wavelength converters are reconfigured of-line and do not vary during the simulation time, and that the scheduling algorithm operates on a fixed logical topology. According to this assumption, a simulation model for the network control unit has been defined. This model consists of (see Figure 3)  $N_p$  request generators representing the PONs, which access the control unit through  $N_c$  TDMA signalling channels each, and the scheduling algorithm module. The requests

generators were implemented as random sources based on the Bernoulli random process, the number of signalling channels was set to  $N_c > 1$  in order to avoid the signalling bandwidth bottleneck, and the scheduling algorithm module was implemented based on the above described slot-by-slot scheduling algorithm. Since what we need to simulate is the case where each slot carries a request with probability  $\mathbf{p}$  and no request with probability  $\mathbf{q} = 1 - \mathbf{p}$ , the Bernoulli processes perfectly model the TDMA signalling channels.



**Fig. 3.** Simulation model for the SONATA network controller.

Two types of traffic modes were implemented, namely the balanced and the unbalanced types. For the balanced type, a uniform distribution of the destination PONs was used (all PONs has the same probability to be destination of the traffic), while for the unbalanced type a non-uniform distribution of the destination PONs was used.

### 3.1 Network Configuration

In order to reproduce a real scale configuration of the SONATA network that could fit in the bounds of the simulation platform, we could not consider the FTTH solution where end-user and SONATA terminals are equivalent terms. We assumed either the FTTCab or the FTTK/B solutions with 200 end-users per SONATA terminal. According to this assumption, in the simulator we set up a network configuration with 100 PONs ( $N_p = 100$ ) and 1000 SONATA terminals per PON ( $N_{st} = 1000$ ). These values lead to 20 millions customers ( $100 \times 1000 \times 200 = 2 \cdot 10^6$ ), which within the project was considered as a national range, but reduces the amount of memory required by matrices  $W$ ,  $S$ ,  $U_{tx}$  and  $U_{rx}$ . The rest of the configuration parameters was the following: 1000 signalling mini-slots per frame ( $K_s = 1000$ , one per terminal in each PON),  $N_c = 3$  or 10 (depending on the simulations) signalling channels per PON, 100 data slots per frame (frame duration, 1 ms.), and a transmission bit rate per channel of 622 Mbps. Finally, the number of dummy ports was fixed either to 0, 100, 200, 300 or 400 (0, 1, 2, 3 or 4 additional wavelengths per PON-to-PON pair).

### 3.2 Traffic Characteristics

We established the following traffic conditions: Only Internet traffic (no connection oriented traffic load). Internet traffic requests modelled by the Bernoulli generators with an offered load per PON  $\mathbf{p} = 0.6, 0.75, \text{ and } 0.9$ . Concerning the distribution of

the data traffic destination PONs. 1) For the balanced traffic type, a uniform distribution among all the PONs with mean 1/100 was used. 2) For the unbalanced traffic type, a Gaussian distribution, with mean = 50 (this means that numbering the PONs between 1 and 100, the most probable destination of the traffic would be the PON number 50) and standard deviation = 20.

## 4 Experiments and Results

Under the configuration and traffic conditions described in the previous section, we carried out a set of simulations assuming two different approaches. A per packet request approach (requests were issued by the IP layer, one per IP packet to be transmitted), and a per packet-flow request approach (requests were issued from the TCP layer, one per TCP session). In each case we simulated 10 seconds of the network operation time, which required an execution time of around 36 hours.

### 4.1 Per Packet Request Approach

In the per packet request approach, we set the number of slots required per request, based on the IP packet length statistics provided in [4]. Thus, we considered that 70% of the requests would ask for 1 slot and the rest (30%) would ask for 2 slots.

Within this approach, the Network Controller serves the requests in a frame by frame basis, i. e., a request can only be served if the 1 or 2 slots which that request needs can be allocated in the data frame which is being scheduled at that moment. If not the request is lost.

Figure 4 shows a sample of the simulation results obtained for the case of unbalanced traffic type,  $N_c = 10$ , and up to a total number of 200 dummy ports.

