Scheduling Optical Packets in Wavelength, Time, and Space Domains for All-Optical Packet Switching Routers

Haijun Yang, Venkatesh Akella, Chen-Nee Chuah, S. J. Ben Yoo
Department of Electrical and Computer Engineering, University of California, Davis
Davis, California 95616, U.S.A.
{haiyang,akella,chuah,yoo}@ece.ucdavis.edu

***Abstract*—This paper describes a novel sequential wavelength-time-space (sWTS) scheduling algorithm to solve the arbitration problem in an all-optical packet switch router and shows the simulation as well as the hardware implementation results. The simulation results with self-similar traffic input demonstrate that the sWTS arbitration algorithm effectively improves the packet blocking rate, forwarding latency and jitter as the number of wavelength channels per port and the number of recirculation buffer ports in the switch fabric increase. The proposed algorithm further facilitates a very fast and hardware-efficient mixed-tree output port arbiter design. The hardware implementation is evaluated in terms of area and delay for various switch sizes with both the Xilinx XCV1000E FPGA and a 0.25-micron commercial ASIC library.**

***Keywords*—Switch fabric arbitration, contention resolution, optical packet switching.**

I. INTRODUCTION

While multi-wavelength optical networking technology has provided transmission capacity approaching 10 Tb/sec on a single-mode fiber carrying more than 400 wavelengths, future switching systems may require total switching capacity exceeding 1 Petabit/sec with 64000–by-64000 non-blocking connectivity. With the phenomenal growth in the Internet traffic, it is also desired to support such large-scale switching capacity and connectivity at the packet level, thus offering greater flexibility and granularity towards seamless integration of data and optical networking. However, realizing such a high-capacity switching system with conventional electronic routers would entail severe difficulties owing to the excessive power and space requirements. All-optical packet switching (OPS) [1] is an attractive approach in this perspective. The OPS router allows the routing and switching of packets in the optical domain directly to provide the scalability compatible with the optical transmission capability. The performance and scalability of OPS routers are directly linked to the optical switch fabric architectures. The arbitration scheme is a service discipline that decides the service order among the input packets. It sends the necessary information to the proper output port arbiter and the output port arbiter picks some packets among all the input packets destined for this port. Thus, both the design of an efficient and fair arbitration scheme and the implementation of a fast and scalable output port arbiter are important for accelerating the packet switching process of the OPS routers. Most of the previous research work on the switch fabric arbitration assumes the input Virtual Output Queueing (VOQ) switch model with synchronized, fixed length packet inputs. However, due to lack of mature optical memory technologies, most current OPS routers instead adopt the fiber delay line (FDL) buffer to provide discrete and deterministic delays to resolve packet contentions. Since the FDL buffer can not provide the true queuing like electronic random access memory, the optical packet switching router can not employ the previous arbitration algorithms based on the electronic queuing model. Furthermore, it is difficult to divide the variable length optical packets into fixed length optical packets for synchronized switching in OPS routers without optical-electrical-optical (OEO) conversion. Thus, the new switch models and arbitration algorithms capable of all-optical asynchronous, variable length packet switching must be developed from the beginning to resolve switching contention. In OPS routers, each switch port carries multiple wavelength channels, which enables a new wavelength dimension to resolve the packet contention through wavelength-aware arbitration algorithms without causing extra packet latency and jitter. At the same time, the wavelength dimension also poses new design challenges for the wavelength-aware arbiters.

This paper focuses on the design, simulation, and implementation of a new OPS arbitration scheme to resolve the optical packet switch fabric contention, as the initial research effort towards making a high-capacity, small footprint all-optical data router. The proposed sequential wavelength-time-space (sWTS) arbitration algorithm efficiently combines the contention resolutions in wavelength [2], time [3]-[4], and space [5] domains hierarchically. The network-wide simulation results show the packet blocking rate, packet forwarding latency and jitter can be improved greatly by increasing the numbers of wavelength channels per port and the number of recirculation buffer ports. This paper further presents a scalable and fair mixed-tree port arbiter (MTA) design with various pipelined architectures to facilitate the hardware implementation of the sWTS algorithm. The paper is organized as follows. Section II discusses the related work. Section III gives an overview of the OPS router switch and control architectures and the contention resolutions in wavelength, time, and space domains. Section IV develops the new wavelength-time-space (WTS) switch model and discusses the proposed sWTS arbitration algorithm for OPS routers. Section V studies the algorithm performance issues in terms of packet loss rate, end-to-end delay and jitter, etc. Section VI describes the design of the mixed-tree port arbiter and discusses the results of various pipelined arbiter implementation architectures. Section VII concludes this work.

