ファイバ遅延線バッファを備えたフォトニックパケットスイッチにおけるパケットスケジューリングアルゴリズム

山口 貴詩† 馬場 健一†† 村田 正幸‡ 北山 研一‡‡

† 大阪大学 大学院基礎工学研究科
〒 560–8531 豊中市待兼山町 1–3
Phone: 06–6850–6863, Fax: 06–6850–6868
E-mail: t-yamagt@ics.es.osaka-u.ac.jp

‡ 大阪大学 サイバーメディアセンター
〒 560–8531 豊中市待兼山町 1–3
Phone: 06–6850–6867, Fax: 06–6850–6868
E-mail: murata@cmc.osaka-u.ac.jp

†† 大阪大学 サイバーメディアセンター
〒 567–0047 茨木市美穂ヶ丘 5–1
Phone: 06–6879–8793, Fax: 06–6879–8794
E-mail: baba@cmc.osaka-u.ac.jp

‡‡ 大阪大学 大学院工学研究科
〒 567–0871 吹田市山田丘 2–1
Phone: 06–6879–7728, Fax: 06–6879–7688
E-mail: kitayama@comm.eng.osaka-u.ac.jp

あらまし 本論文では、フォトニックパケットスイッチとして、ファイバ遅延線バッファを備え共有バッファとして利用するアーキテクチャを対象とし、提案するパケットスケジューリングアルゴリズムを適用した場合の性能を明らかにしている。共有バッファ型スイッチでは負荷が高い場合にその性能が大きく低下することから、その問題を解決するために、パケット間空き領域低減手法を提案した。その結果、高負荷時においても安定した性能を示すことを明らかにした。次に、パケットスケジューリングアルゴリズムのハードウェアでの実現性を考慮した上で、その動作シミュレーションを行うことにより、処理遅延時間の観点からその評価を行った。その結果、スケジューリングの際に扱う波長数が処理遅延時間に大きな影響を与えることを明らかにした。

キーワード WDM，フォトニックパケットスイッチ，FDL バッファ，パケットスケジューリングアルゴリズム，ハードウェア実現性

On Packet Scheduling Algorithm for WDM-based Photonic Packet Switch with Fiber Delay Line Buffers

Takashi YAMAGUCHI†, Ken-ichi BABA††,
Masayuki MURATA‡, Ken-ichi KITAYAMA‡‡

†Graduate School of Engineering Science,
Osaka University
1–3 Machikaneyama, Toyonaka, Osaka 560–8531, Japan
Phone: +81–6–6850–6863, Fax: +81–6–6850–6868
E-mail: t-yamagt@ics.es.osaka-u.ac.jp

‡Cybermedia Center, Osaka University
1–3 Machikaneyama, Toyonaka, Osaka 560–8531, Japan
Phone: +81–6–6850–6867, Fax: +81–6–6850–6868
E-mail: murata@cmc.osaka-u.ac.jp

††Cybermedia Center, Osaka University
5–1 Mihogaoka, Ibaraki, Osaka 567–0047, Japan
Phone: +81–6–6879–8793, Fax: +81–6–6879–8794
E-mail: baba@cmc.osaka-u.ac.jp

‡‡Graduate School of Engineering, Osaka University
2–1 Yamada-Oka, Suita, Osaka 565–0871, Japan
Phone: +81–6–6879–7728, Fax: +81–6–6879–7688
E-mail: kitayama@comm.eng.osaka-u.ac.jp

Abstract In this paper, we evaluate the performance of the photonic packet switch architecture to which packet scheduling algorithms is applied with fiber delay line buffers which function as the shared buffer. Since, in a shared buffer type switch, a void space introduces unacceptable performance degradation when high traffic load conditions, we propose a void space reduction method to resolve the problem. Our simulation results show that our proposed method achieves the stable performance in the shared buffer type switch under high traffic load conditions. Next, we consider the feasible design of the architecture for the algorithms with a PLD design software, and we discuss the feasibility of the algorithms from the viewpoint of the processing delay time. Through the simulation experiments, we found that the number of wavelengths in the switch greatly influenced the processing delay time.