Note that  $\mathbf{p}$  is the requests offered load (to the signalling channels), while  $\square$  is the throughput requested from the network (i. e. the offered load in terms of resources required from the network over the maximum resources available). For the case of 200 dummy ports, the maximum value of  $\square$  is 0.43. This is because the total capacity of this network configuration is:

$$\begin{aligned} & [(100 \text{ direct ports} + 200 \text{ dummy ports}) 100 \text{ wavelength per port}] 622 \text{ Mbps} = \\ & = 18.66 \text{ Tbps} \end{aligned}$$

while the maximum requested capacity (when  $\mathbf{p} = 1$ ) is:

$$\begin{aligned} & 10^9 \text{ requests per second} \times 1.3 \text{ required slots per request} \times 6220 \text{ bit per slot} = \\ & = 8.086 \text{ Tbps} \quad (1.3 \text{ is the average of required slots per request} = 0.7 \times 1 + 0.3 \times 2) \end{aligned}$$

thus,

$$\square(\max) = 8.086 \text{ Tbps} / 18.66 \text{ Tbps} = 0.43.$$

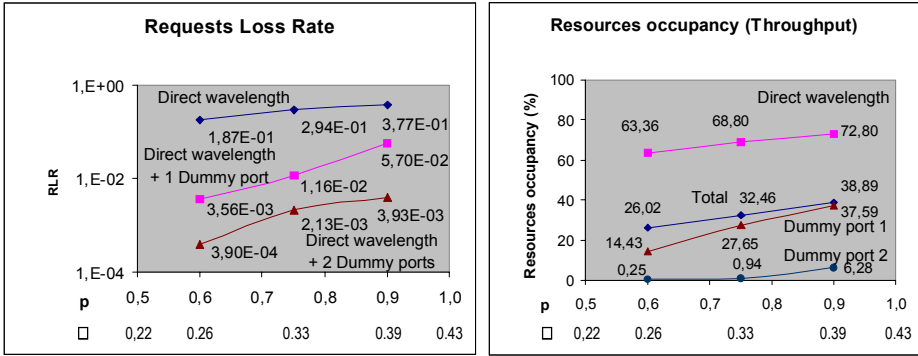


Fig. 4. Sample of Request Loss Rate and Resource Occupancy when using a per packet request approach (using two wavelength converters plus the direct wavelength per PON).

Figure 4 shows that the SONATA network exhibits low efficiency of its transmission capacity. This is mainly due to the fact that the simulation model is based on the per packet request approach. Using this approach, the signalling bandwidth easily becomes a bottleneck. In this particular case (using two wavelength converters plus the direct wavelength), we can only ask for a 43 % ( $\square = 0.43$ ) of the total capacity of the network. This is a little bit better than when only a single wavelength converter is added to the direct wavelength (see Figure 5). In this case, a higher resource occupancy can be reached (up to 65 %), but then the Request Loss Rate performance degrades drastically.

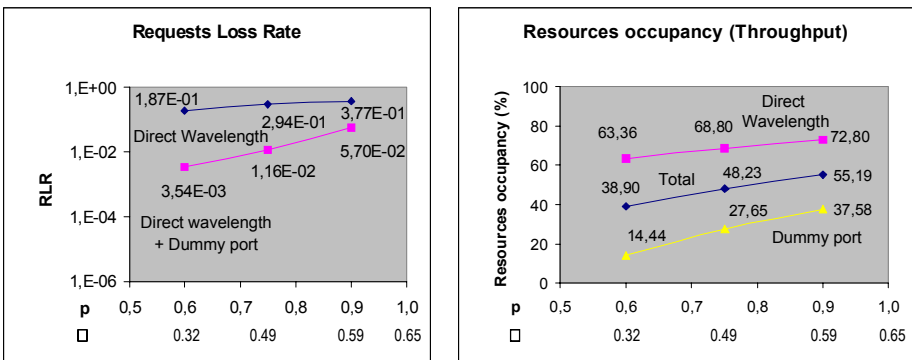


Fig. 5. Sample of Request Loss Rate and Resource Occupancy when using a per packet request approach (using a single wavelength converter plus the direct wavelength per PON).