II. RELATED WORK

Most previous research work on arbitration algorithms is based on input VOQ switch model, which assumes the input queuing buffer [6]. Also, most of them rely on *wavelength-blind* arbitration algorithms because the output port can only grant one request within one arbitration period. Yang *et al* proposed a wavelength-aware multi-server switch model and the wavelength-aware arbitration algorithm kDRR [7]. However the switch model still assumes the electronic input buffer. Gupta *et al* proposed the round-robin based programmable priority encoder (PPE) designs with the O(*log N*) gate delay [8]. Chao *et al* proposed a ping-pong arbiter (PPA) design based on the binary-tree architecture [9]. Lee *et al* proposed a similar design [10]. However, these designs can only grant one request in one arbitration cycle. Zheng *et al* proposed a programmable k-selector (PkS), which may grant at most *k* requests in one arbitration cycle, but it still assumes input VOQ model [11]. The Knockout switch concentrator may grant multiple requests but it assumes the electronic output buffer and fixed-size packets [12].

The key differences between our work and the related work are as follows. First, the all-optical packet switch adopts the shared recirculation fiber delay line buffer instead of the electronic queueing buffer. Second, the proposed arbitration algorithm and hardware design are *wavelength-aware*, which means multiple packets may be granted by the same output port arbiter within one arbitration round. Third, the proposed arbiter supports asynchronous, variable length packet switching while keeping the packets in the optical domain.

III. OPTICAL PACKET SWITCHING ROUTER BACKGROUND

A. General OPS Router Architecture

Optical-label switching (OLS) [13] has been proposed as a promising technology to facilitate practical implementations of OPS by decoupling the data plane from the control plane. Each optical packet is attached to an optical *label* containing the information necessary for forwarding the packet such as the packet destination, packet length, QoS/CoS parameters. Fig. 1 shows the conceptual architecture of an OPS router system. Upon the arrival of an optical packet, its label is extracted and sent to the label processor of the electronic controller for processing by OEO conversion. The wavelength-aware switch fabric arbiter resolves the packet contentions, makes the switching decision, and instructs the optical switching fabric to take action. *Meanwhile, the optical packet travels through the fiber delay line to compensate for the time it takes to electronically process the labels*. Then, it is switched to the desired output port. Some packets are dropped to the local clients through the client interface. Some are added from the local clients. The data packet is kept in the optical domain. In this way the packet forwarding process is independent of the optical payload data format, bit rate, and underlying protocols. The control processor makes the routing computation and updates the forwarding tables. The network control and management system (NC&M) interfaces with the controller to collect the network statistics and management information.

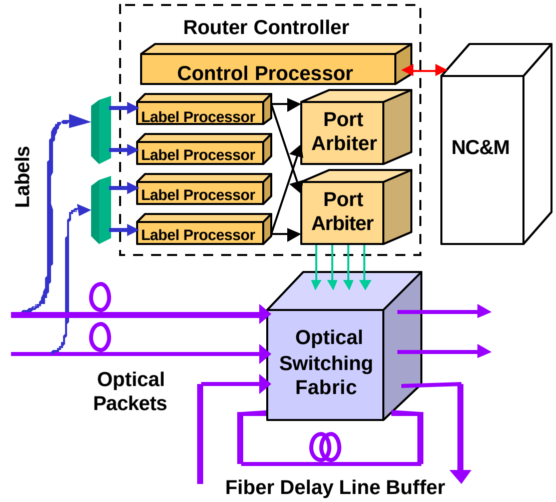


Fig.1. General OPS router architecture.

B. OPS Switch Architecture and Contention Resolutions

The general OPS router switch architecture consists of *K* input fiber ports and *K* output fiber ports as shown in Fig. 2. Each optical fiber port carries *W* wavelength channels, which enables packet contention resolution in wavelength domain. The optical switch fabric connects any input wavelength channel of any input port to any output wavelength of any output port without blocking. The shared optical recirculation buffer takes advantage of the special recirculation ports and the optical fiber delay lines to provide limited buffering capability in the time domain.

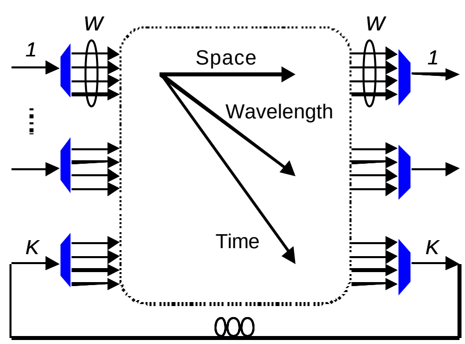


Fig.2. Switch fabric architecture and contention resolutions.