Keywords WDM, Photonic Packet Switch, FDL Buffer, Packet Scheduling Algorithm, Hardware Feasibility

1 Introduction

The progress of optical transmission technology in recent years has been remarkable especially in achieving a Tbps class of transmission speed. However, as the bandwidth is increasing sharply because of advances in optical transmission technology, the electronic technology for switching systems is approaching its limit. Thus, we need a photonic network which can incorporate functions such as the multiplexing, demultiplexing, switching, and routing functions in an optical domain, through which electronic control can be minimized. Then, we can expect to see a super–high speed network that exceeds the speed limit of the electronics devices.

In this paper, we study packet scheduling algorithms for the photonic packet switch. In the packet switch, packet loss is caused by the contention of more than two packets destined for the same output port. In the conventional electronic switch, the output times of those packets are shifted by a store-and-forward technique utilizing RAM (Random Access Memory), and resolving packet contention is a simple procedure. However, in the photonic packet switch, we need to take other approaches because RAM in an optical domain is still not available. For instance, optical buffering is achieved by using optical fiber delay lines (FDL) for packet contention resolution [1, 2, 3, 4]. Using FDL, packets are stored in different lengths of delay lines, through which the departing times of packets are time-shifted. Another technique used for resolving packet contention is to introduce wavelength conversion on FDL, where the wavelengths of more than two packets contending the same output port are converted to different wavelengths by using tunable wavelength converters. Although wavelength conversion requires a higher hardware cost, it results in a better performance [5, 6]. However, once the packet is injected into the FDL, it cannot be sent to the output port for the time duration corresponding to the length of FDL. Thus, we need an effective packet scheduling algorithm for WDM-based FDL (or WDM-FDL in short), and this is the main subject of this paper.

In this paper, we evaluate the performance of photonic packet switch with WDM-FDL supporting variable–length packets. We assume that all arriving packets are synchronized at the predefined time slot, and packet length is given by an integer multiple of the time slot. Note that time–synchronization of asynchronously arriving packets can be realized by the technique presented in [7]. In this paper, we consider a shared buffer type switch, where all the packets failing to acquire the output port are sent to the single FDL buffer within the switch. In addition, we also studied an output buffer type switch, which stores packets in the WDM-FDL buffer attached to each output port, and found the advantage of the shared buffer type switch from the viewpoint of hardware cost through the comparative evaluation between the shared buffer type switch and the output buffer type switch in [8]. As described above, the use of a packet scheduling algorithm is important for enabling the photonic packet switch to achieve a high performance. We apply packet scheduling algorithms proposed in [9, 10] to the above packet switching architecture and evaluate the performance of the switch. Also, we propose a new packet scheduling algorithm called the *void space reduction method*. Furthermore, we consider the feasible design of the architecture for the algorithms with a PLD design software, and we discuss the feasibility of the algorithms from the viewpoint of the processing delay time.

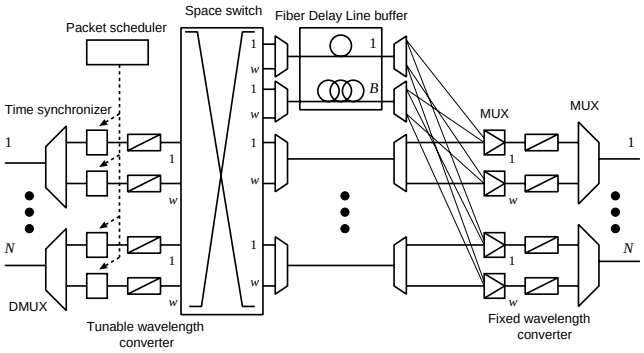

Figure 1: Shared buffer type photonic packet switch architecture

The rest of the paper is organized as follows. In Section 2, we briefly present shared buffer type architecture for photonic packet switch supporting variable length packets. In Section 3, we describe packet scheduling algorithms that determine the wavelength of packets inserted in the FDL buffer, and then present the void space reduction method. In Section 4, we introduce the simulation model and evaluate the architecture. In Section 5, we discuss the feasibility of the packet scheduling algorithms. Conclusions and future work are summarized in Section 6.

