Enhanced Parallel Iterative Schedulers for IBWR Optical Packet Switches

M. Rodelgo-Lacruz*, P. Pavón-Mariño+, F. J. González-Castaño*, J. García-Haro+, C. López-Bravo* and J. Veiga-Gontán+

*University of Vigo, Spain, {mrodelgo,javier,clbravo}@det.uvigo.es
+Polytechnic University of Cartagena, Spain, {pablo.pavon,joang.haro}@upct.es, javg@alu.upct.es

Abstract. In this paper we propose an enhanced parallel iterative scheduler for IBWR synchronous slotted OPS switches in SCWP mode. It obtains a maximal matching of packet demands without resource conflicts. The analytical and numerical results are highly competitive regarding previous work.

Keywords: OPS, IBWR, Scheduling Algorithms.

1 Introduction

In the Optical Packet Switching (OPS) paradigm of Wavelength Division Multiplexing (WDM), packet payloads stay in the optical domain. OPS offers high bandwidth efficiency due to statistical multiplexing, but it is well-known that packet granularity and optical buffering impose extreme constraints to photonic switching, incurring in unacceptable hardware costs with state-of-the-art technology.

In this paper, we focus on synchronous slotted OPS in Scattered Wavelength Path (SCWP) operational mode [1]. This mode specifies a fixed packet size (slot length) and packet alignment with slot boundaries at the input ports (and thus optical synchronizing stages, which increases cost). However, the performance improvement due to the better contention behavior has encouraged the study of this alternative. Packet length in OPS networks is a current topic of discussion. The European DAVID project [2] selected synchronous slotted OPS with slot lengths of ~1 μs for the WDM backbone network. In WDM OPS networks, there is a mapping of permanent end-to-end connections to link wavelengths. In the SCWP operational mode, optical packet paths (OPP) univocally determine a fixed sequence of transmission fibers, but the transmission wavelength may change in each hop. This provides extra freedom to switch schedulers in packet wavelength selection, boosting the statistical multiplexing effect. Therefore, SCWP achieves a high throughput with a low packet delay in OPS networks, also lowering optical buffering requirements [3][4].

In SCWP it is possible to *simultaneously* transmit several packets of the same OPP through a fiber, in different wavelengths. In this paper we adopt the round-robin packet ordering criterion in [5] that avoids the performance degradation due to unbalanced wavelength assignments. The wavelengths are assigned cyclically. Each node uses two sets of round-robin pointers to track packet sequence: one round-robin pointer per input fiber, tracking the wavelength of the next packet in the input traffic sequence, and one round-robin pointer per output fiber, determining the output wavelength of the next packet to be transmitted. Figure 1 shows an example.

This paper focuses on the Input-Buffered Wavelength-Routed (IBWR) switch architecture, for its scalability. Figure 2 shows the WDM adaptation of this architecture [6]. The switch has N input/output fibers, and n wavelengths per fiber. It has a buffering section followed by a non-blocking switching section. The buffering section consists of $n{\cdot}N$ Tunable Wavelength Converters (TWC) with a tuning range $\lambda_0 ... \lambda_{K-1}$, K=max ($n{\cdot}N,M$) and two $K{\times}K$ Arrayed Waveguide Gratings (AWGs), which are interconnected by M delay lines of 0 to $M–1$ slots. Due to AWG symmetry, a packet arriving at port i leaves the buffering section at port i, regardless of the selected delay. The wavelength conversion determines the delay line for the packet. The switching section is composed of $n{\cdot}N$ TWCs followed by a $nN{\times}nN$ AWG. The switching AWG routes each packet to the proper output fiber/wavelength.

The IBWR switch scheduler assigns packet delays and packet output wavelengths. These two tasks are independent.

- *Packet delay assignment*. Current optical switches employ Fiber Delay Lines (FDLs) due to the lack of optical RAMs. In IBWR switches, delays are assigned at packet arrivals. The scheduler discards a packet if it cannot assign a delay fulfilling two contention conditions: *(i) output fiber contention*: at most n packets can reach any output fiber in a given slot, *(ii) input port contention*: the packets that arrive to the same i-th input port (same fiber and wavelength) in different time slots cannot leave the switch in the same time slot. Otherwise they would collide at the i-th TWC of the switching section, which can only handle one packet at a time.
- *Output wavelength assignment*. The scheduler assigns output wavelengths to the packets when they leave the switch, according to the round-robin criterion.