When two or more packets attempt to go to the same output fiber port on the same wavelength at the same time, the OLS router can resolve the contention in the following three methods hierarchically:

- **Wavelength conversion (wavelength domain):** the packet may be converted and switched to any free wavelength channel on the packet's first preferred output fiber port without delay.
- **Time buffering (time domain):** if the above fails, the router will attempt to forward the packet toa recirculation fiber port for buffering and circulate it back to the input side of switch fabric.
- **Deflection routing (space domain):** if the above methods fail, the router will resort to space dimension by sending the contending packet to a second preferred output port and

deflecting it to a neighboring node from which it can be forwarded towards its destination.

IV. ARBITRATION IN WAVELENGTH, TIME, AND SPACE DOMAINS

A. OPS Router Controller Function

Without loss of generality, we assume that the switch fabric has K input ports and K output ports and that each port carries W wavelength channels. The main function of the controller is to make packet-forwarding decisions based on the label contents and to instruct the switch fabric to take action. Fig. 1 shows the OPS router controller block diagram. Each input wavelength channel is assigned to one label processor and each output fiber port is associated with one output port arbiter. Upon arrival of an optical packet, the label processor analyzes the label fields, determines its first and second preferred output ports by looking up the first and second forwarding tables, and sends a forwarding request to its desired output port arbiter. Each output port arbiter may receive at most KW forwarding requests from input label processors and grant at most W of them based on the number of available wavelength channels on the output port.

B. WTS Switch Model

Fig. 3 shows the proposed asymmetric bipartite switch model combining wavelength-time-space domain contention resolutions (WTS model) for OPS routers. The WTS model consists of KW input wavelength channel nodes denoted by I_{11}, … , I_{KW} (($|I_{11}|$ = … $|I_{KW}|$ = 1)and K output fiber nodes denoted by O_1, … , O_K ($|O_1|$ = …$|O_K|$=W). Each input wavelength channel can be modeled as a single-server. Also, each output fiber carries W servers and can be modeled as a multi-server. Thus, the proposed WTS switch model consists of KW input channel single-servers and K output fiber multi-severs.

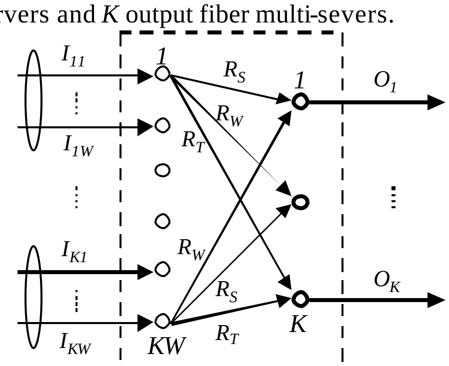


Fig. 3. WTS switch model.

By applying the contention resolutions in wavelength, time, and space domains, each input single-server I is associated with three output multi-servers: the wavelength domain output multi-server O_W (associated with its first preferred output port), the time domain output multi-server O_T (associated with the shared output recirculation buffer port), and the space domain output multi-server O_S (associated with its second preferred output port). In one arbitration operation, each single-server may send three forwarding requests to its three associated output multi-severs and each output multi-server may grant at most W forwarding requests and then send back at most W grants to the input single-servers. Three types of grants are defined, *i.e.*, the wavelength domain grant G_w, time domain grant G_T, and space domain grant G_S. The arbitration process may consist of one request phase, one grant phase, and one optional accept phase. The input single-server may send multiple requests either sequentially or in parallel. The requests, grant, and accept messages are denoted as the directional edges in the bipartite graph. Hence, the bipartite graph can be expressed as $G=(V, E)$, where

- $V=I$? O
- I={ $I_0, I_1, …, I_{KW-1}$}: input single-servers, $|I|=KW$,
- O={$O_0, O_1, …, O_{K-1}$}: output multi-servers, $|O|=K$.
- E={forwarding requests from input single-servers I to output multi-servers O}.

Thus, assuming a bipartite graph model G, the OPS switch fabric arbitration problem is defined to find the maximum number of edges between the input single-servers and output multi-servers so that no output multi-sever is connected to more than W edges and no input single-server is starved if there is still available servers on the output side.