2 Photonic Packet Switch Architectures

The photonic packet switch that we consider in this paper accepts variable–length packets arriving asynchronously at the input port. Arriving packets are synchronized at a time with a predefined size. A synchronization mechanism for asynchronously arriving packets is presented in [7].

The packet length is an integer multiple of the time slot size. When we utilize FDL, the time slot size affects the performance of the switch when the variable–length packets are treated. For example, in [11], it is shown that the best performance is obtained when the time slot size is set to about 30 percent of the average packet size. We will also use this value in the simulation experiments presented in Section 4 and Section 5.

The photonic packet switch is equipped with wavelength converters and optical buffers in order to resolve contentions of packets. A number W of the wavelengths are multiplexed on the fiber and the packets are carried on the wavelength. The wavelengths are demultiplexed at the input port of the switch. The packet on the wavelength is then time-synchronized at the time slot. Then, the packet scheduling algorithm determines the destination of each arriving packet. If the corresponding output port is available, the packet is sent to the output port directly after being assigned the appropriate wavelength. Otherwise, it is inserted in the optical buffer according to the scheduling algorithm. The scheduled packets are sent through a space switch. The wavelength of the packet is converted to the proper wavelength by a fixed wavelength converter at the output port.

One FDL buffer consists of a number B of delay lines, which are set up in parallel. The length of n-th delay line is n in time slot size. As we will describe later, the number of wavelengths on FDL (denoted by W_i) is equal to or larger than the number of wavelengths on the input and output fibers, W. In the following, we call the number of delay lines in one FDL buffer a *buffer depth* (denoted by B), and the number of

delay lines in the whole switch a *buffer size* (denoted by B_T). The *virtual buffer size* is denoted by $B_T \times W_i$.

Figure 1 shows the architecture of the shared buffer type switch, which has one shared FDL buffer, and the packets are stored at the same buffer regardless of the destination output port. When the contention cannot be resolved by wavelength conversion, the packets are sent to the FDL buffer. When the contention of packets can be resolved by wavelength conversion, on the other hand, the packets are sent to the output ports directly. The shared buffer type switch has only one FDL buffer with W virtual input lines. The buffer size B_T is equal to B.

3 Packet Scheduling Algorithms

A packet scheduling algorithm is needed in order to determine the wavelength and FDL for the arriving packets. In this section, we consider that time is synchronized and multiple packets may arrive at the time slot boundary because the packets one synchronized by an appropriate method as described in the previous section. For each of the packets arriving within the time slot, the packet scheduler finds the appropriate wavelength and delay line as follows. If an unused wavelength on the output port is found, the packet is sent to the output port directly. When no wavelength is available at the output port, the appropriate FDL is found.

3.1 Buffer Control Algorithms

Algorithm A0: Assign the Wavelength in Round-Robin Fashion

One of simplest forms of the algorithm is to assign the wavelength for packets arriving within the time slot in a round-robin fashion. This is simple and easy to implement. The information that the algorithm should hold includes (1) the latest number of the wavelength to which the previous packet is assigned, and (2) the queue lengths of the wavelengths. The latter can be implemented by using a counter associated with the wavelength, which is increased incrementally by the packet length (in time slot) when the wavelength is chosen by the algorithm and decreased decrementally by one at every time slot.

Algorithm A1: Assign to the Buffer with Minimum Queue [9, 12]

Algorithm A1 assigns the packet to the wavelength with the minimum queue length. The order selection of the packet from among the ones arriving within the time slot is random, or is simply decided according to the input port number at which the packet has arrived. For this purpose, a simple counter associated with the wavelength is utilized, as in Algorithm A0. Then, the appropriate FDL is selected for the packet to be sent to. If the FDL buffer is full, the packet is discarded. This algorithm is simple and packet scheduling is easy to implement because the procedure used by the scheduler only seeks the minimum queue length for each packet.

3.2 Void Space Reduction Method

In order to prevent errors in the ordering of packets, the switch processes packets in order of arrival. Thus, when the packet is sent to FDL, a newly arriving packet with same input/output ports as the previously arriving packet should not be sent to the shorter FDL. The Algorithm A1 have this feature. However, this feature causes the unacceptable performance degradations as we will demonstrate in the next section.