To solve this problem without trying to enhance the MAC protocol established for signalling, either we increase the number of signalling wavelengths ( $N_c$ ) or we reduce the amount of signalling by issuing per packet flow requests instead of per individual packet requests. The former is easy to implement, but then in practice we can come to the problem that, in such a big network, the computing requirements to execute the

scheduling algorithm exceed the current available limits. Thus, as mentioned before, in our simulator we decided to implement the per packet-flow request approach option. This option is also not realistic since (at least as of today) signalling can not be implement at TCP/UDP level –only *ftp* announces the amount of data to be transferred when the TCP session is set up. We any way assumed such a visionary scenario because it completely avoids the bottleneck-signalling problem, and facilitate to cope with our main objective, namely to test the SONATA potential capabilities.

## 4.2 Per Packet-Flow Request Approach

In order to carry out the simulations to assess the feasibility and performance of the network control unit using a per packet-flow request approach, we assumed that the traffic in the Internet has the following composition: 42% of the traffic is *http* (35%) plus *ftp* (7%). 45% of the traffic is *irc* (23%) plus *telnet* (7%) plus *smtp* (5%) plus games (5%) plus other applications (5%) which typically implies low data transfer. The rest (around 13%) is IP over IP traffic (tunneling). We also assumed that within the *http* plus *ftp* traffic, 20% of flows in average require 2 slots, 21% of flows in average require 13 slots, 6% of flows in average require 130 slots, and finally 0.5% of flows in average require 1300 slots. For the *irc* plus *telnet* plus *smtp* plus games etc. traffic we assumed that 100% of flows require a single slot. And finally for the IP over IP traffic we considered as it has the same composition than the ordinary IP traffic (i. e., 70% of packets require 1 slot while 30% require 2 slots). According to these assumptions, the number of required slots per request distribution we adopted in our simulations was 46% of requests 1 slot, 23% of requests 2 slots, 23% of requests 13 slots, and 8% of requests 130 slots.

In contrast to the per packet request approach, in the per packet-flow request approach, the requests are served by allocating the requested slots in consecutive data frames. For example, consider the case of a request needing for 2 slots. If this request can be served, the first slot is allocated in the data frame, which is being scheduled at that moment, and for the second slot a reservation is made to allocate it in the next data frame.

Figure 6 shows a sample of the results obtained in the per packet-flow request approach simulations. These results correspond to the case of unbalanced traffic type, number of signalling channels  $N_c = 3$ , and up to a total number of 400 dummy ports. The total capacity of the network configuration, in this case, is:

$$\begin{aligned} & [(100 \text{ direct ports} + 400 \text{ dummy ports}) 100 \text{ wavelength per port}] 622 \text{ Mbps} = \\ & = 31.10 \text{ Tbps} \end{aligned}$$

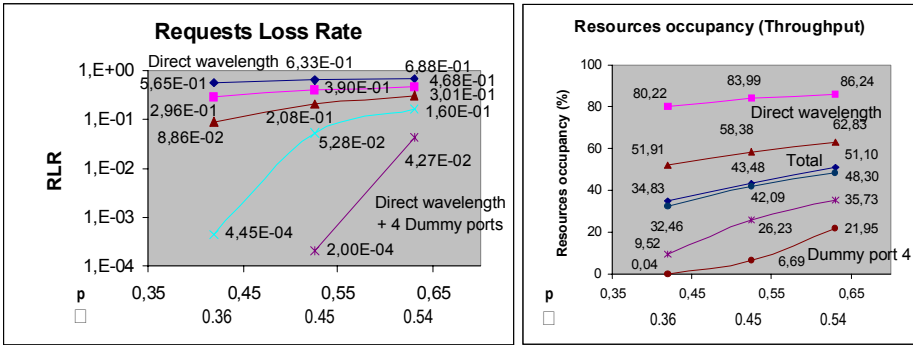
while the maximum requested capacity (when  $\mathbf{p} = 1$ ) is:

$$\begin{aligned} & 3 \times 10^8 \text{ requests per sec.} \times 14.31 \text{ required slots per request} \times 6220 \text{ bit per slot} = \\ & = 26.702 \text{ Tbps} \text{ (14.31 is the average of required slots per request, i.e.,} \\ & 0.46 \times 1 + 0.23 \times 2 + 0.23 \times 13 + 0.08 \times 130) \end{aligned}$$

thus,

$$\square (\max) = 26.702 \text{ Tbps} / 31.10 \text{ Tbps} = 0.86.$$



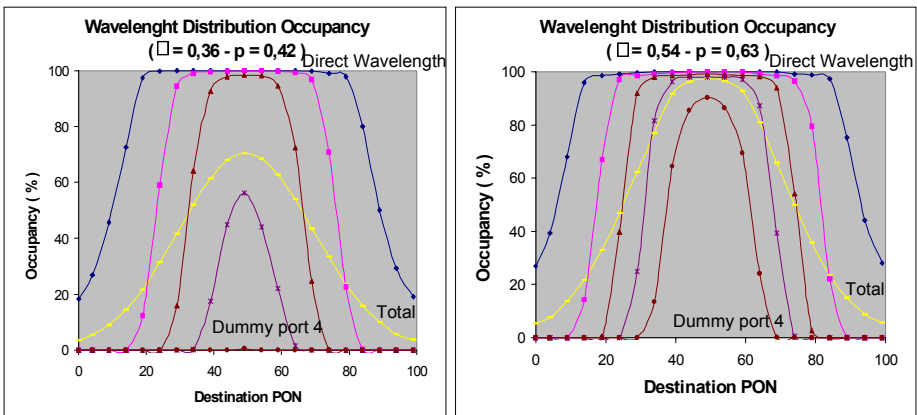


**Fig. 6.** Sample of Request Loss Rate and Resource Occupancy. Per packet-flow request approach and unbalanced traffic type.

Figure 6 shows that using the per packet-flow request approach, the network capacity can cope with the offered load, i. e. in this case  $\rho$  is closer to  $p$  than in the case of per packet request approach. Nevertheless, the performance is again very bad. As an example in Figure 6 it can be seen that for the case of 51.1% of the resource occupancy (a very low occupancy), the Request Loss Rate is  $4.27 \cdot 10^{-2}$ , which is a very bad figure. This is due to the fact that the wavelengths of the dummy ports were uniformly assigned to the destination PON, while the destination traffic distribution was Gaussian with a traffic concentration between PONs from 30 to 70. Figure 7, which includes the per PON resource occupancy distribution, shows the effect of this traffic concentration.

Both diagrams of Figure 7 show that wavelengths of the dummy ports assigned to PONs from 30 to 70 are overloaded, while wavelengths of the dummy ports assigned to PONs from 1 to 29 and from 71 to 100 are almost unused.

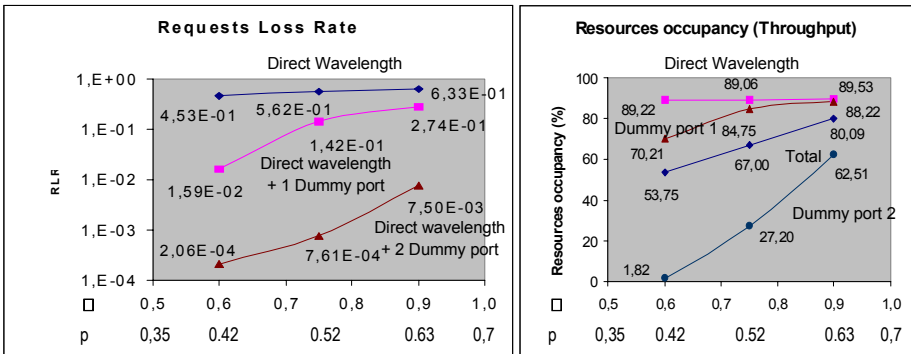
Obviously, a better performance can be obtained by an unbalanced assignment of the wavelengths to PONs, i. e., assigning more wavelengths to reach PONs from 30 to 70 than to reach PONs from 1 to 29 and from 71 to 100. This is shown in the next