Remark: Other OPS architectures, with higher hardware costs and less scalable than IBWR, emulate output buffering (OB) [6][7] (the only factor limiting packet delay assignment is output fiber contention).

Previous work has characterized IBWR delay assignment as a matching in bipartite graphs [4]. At every slot, the scheduler seeks a feasible assignment maximizing the number of packet delay assignments (i.e. minimizing packet losses). If there are several alternatives, it minimizes average packet delay. The *sequential* IBWR scheduler for the SCWP mode in [8] is unfeasible in practice (for ~1 μs slots). Conversely, our proposal is parallel, as Virtual Output Queuing (VOQ) schedulers.

The rest of this paper is organized as follows: in section 2 we describe the Parallel Desynchronized Block Matching Scheduler (PDBM), which is the basis for this proposal. In section 3 we present the Insistent PDBM (I-PDBM) algorithm. In section 4 we discuss simulation results. Section 5 concludes the paper.

2 PDBM scheduler

PDBM [9][11] was the first parallel iterative matching scheduler for IBWR switches. We reproduce it here since it is the basis for the enhancements in this paper.

Figure 3 shows an electronic PDBM implementation. The *nN* input modules (one per input fiber wavelength) are interconnected with *NM* output modules (one per output fiber and delay line) by three types of signals. The main ones are the *request* signals, from input to output modules, and the *grant* signals, from output to input modules.

Input module *i, i*=0,...,*nN*–1, keeps an input TWC availability state vector $\overline{x_i}\,(t)$, $t=0,...,M-1$. Component $\overline{x_i}\,(t)$ equals 1 if a packet is scheduled to leave the buffering section at the *i*-th port in *t* slots (0 otherwise). At every slot the state vector is shifted: $\overline{x_i}\,(t\text{-}1)=\ \overline{x_i}\,(t)$ and $\overline{x_i}\,(M\text{-}1)=0$, to reflect FDL propagation after each slot.

Output module (*j,t*), *j*=0,...,*N*–1, *t*=0,...,*M*–1, keeps: (a) a value $n-\ y_{jt}$ (delay availability) of *log*$_2$(*n*) bits. Variable y_{jt} denotes the number of packets for output fiber *j* that will leave the switch in *t* time slots; (b) a *grant pointer* FG_{jt}, of *log*$_2$(*N*) bits. It indicates the first input fiber in the scan; (c) an alternating bit CW_{jt} indicating the search direction. Note that, at every slot, the delay availability register in module (*j,t*) must be transferred to module (*j,t*–1), *j*=0,...,*N*–1, *t*=1,...,*M*–1. Also, modules (*j,M*–1), *j*=0,...,*N*–1, reset the availability registers to *n*.

At each input fiber controller, a round-robin grant pointer WG_f, *f*=0,…,*N*-1, indicates the first wavelength to scan in input fiber *f*.

PDBM Algorithm

At system initialization, $\overline{x_i}\,(t)$, y_{jt}, WG_f and CW_{jt} are set to 0. All FG_{jt} grant pointers associated to the same output fibers are initialized by maximizing the minimum distance between pointer positions:

$$FG(f,0) = 0$$

$$FG(f,t) = FG(f,t-1) + \min\left(1, \left\lfloor \frac{N}{M} \right\rfloor\right) \quad \begin{matrix} \forall f = 0 \ldots N-1 \\ \forall t = 1 \ldots M-1 \end{matrix}$$

Algorithm iterations consist of three steps (*request, grant,* and *accept*):

Step 1. Request: Each input module *i* with a packet for output fiber *j* sends a request signal to every output module in fiber *j* whose associated delay satisfies the input contention constraint. That is, output modules (*j,t*) such that $x_i(t)$=0.

***Step 2. Grant*:** Each (*j,t*) output module scans the request signals from the input modules, starting by the input module indicated by grant pointers FG_{jt} and WG_f. The scans from other input modules proceed in a clockwise or counter-clockwise sense, according to the alternating bit CW_{jt}. The first $n-\ y_{jt}$ scanned request signals are acknowledged, and a grant signal is sent to the associated input module.