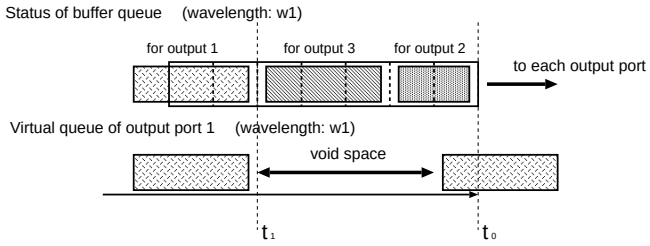

Figure 2: Void space in shared buffer type switch

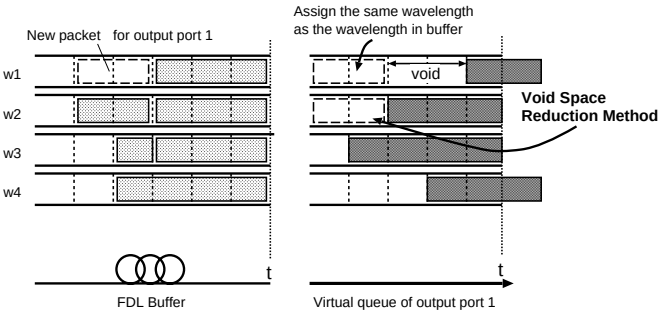

Figure 3: Void space reduction method

Since the shared buffer type switch has a single buffer, the queue length of the buffer becomes in long a high traffic load condition. Consequently the output interval between two packets destined for the same output port becomes large, and this is called the *void space* in this paper. As an example, Figure 2 illustrates why and how the void space appears. At output port 1, a packet is being sent on wavelength w_1. The queue counter is then increased by the packets sent to output ports 2 and 3. Now, a new packet destined for output port 1 arrives at the switch. If the packet is assigned wavelength w_1, the packet will be stored at the back of the queue of the buffer because wavelength w_1 of output port 1 is in use. Then, a void space of length 4 appears, leading to low utilization of output port 1. In this case, it is impossible to use output port 1 until all the buffered packets are transmitted, regardless of whether the port is actually in use or not.

The void space and the excess load are considered in [13] and [14], respectively. However, the void space used in this paper is different from those. The void space and the excess load are the actual empty spaces between the asynchronously arriving packets, which are destined for the same output port. On the other hand, the void space in this paper is the portion of fiber that the packets for the different output ports are stored between two packets for the same output port. Then it causes the low utilization of output ports. In [13], a void filling algorithm has been proposed. However, it needs to maintain the arriving/departing times of all packets stored in the buffer in order to insert a new packet within the void space. Therefore, the algorithm complexity is very high and is difficult to implement.

Our proposal, called the *void space reduction method*, reduces the ill–effect of the void space by using wavelength conversion. The wavelength of the packet is converted such that the influence of the void space is minimized. Figure 3 illustrates our approach. Suppose that a new packet destined for output port 1 arrives at the switch. The packet is as-

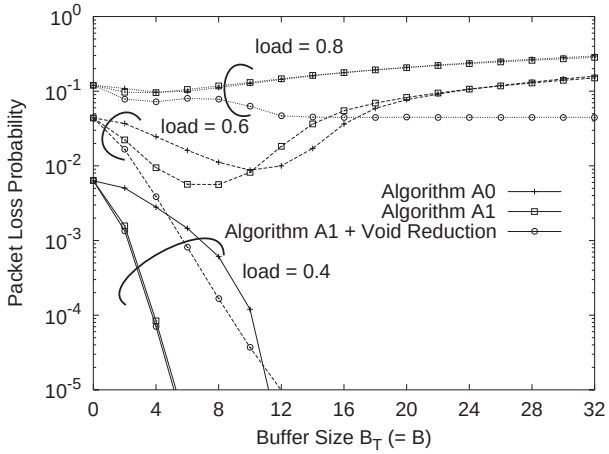

Figure 4: Packet loss probability (packet scheduling algorithms, load = 0.4, 0.6, 0.8)

signed wavelength w_1 and is stored in the buffer. If the next arriving packet is assigned wavelength w_1, a void space between two time slots appears. On the other hand, our method compares the queue lengths of the wavelength buffers and selects a wavelength which will minimize the void space. In the above case, the new packet is assigned wavelength w_2, and thus we can avoid void space completely. Note that this method can only be applied to Algorithms A1.