C. sWTS Arbitration Algorithm

The key point for the proposed sequential Wavelength-Time-Space (sWTS) arbitration algorithm is that each input single-server sends the wavelength, time, and space domain forwarding requests to the three associated output multi-servers (the first preferred output port, the recirculation buffer output port, and the second deflection output port) sequentially in time axis. The input single-sever first sends its wavelength domain request. If the request is granted, the arbitration process is finished. Otherwise, the input-single sever will continue to send its time and space domain requests sequentially. Thus, in the worst case the overall arbitration period of sWTS consists of three sequential arbitration steps: Wavelength Domain Arbitration Step, Time Domain Arbitration Step, and Space Domain Arbitration Step as shown in Fig. 4.

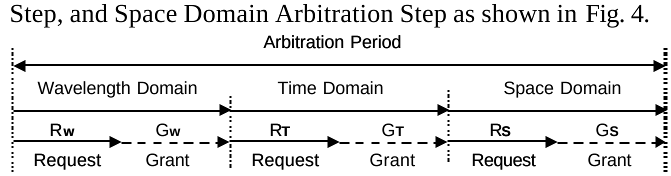


Fig. 4. sWTS arbitration algorithm.

In each sub step, the input single-server sends at most one forwarding request. Thus, it may receive at most one grant from the corresponding output multi-server in each step. With the sequential arbitration process, the sWTS algorithm has the following two characteristics. First, each arbitration step only needs to accommodate two phases: the request phase and the grant phase. Second, unlike most electronic packet switch arbitration algorithms, no iterative arbitration operation is needed to maximize the matching since each input single-server sends no more than one request and receives at most one grant at one time. These two characteristics facilitate a very fast and hardware-efficient arbiter implementation architecture as discussed in Section V. The arbitration operation in each step is explained as follows.

- **Step 1: Wavelength Domain Arbitration.**

Upon the arrival of an optical packet, a wavelength domain forwarding request R_W is sent to the output port arbiter (named wavelength domain arbiter) associated with the packet's first preferred output port. The wavelength domain arbiter makes the arbitration decision and assigns the wavelength domain grant G_W to each granted packet. Then the granted packets will be transmitted to the output port.

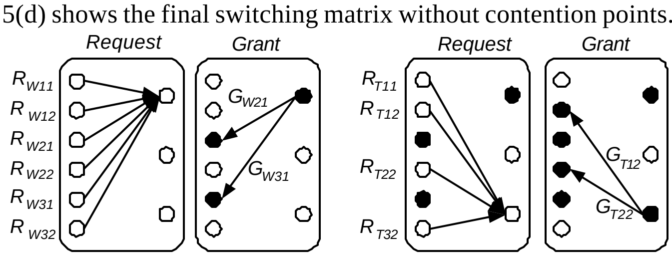
- **Step 2: Time Domain Arbitration**.

The un-granted packet after wavelength domain arbitration sends time domain forwarding request R_T to the output port arbiter (named time domain arbiter) associated with the recirculation buffer ports. The time domain arbiter makes the arbitration decision and assigns the time domain grant G_T to each granted packet. Then the granted packets will be transmitted to the recirculation port for buffering.

- **Step 3: Space Domain Arbitration**.

If the packet is still not granted in step 2, the un-granted packet sends space domain forwarding request R_S to the output port arbiter associated with the packet's *second* preferred output port (named space domain arbiter). The space domain arbiter makes the arbitration decision and assigns the space domain grant G_S to each granted packet. Then the granted packets will be *deflected* to their second preferred output port and expect the neighboring nodes to forward them to their destinations.

Fig. 5 shows an example of sWTS arbitration algorithm and illustrates how contentions are resolved in each arbitration step. Assume the switch fabric has three input ports and three output ports, and each port carries two wavelength channels. In addition, each input channel has one packet destined to the same output port 1. The WTS switch model has six input single-servers and three output multi-servers. Fig. 5 (a) shows that in the wavelength domain request phase, each input single-server sends one wavelength domain forwarding request to the same output port 1. In the wavelength domain grant phase, R_{W21} and R_{W31} are granted. Fig. 5 (b) shows that in the time domain request phase, the four un-granted single-servers send time domain forwarding requests to the same shared recirculation buffer port. Then R_{T12} and R_{T22} are granted. Fig. 5(c) shows that in the space domain request phase, the remaining two un-granted single-servers send forwarding requests to their second preferred port and both them are granted. Finally, Fig. 5(d) shows the final switching matrix without contention points.