Step 3. Accept: Each input module *i* receives at most *M* grants, from the *M* delays associated to the destination output fiber. The shortest granted delay *t* is accepted and assigned to the packet that is present at input *i*. If the input does not receive any grants during algorithm execution, the packet is discarded. Otherwise, an accept signal is sent to the accepted output module and the $\overline{x_i}\,(t)$ and y_{jt} state vectors are updated to reflect packet allocation. When a packet is granted, its input port does not participate in subsequent algorithm iterations.

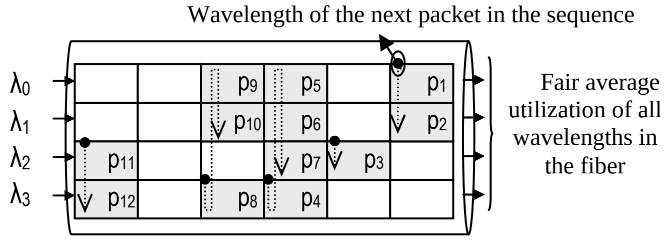

Wavelength of the next packet in the sequence
λ0
λ1
λ2
λ3
p9
p5
p1
p10
p6
p2
p11
p7
p3
p12
p8
p4
Fair average utilization of all wavelengths in the fiber

Fig. 1. Round-robin wavelength sequence criterion, fiber with four wavelengths λ_0,..., λ_3.

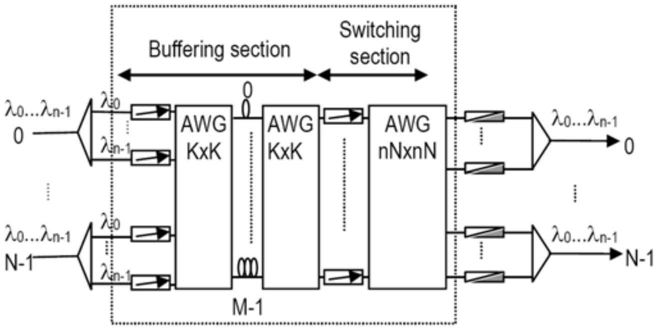


Fig. 2. Adaptation of the Input-Buffered Wavelength-Routed switch (IBWR) to WDM.

At each time slot, after the last iteration, $\overline{x_i}\,(t)$ and y_{jt} are updated and shifted as described above to consider the allocation and the propagation of the packets in the delay lines. The CW_{jt} bits are negated to alternate request scanning directions each time slot. The FG_{jt} grant pointers are incremented by one (module N), every *two* time slots and the WG_f round-robin grant pointers are incremented by the number of received packets at fiber f in the current slot (modulo n).

Algorithm justification

PDBM converges in $min(M,nN)$ iterations at most [9]. Thus, convergence speed is independent from switch size.

The initialization of the pointers and their evolution are inspired by the desynchronizing scheme of the RDSRR [10] algorithm to minimize the grant block overlapping effect: if an output module (j,t) receives more requests than available delays, it only acknowledges signals from the modules whose indexes are closest to grant pointer FG_{jt}. If the grant pointers take the same value, "close" input modules receive several grants, and "far" input modules receive no grants at all. In PDBM, all grant pointers of a given output fiber get initial values that maximize the minimum distance (modulo N) between two input nodes. The scheduler keeps the desynchronization by increasing (modulo *N*) all pointers every two time slots. The scanned direction is inverted at each time slot to enforce fairness in case of non-uniform packet arrivals.

Although PDBM does not guarantee packet sequence, input modules are scanned following the round-robin criterion to mitigate mis-sequencing.

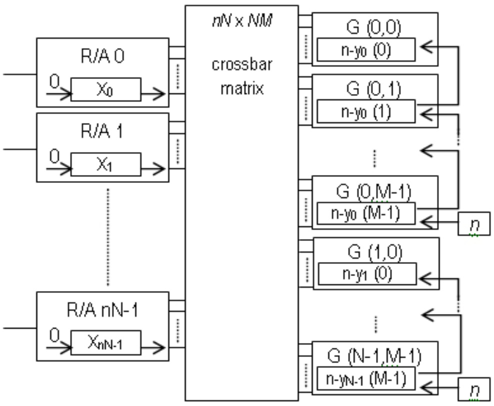


Fig. 3. Electronic implementation scheme for the PDBM scheduler.