More specifically, our method works as follows. We introduce a *virtual queue* within the physical shared buffer. A virtual queue is a logical queue maintained for each of the combinations of the output port and wavelength on the output fiber. There are a number $N \times W$ of virtual queues in the shared buffer. We also introduce a counter to maintain the output time of the last packet in the virtual queue. When a new packet arrives and is decided to be stored in the buffer due to no available wavelength, the scheduler finds the smallest difference between the physical queue length of the wavelength and the virtual queue counter. Then, the packet is inserted into FDL. After the packet goes through the FDL, the wavelength of the packet is tuned to the wavelength that is actually used on the output fiber.

Lastly, it should be noted that in order to implement this method, wavelength conversion is necessary, which may lead to a higher switch cost, but the improvement in performance is remarkable, as we will demonstrate in the next section.

4 Performance of the Photonic Packet Switches

4.1 Simulation Model

For evaluation, the photonic packet switch and arriving traffic are modeled as follows. The numbers of input/output ports N and wavelengths on the fiber W are set to be 16 and 8, respectively. The wavelength capacity is 40 Gbps. A packet arrives according to a Poisson process. The average packet length is 400Bytes. The packet length is exponentially distributed, but truncated at 1000Bytes. The time slot size is 20ns, which corresponds to 30% of the average packet length [11]. Every input fiber and wavelength has the same packet arrival rate, and the destination output port of the packet is chosen randomly.

4.2 Evaluation of the Packet Scheduling Algorithms

In this subsection, we evaluate the packet scheduling algorithms A0, A1 and the void space reduction method described in Section 3. Figure 4 shows the simulation results of packet loss probability dependent on the buffer size B_T in the shared buffer type switch. In this figure, the performance of the switch decreases when the switch is equipped with a larger buffer size. This is because the queue length becomes long and the possibility of a void space appearing becomes high, as was described in Section 3.2. It can be observed that the performance is dramatically improved by introducing the void space reduction method.

5 Feasibility of the Packet Scheduling Algorithms

In the simulation results, we have observed that our proposed method can achieve the drastic improvement of the performance. However, we have another issue which should be discussed. It is the feasibility whether these algorithms can be actually executable, or how much progress of hardware devices is required in order to implement those algorithms. In this section, we discuss it from the viewpoint of the processing delay time by the algorithms.

5.1 PLD Implementation and Simulation

In order to consider the feasibility of the packet scheduling algorithms, we design the hardware architecture for the algorithm by using the PLD (Programmable Logic Device) design tool. We describe the source code in VHDL (VHSIC Hardware Description Language), and we measure the processing delay time of the algorithms through the simulations.

Now, we explain the detail of the PLD implementation and simulation. The PLD design tool that we used is MAX+PLUS®II software by Altera Corporation [15]. First, we consider the scheduling phase of the algorithm by dividing it into three parts separately. The reason why we introduce the scheduling phase is that we want to simplify the implementation of the algorithms.

We define three scheduling phases as follows.

- Phase P1: Zero Decision of the Queue Length, which examines whether free wavelength exists in the output port. If the queue of which length is zero exists, the packet is sent to the destination output port directly. Otherwise, it is inserted to the optical buffer. This scheduling phase is necessary in algorithms A0 and A1.
- Phase P2: Comparison of Buffer Queue and Virtual Queue, which compares the length of the buffer queue with that of the virtual queue in each wavelength. This is necessary in algorithm A0 and A1.
- Phase P3: Search for the Shortest Queue, which searches the queues in the buffer or the destination output port for the shortest one. This is necessary in algorithm A1 and the void space reduction method.

In the packet scheduling algorithm, the other processes including the increment of counters are required. However, we ignore such processing time since the processing delay time of them are much smaller than those of the above three scheduling phases.