**Fig. 7.** Per PON wavelength occupancy distribution.

experiment, which results are summarised in Figure 8. Figure 8 includes a sample of the simulation results obtained with the per packet-flow request approach for the case of balanced traffic type,  $N_c = 3$ , and up to a total number of 200 dummy ports.

Figure 8 shows that, when the wavelengths of the dummy ports to the destination PONs assignment fits with the distribution of the traffic destination PONs, SONATA exhibits a very good performance. Note that even when reaching a high resource occupancy, as for example 80%, the Request Loss Rate stays below  $10^{-2}$  ( $7.5 \cdot 10^{-3}$  in the case of a total resource occupancy of 80%). This is, in fact, the effect that would result from a joint collaboration of the scheduling algorithm and a dynamic logical topology design algorithm, which would be able to adapt the wavelength converters to the instantaneous traffic requirements.



**Fig. 8.** Sample of Request Loss Rate and Resource Occupancy. Per packet-flow request approach and balanced traffic type.

Figure 8 depicts, in our opinion, the most representative performance figures of the potential capabilities that can be obtained from, in general, a large scale switchless based network solution, and in particular from SONATA using a heuristic scheduling algorithm.

## 5 Conclusions

SONATA Network is a switchless single layer optical transport network designed to cover national and metropolitan areas, which exemplifies a possible vision of future optical Internet backbones. In this paper we have presented some performance evaluation results obtained by simulation, which provide a picture of the potential capabilities of a national scale configuration of SONATA network based on a simple heuristic algorithm of scheduling the user transmission requests that was adopted within the SONATA project.

This picture shows that the SONATA approach is feasible and can provide good Resource Occupancy and Request Loos Rate figures. Nevertheless, In order to reach these figures, apart from overcoming the signalling-bottleneck problem, a dynamic assignment of the dummy ports wavelength to the PONs (dynamic logic topology design) algorithm has to be used, jointly with the scheduling algorithm, to implement the Network Controller. Such an algorithm was already proposed last year of the

SONATA project and is currently incorporated to the SONATA simulation platform. A future version of this paper will include the simulation results that will be obtained from the new platform.

## Acknowledgements

This work is part of the SONATA project (AC351), partially funded by the European Commission under the Advanced Communications Technology and Services (ACTS) program. This work was also partially funded by CICYT (Spanish Ministry of Education) under contract TIC99-0572-C02-02. The authors thank CEPBA (the European Center for Parallelism of Barcelona) for allowing them to use the Silicon Graphics 2000. The authors also thank Dr. Alan Hill of British Telecom for his comments and suggestions.

## References

1. Nuncio P. Caponio, Alan M. Hill, Fabio Neri and Roberto Sabella, "Single Layer Optical Platform Based on WDM/TDM Multiple Access for Large Scale "Switchless" Networks", *European Transactions on Telecommunications*, Vol. 11, No. 1, January-February 2000.
2. S. Bhatt, R. Fujimoto, A. Ogielski, K. Perumalla, "Parallel Simulation Techniques for large-scale Networks", *IEEE Communications Magazine*, pp. 42-47, August 1998.
3. A. Bianco, E. Leonardi, M. Mellia, M. Motisi and F. Neri. "Initial Considerations on the Implementation of Control Functions in SONATA", Working Document, ACTS project AC351 SONATA, Politecnico di Torino, 1998.
4. Kevin Thompson, Gregory J. Miller and Rick Wilder, "Wide-Area Internet Traffic Patterns and Characteristics", *IEEE Network*, November-December 1997.