3 Insistent PDBM (I-PDBM) scheduler

The basic PDBM scheduler may assign longer delay lines than strictly necessary, ignoring shortest ones even in absence of contention. Specifically, it converges to a maximal size match (no more connections can be established without replacing any existing connections) with suboptimal aggregated delay, i.e., some connections could be individually removed and reassigned to a shorter delay output port. We call this effect “PDBM impatience”. We will illustrate it with an example:

Let us assume a switch with two inputs (outputs), two wavelengths per fiber and three delay lines (*N*=2, *n*=2, *M*=3). Two packets arrive at input fiber 0 requesting output fiber 1. The state of the switch is:

- FG_{ft} and CW_{1t} are indifferent because there are no packets in fiber 1.
- WG_0 = 0. The round-robin grant pointer of input fiber 0 points to input module 0. The first input wavelength of fiber 0 to be scanned is 0 for all iterations.
- $x_0(t)$=$x_1(t)$=0 $\forall t$. No input contention.
- y_{10}=1, y_{11}=1, y_{12}=0. There is a free delay line for *t*={0,1} and two delay lines for *t*=2.

Figure 4 summarizes the state of the node. From that state, the algorithm iteration evolves as follows (figure 5):

***Request*:** input modules 0 and 1 send request signals to all output modules *(1,t)*, since there is no input contention at them.

Grant:

- t=0. Output module (1,0) scans the signal of input module 0 (WG_0=0) and acknowledges it. The request signal of input module 1 is not acknowledged because there is a single available wavelength, n–y_{jt}=1 (2-y_{10}=1).
- t=*1*. Output module (1,1) scans the request signal of input module 0 (WG_0 = 0) and acknowledges it. The request signal of input module 1 is not acknowledged because there is a single available wavelength, n–y_{jt}=1 (2-y_{11}=1).
- t=*2*. Output module (1,2) scans the signals of input modules 0 and 1, and acknowledges them both because there are two available wavelengths, n–y_{jt}=2 (2-y_{12}=2).

Accept. Input module 0 receives three grant proposals (t = 0,1,2) and accepts the best one (delay 0). Input module 1 receives a single grant proposal (t=2) and accepts it. The packet at input 1 is assigned to delay line 2. However, there is room in delay line 1, which has no input contention. Thus, the assignment is suboptimal. To solve the impatience problem we propose a new algorithm: **Insistent PDBM or I-PDBM**.

I-PDBM algorithm

In the PDBM accept step, a granted input module confirms the received grant to update the state vector y_{jt} and deactivates the other request signals to allow input ports with lower priorities to be granted. It is possible to simplify the algorithm to execute a single accept step after the last iteration. It suffices to change input modules to keep the request signal active for the "accepted" grant and to deactivate all others. Since the number of wavelengths does not decrease and the pointers do not change until the accept step at the end of the slot, each granted input module that keeps an active request signal is granted again, whereas the unrequested granted delays are released and reassigned to other input modules.

The previous simplified scheme easily solves PDBM impatience if each granted input module stops requesting higher delays but it keeps the request signal active for better ones. By stopping higher delay requests, it releases some wavelengths that can be granted to other modules. Subsequent iterations may increase the number of packet assignments and further reduce the delay of previously assigned packets. Thus, grants are *provisional*, until the very last iteration when the accept stage takes place.

Therefore, the differences with PDBM are:

Step 1. *Request*: each input module i with a packet destined to output fiber j sends a request signal to every output module of fiber j whose associated delay satisfies that the input contention constraint is not worse than any granted delay to the same input module in the previous iteration, i.e. the input module sends request signals to output modules (j,t) such that $\overline{x_i}$ (t)=0 and $j \leq p$, where p is the shortest granted delay.

End of the algorithm. *Accept*: the accept step takes place after the last iteration. So, state vectors and pointers are not updated until the end of the time slot and the granted input modules participate in subsequent algorithm iterations.

Destination of packets that arrive in the current time slot			
Input fiber 0		Input fiber 1	
Wavelength 0	Wavelength 1	Wavelength 0	Wavelength 1
Output fiber 1	Output fiber 1	No packet	No packet

Delay lines of output fiber 0	
Delay length (time slots)	Delay availability (packets)
0	1
1	1
2	2

Fig. 4. Node state information.