We now take a close look at the simulation results. Figure 5 shows processing delay time dependent on the number of wavelengths in the fiber in each scheduling phase. As shown in this figure, the processing delay time in phase P2 is fixed regardless of the number of wavelengths. It is because the comparison of two queue lengths can be processed simultaneously. On the other hand, those in phase P1 and P3

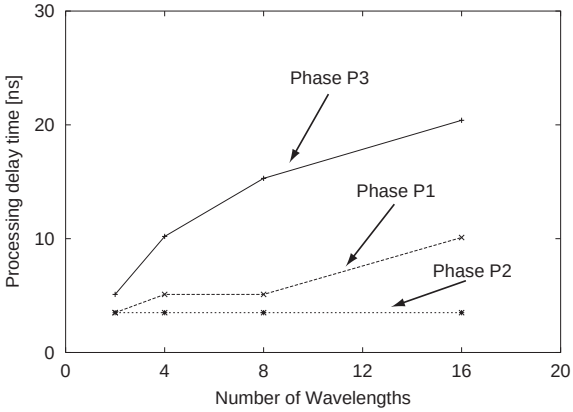

Figure 5: Processing delay times of scheduling phases

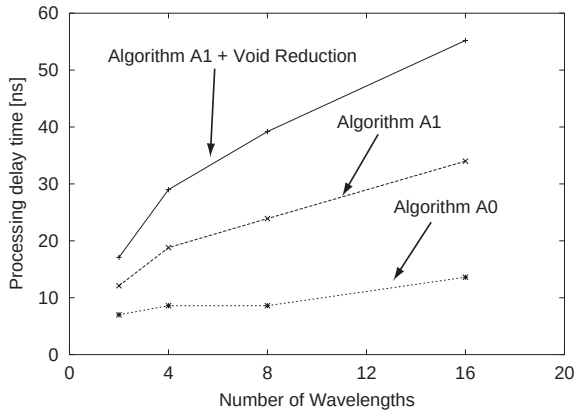

Figure 6: Processing delay times of packet scheduling algorithms

increase as the number of wavelengths increases, because the procedures of phase P1 and phase P3 are sequential depending on the number of wavelengths. In Figure 6, we derive the processing delay time for packet scheduling algorithms based on the results in Fig. 5. In this figure, the processing time of scheduling phase P3 is a burden for the packet scheduling algorithms. For example, when the number of wavelengths is 8, the processing delay time of algorithm A0, A1 and A1 with void space reduction method is 8.6ns, 23.9ns and 39.2ns, respectively. Therefore, we can observe it may be concluded that the algorithms which include the procedure of phase P3 needs much more processing delay time for their execution. However, we can reduce the processing delay time for the packet scheduling algorithm with the reduction of the relevant wavelengths as described in the next subsection.

5.2 Wavelengths Grouping Method

If the too complicated packet scheduling algorithm, requiring slow electronic is applied to a switch, it is difficult that the controller processes all the arriving packets in one slot time. Therefore, it is important to introduce the packet scheduling algorithm posing a reasonable processing delay time within a time slot. One effective way to reducing the processing delay time is to decrease the number of the wavelengths that the scheduler have to examine, because the processing delay time much depends on the number of wavelengths as discussed in the previous section. To realize it, we propose the *wavelengths grouping method*, which divides the wavelengths on the fiber into several groups and determines the group depending on the combination of its input port and output port

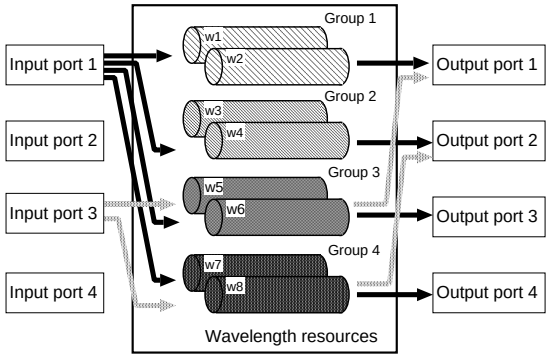

Figure 7: An example of wavelengths grouping method (4 $\times$ 4 switch, 8 wavelengths, 4 groups)