Signals from input fiber 0 to output fiber 1	Request		Grant		Accept	
Delay	λ_0	λ_1	λ_0	λ_1	λ_0	λ_1
0	yes	yes	yes		yes	
1	yes	yes	yes			
2	yes	yes	yes	yes		yes

Fig. 5. Algorithm stages.

Algorithm justification

I-PDBM converges when the signals get stabilized, i.e. there are no new packet allocations nor assignments of better delays to granted packets. As PDBM does, I-PDBM converges in min(M,nN) iterations at most to a maximal size matching. ***Proof***: *i*) an output port only changes a grant signal if a previous input port (according to the grant pointers) releases a request signal. An input port only releases a request signal if it received a grant in the previous iteration from an output port that is associated to a shorter delay. Since the grants from delay 0 do not change after the first iteration, the algorithm converges in M iterations at most. *ii*) An input port is granted a shorter delay only if another input port was granted a shorter delay in the previous iteration. Since there are nN input ports, the algorithm converges in nN iterations at most.

I-PDBM avoids PDBM impatience and it is simpler to implement. It has two steps (request and grant), whereas PDBM needs three (request, grant and accept).

4 Results

In this section we present simulation results to compare I-PDBM (in terms of average delay, buffer requirements and practical convergence) with OB architectures and the previous PDBM algorithm, under benign or bursty traffic conditions.

Figures 6(a) and 6(b) show the average delay of I-PDBM and PDBM under n-SCWP Bernoulli traffic (I-PDBM: continuous line, PDBM: dotted line). Switch sizes were N={2,4}, n={2,8,32,64} (OPS switches operate in the core network, with a high aggregate bandwidth but few ports). Buffer sizes were adequate for OB architectures (packet loss probability below 10^{-9} under 90% load): $M = \{35,10,3,2\}$ for $n = \{2,8,32,64\}$, respectively. To illustrate the effect of traffic burstiness, figures 6(c) and 6(d) show the average delay of I-PDBM and PDBM under an n-SCWP arrival

Markov-modulated ON-OFF Poisson process (MMPP), for burst lengths of $\beta = 16$ (Figure 6(c)) and $\beta = 64$ (Figure 6(d)). Switch sizes were $N = 4$, $n = \{2,8,32,64\}$, and buffer sizes were the same as above. For ON/OFF input traffic, the average delay of both algorithms is very similar. Bursty traffic affects I-PDBM performance as in the case of PDBM and OB architectures [11]. However, for Bernoulli traffic, the average delay of I-PDBM is lower in all configurations. We conclude that packet delay decreases by avoiding PDBM impatience and thus I-PDBM outperforms PDBM.

Table 1 shows buffer requirements for a packet loss probability of 10^{-7} under Bernoulli traffic (simulations with 10^{9} packets). This is a good feasibility metric for OPS nodes, because FDL length is a serious bottleneck nowadays. As we would expect, reducing packet delay leads to lower buffer requirements. We observe that I-PDBM buffer length is very small, and it is close to the ideal OBS case.

Tables 2 and 3 compare the theoretical convergence bound with the number of iterations K to converge with a probability above $1\text{-}10^{-6}$ (90% input load). PDBM and I-PDBM behave similarly. Under Bernoulli traffic, they only need extra iterations for few wavelengths (n=2). However, in all cases the number of iterations is quite low.

5 Conclusions

In this paper we have proposed the enhanced I-PDBM parallel iterative matching scheduler for IBWR optical packet switches [6], which is significantly advantageous over PDBM [9] in terms of performance and hardware complexity.

Acknowledgements

This research has been supported by project grants TEC 2004-05622-C04-01/TCM CAPITAL/TEC 2004-05622-C04-02/TCM ARPAq (MEC, Spain) and PGIDIT04TIC322003PR/PGIDIT05XIC32201PN (Xunta de Galicia, Spain). The authors participate in the COST 291 action and the e-Photon/ONe+ network of excellence.

Switch size	ρ=0.1	ρ=0.2	ρ=0.3	ρ=0.4	ρ=0.5	ρ=0.6	ρ=0.7	ρ=0.8	ρ=0.9
N=2,n=2	2/4/2	3/4/3	3/4/3	4/5/4	5/6/5	5/7/6	7/8/8	10/11/10	18/20/20
N=2,n=8	1/1/1	2/3/2	2/3/2	2/4/2	2/4/2	2/5/2	3/6/3	3/7/4	6/9/8
N=2,n=32	1/1/1	1/1/1	1/1/1	1/1/1	2/3/2	2/3/2	2/4/2	2/4/2	2/5/3
N=2,n=64	1/1/1	1/1/1	1/1/1	1/1/1	1/1/1	1/1/1	2/3/2	2/3/2	2/4/2
N=4,n=2	3/5/3	3/5/4	4/6/4	5/7/5	6/8/6	7/10/8	9/13/11	14/19/16	26/30/30
N=4,n=8	1/1/1	2/3/2	2/3/2	2/3/2	2/4/2	3/4/3	3/5/3	4/8/5	8/13/10
N=4,n=32	1/1/1	1/1/1	1/1/1	1/1/1	2/3/2	2/3/2	2/4/2	2/4/2	3/5/3
N=4,n=64	1/1/1	1/1/1	1/1/1	1/1/1	1/1/1	2/3/2	2/4/2	2/4/2	2/5/2

Table 1. Buffer requirements (OB/PDBM/I-PDBM). Bernoulli input traffic, 10^{-7} packet loss probability.

Bernoulli ρ=0.9	*n*=2	*n*=8	*n*=32	*n*=64
N=2	1/ 3 **4**	2/ 3 **9**	2/ 2 **5**	2/ 2 **4**
N=4	2/ 5 **8**	3/ 3 **13**	2/ 2 **5**	2/ 2 **5**

Table 2. Practical number of iterations for PDBM/I-PDBM convergence vs. theoretical convergence bound (bold), Bernoulli traffic.

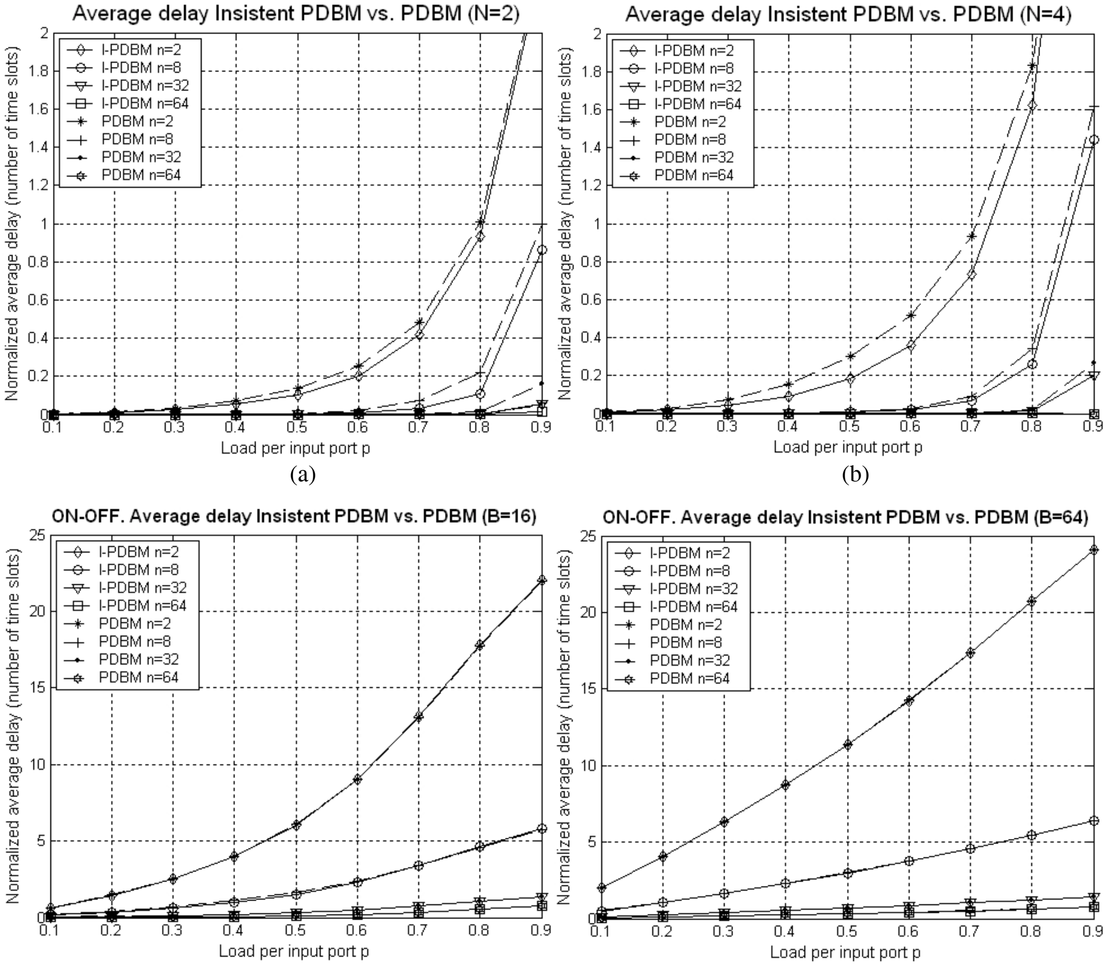
MMPP ρ=0.9, *N*=4	*n*=2	*n*=8	*n*=32	*n*=64
β=16	5/ 5 **8**	6/ 6 **10**	3/ 3 **3**	2/ 2 **2**
β=64	4/5 **8**	6/ 6 **10**	3/ 3 **3**	2/ 2 **2**