Table 1: Wavelength groups (4 $\times$ 4 switch, 8 wavelengths, 4 groups)

input \ output	1	2	3	4
1	w_1, w_2	w_3, w_4	w_5, w_6	w_7, w_8
2	w_7, w_8	w_1, w_2	w_3, w_4	w_5, w_6
3	w_5, w_6	w_7, w_8	w_1, w_2	w_3, w_4
4	w_3, w_4	w_5, w_6	w_7, w_8	w_1, w_2

for each packet. The wavelength in the selected group is assigned to the packet. We illustrate an example in Figure 7 and Table 1. When N numbers of input/output ports and W wavelengths on the fiber are four and eight (from w_1 to w_8), respectively, and we set four groups of wavelengths, the packet that arrives at the input port 1 and is destined for output port 1 will be assigned wavelength w_1 or wavelength w_2, and the packet that arrives at the input port 2 and is destined for output port 3 will be assigned wavelength w_3 or w_4. In this way, the processing time for wavelength search can be reduced. Of course, the range of wavelength selection becomes narrow. Figures 8 and 9 show the simulation results of packet loss probability in the switch applied the wavelengths grouping method. As shown in these figures, as the number of groups increases, the performance of a switch is degraded. For example, when the number of groups is more than 4, even the void space reduction method can not achieve the superior performance. Therefore, it is important to consider the trade-off between the performance of the switch and the processing delay time of the packet scheduling algorithm.

5.3 Consideration of the feasibility

The simulation results for the implementation have shown that it is difficult to apply the packet scheduling algorithms to the switch with the present technology of electronic devices. However, the progress of the technology for semiconductor is remarkable, and that does not deny the feasibility in the near future.

For example, Table 2 shows the technology road map for ASIC (Application-Specific Integrated Circuit) [16]. The road map predicts that the chip frequency improves twice or more for 8 years. If the prediction turns out to be correct, the processing delay time of algorithm A1 with void space reduction method will be decreased to less than 20ns about 10 years after. It encourages us to expect the realization of the packet scheduling algorithms.

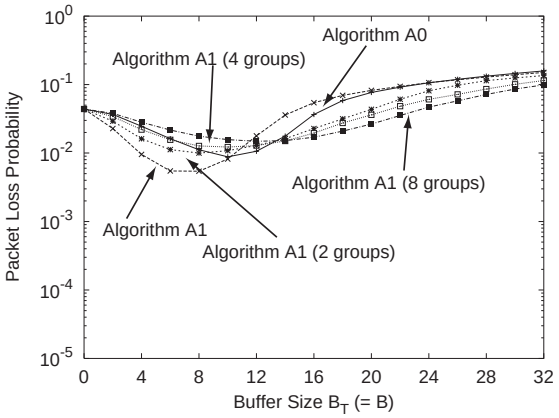

Figure 8: Packet loss probability (wavelengths grouping method, algorithm A1 without void space reduction method, load = 0.6)

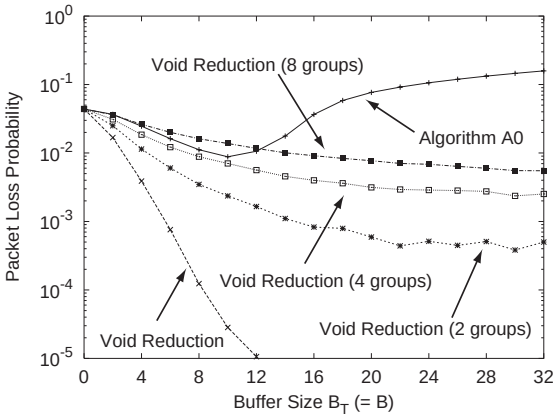

Figure 9: Packet loss probability (wavelengths grouping method, algorithm A1 with void space reduction method, load = 0.6)

Table 2: Technology road map for performance of ASIC [16]

Year	2000	2002	2004	2008	2014
Frequency (MHz)	559	700	828	1,200	1,800

6 Conclusion

In this paper, we have evaluated the performance of the photonic packet switch architecture to which packet scheduling algorithms is applied with fiber delay line buffers which function as the shared buffer. We proposed the void space reduction method, and the method could improve the performance of the shared buffer type switch. Furthermore, we discussed the feasibility of the packet scheduling algorithms from the viewpoint of the processing delay time. And, we considered the feasibility with a hardware implementation and a technology road map.