Table 3. Practical number of iterations for PDBM/I-PDBM convergence vs. theoretical convergence bound (bold), MMPP traffic.

References

1. Hunter D. et al. "WASPNET: A Wavelength Switched Packet Network". *IEEE Communications Magazine* 1999; 37(3):120-129.
2. Dittman L. et al. "The European IST Project DAVID: A Viable Approach Toward Optical Packet Switching". *IEEE Journal of Selected Areas in Communications* 2003; 21(7): 1026-1040.
3. Pavon-Mariño P., García-Haro J., Malgosa-Sanahuja J., Cerdán F. "Scattered Versus Shared Wavelength Path Operation, Application to Output Buffered Optical Packet Switches. A Comparative Study". *SPIE/Kluwer Optical Networks Magazine* 2003; 4(6):134-145.
4. Pavon-Mariño P., García-Haro J., Malgosa-Sanahuja J., Cerdán F. "Maximal Matching Characterization of Optical Packet Input-Buffered Wavelength Routed Switches". In *Proc. of 2003 IEEE Workshop on High Performance Switching and Routing (HPSR 2003)*, Torino, Italy, June 2003, pp. 55-60.
5. Pavon-Mariño P., Gonzalez-Castaño F.J., Garcia-Haro J. "Round-Robin wavelength assignment: A new packet sequence criterion in Optical Packet Switching SCWP networks". *European Transactions on Telecommunications* 2006; 17(4): 451-459.
6. Zhong W.D., Tucker R. S. "Wavelength routing-based photonic packet buffers and their applications in photonic packet switching systems". *IEEE J. Lightwave Technol.* 1998; 16(10): 1737-1745.
7. Guillemot C et al. "Transparent optical packet switching: the European ACTS KEOPS project approach". *IEEE J. Lightwave Technol.* 1998; 16(12): 2117–2134.
8. Chia M.C. et al. "Packet loss and delay performance of feedback and feed-forward arrayed-waveguide gratings-based optical packet switches with WDM inputs-outputs". *IEEE J. Lightwave Technol.* 2001; 19(9): 1241-1254.
9. Pavón-Mariño, P. *Contribution to Optical Packet Switching: Architectures, Performance Evaluation and Comparative Analyses* (in Spanish). PhD Thesis. Dep. de Tecnologías de la Información y las Comunicaciones, Univ. Politécnica de Cartagena, Spain.
10. Jiang Y., Hamdi M. "A Fully Desynchronized Round-Robin Matching Scheduler for a VOQ Packet Switch Architecture". In *Proc. 2001 Workshop on High Performance Switching and Routing*, May 2001, Dallas, USA, pp. 407-411.

11. Pavon-Mariño P., García-Haro J., Jajszczyk A. "Parallel Desynchronized Block Matching: A Feasible Scheduling Algorithm for the Input-Buffered Wavelength-Routed Switch". Submitted to *Computer Networks* for publication.



(a)

(b)

(c)

(d)

Fig. 6. (a) and (b) Average delay under SCWP Bernoulli traffic; (c) and (d) Average delay under SCWP MMPP traffic; (c) β=16; (d) β=64.