In future work, we need to consider the more effective packet scheduling algorithm in the viewpoint of reduction of the scheduling time and improvement of the performance of the switch.

Acknowledgments

This work was supported in part by TAO (Telecommunications Advancement Organization of Japan) under the project "Research and Development Project of Photonic Router for High Speed, High Quality, and Highly Efficient Internet ".

References

[1] S. L. Danielsen, B. Mikkelsen, C. Joergesen, T. Durhuus, and K. E. Stubkjaer, "WDM packet switch architectures and analysis of the influence of tuneable wavelength converters on the performance," *IEEE Journal of Lightwave Technology*, vol. 15, pp. 219–227, Feb. 1997.

[2] S. L. Danielsen, C. Joergesen, B. Mikkelsen, and K. E. Stubkjaer, "Analysis of a WDM packet switch with improved performance under bursty traffic conditions due to tuneable wavelength converters," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 729–735, May 1998.

[3] D. Hunter, M. C. Chia, and I. Andonovic, "Buffering in optical packet switches," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 2081–2094, Dec. 1998.

[4] K. L. Hall and K. A. Rauschenbach, "All-optical buffering of 40-Gb/s data packets," *IEEE Photonic Technology Letters*, vol. 10, pp. 442–444, Mar. 1998.

[5] S. Yao, B. Mukherjee, S. J. B. Yoo, and S. Dixit, "All-optical packet-swtiched networks: A study of contention-resolution schemes in an irregular mesh network with variable-sized packets," in *Proceedings of OptiComm 2000*, pp. 235–246, Nov. 2000.

[6] S. L. Danielsen, B. Mikkelsen, C. Joergesen, T. Durhuus, and K. E. Stubkjaer, "Wavelength conversion in optical packet switching," *IEEE Journal of Lightwave Technology*, vol. 16, pp. 2095–2108, Feb. 1997.

[7] M. Murata and K. Kitayama, "Ultrafast photonic label switch for asynchronous packets of variable length," in *Proceedings of INFOCOM 2002*, vol. 1, pp. 371–380, June 2002.

[8] T. Yamaguchi, K. Baba, M. Murata, and K. Kitayama, "Packet scheduling for WDM fiber delay line buffers in photonic packet switches," in *Proceedings of OptiComm 2002*, pp. 262–273, July 2002.

[9] A. Ge, L. tancevski, G. Callegati, and L. Tamil, "WDM fiber delay line buffer control for optical packet switching," in *Proceedings of OptiComm 2000*, pp. 247–257, Nov. 2000.

[10] F. Callegati, W. Cerroni, and G. Corazza, "Optimization of wavelength allocation in WDM optical buffers," *Optical Networks Magazine*, vol. 39, pp. 66–72, Nov. 2001.

[11] F. Callegati, "Optical buffers for variable length packets," *IEEE Communications Letters*, vol. 4, pp. 292–294, Sept. 2000.

[12] L. S. Tamil, F. Masetti, T. McDermott, G. Castanon, A. Ge, and L. Tancevski, "Optical IP routers: Design and performance issues under self-similar traffic," *Journal of High-Speed Networks*, vol. 8, pp. 59–67, June 1999.

[13] L. Tancevski, S. Yegnanarayanan, G. Castanon, L. Tamil, F. Masetti, and T. McDermott, "Optical routing of asynchronous, variable length packets," *IEEE Journal on Selected Areas in Communications*, vol. 18, pp. 2084–2093, Oct. 2000.

[14] P. B. Hansen, S. L. Danielsen, and K. E. Stubkjaer, "Optical packet switching without packet alignment," in *Proceedings of ECOC '98*, pp. 591–592, Sept. 1998.

[15] Altera Corporation, "Max+plus II software." available at `http://www.altera.com/products/software/pld/products/maxplus2/`.

[16] *International Technology Roadmap for Semiconductors 1999 Edition*. Semiconductor Industry Association, 1999.