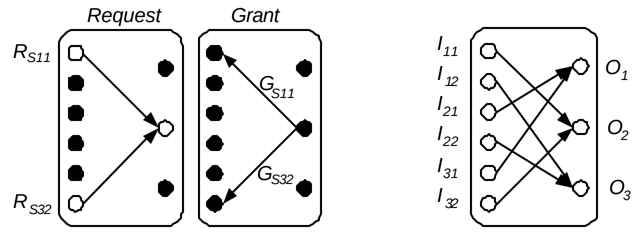


Fig. 5. An example of sWTS arbitration algorithm, (a) wavelength domain, (b) time domain, (c) space domain, (d) final switching matrix after arbitration.

V. Simulation Analysis

For the proposed WTS OPS switch model, the number of wavelength channels and the number of recirculation buffer ports are two most important switch fabric parameters. This work presents simulation studies driven by the self-similar traffic to investigate the impact of these two parameters on the proposed sWTS arbitration algorithm performance. This work simulates an OPS network, where the network-wide algorithm performance is evaluated with uniform background traffic present. This work adopts a self-similar model called "Sup_FRP" to generate packet traces with Hurst parameter $H = 0.8$. The simulation is conducted with OPNET modeler. Fig. 6 shows the 6-node WDM simulation network, where each WDM fiber link carries multiple wavelengths transmitting at 2.5 Gb/s.

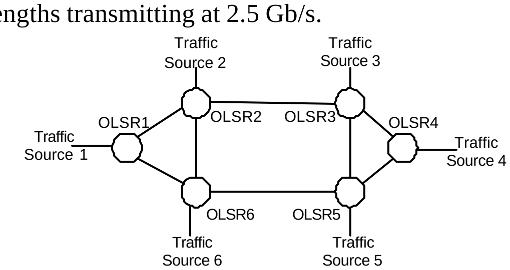


Fig. 6. Simulation networks setup.

Each node denotes an OPS router with one dedicated traffic source to generate IP packets with a realistic IP packet length distribution shown in Fig. 7 (the average packet size = 404.5 bytes, the maximum packet length = 1500 bytes, and the minimum packet length = 40 bytes).

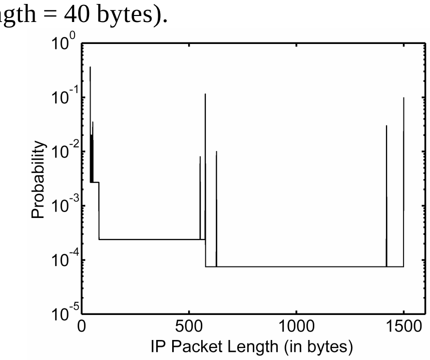


Fig. 7. IP packet length distribution used for simulations.

Fig. 8 and 9 show the simulation results, where the load of the local transmitter, defined as the ratio between the total numbers of bits offered per unit time and the line-speed, varies from 0.3 to 0.7 in the simulation. Table I shows the abbreviations used in the simulation results.

TABLE I. Switch Fabric Configurations

2W-1B	2 channels per port with 1 recirculation buffer port.
2W-2B	2 channels per port with 2 recirculation buffer port.
4W-1B	4 channels per port with 1 recirculation buffer port.
4W-2B	4 channels per port with 2 recirculation buffer port.

A. Arbitration Grant Ratios in Wavelength, Time, and Space Domains

Fig. 8 (a), (b), and (c) investigate the packet grant ratios (GR_w, GR_T, GR_S) in each domain. This gives an insight of the number of requests granted in each domain. The grant ratio in each domain is defined as follows, where NG_W, NG_T and NG_S are the number of packets granted in wavelength, time, and space domains respectively.

TABLE II. Grant Ratio

GR_W	$NR_W/(NR_w+NR_T+NR_S)$
GR_T	$NR_T/(NR_w+NR_T+NR_S)$
GR_S	$NR_S/(NR_w+NR_T+NR_S)$

Wavelength domain arbitration simulation analysis

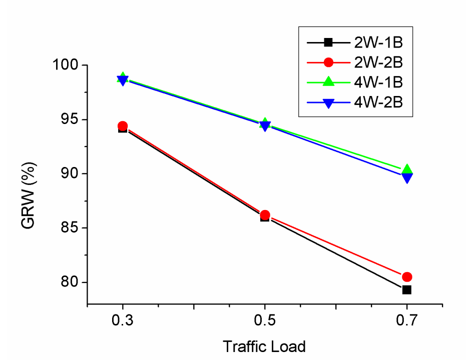
Fig. 8 (a) shows the wavelength domain packet grant ratio GR_W. At traffic load 0.7, GR_w is more than 80%, which implies that among all the granted packets, the wavelength domain granted packets take the largest proportion. This is due to the wavelength domain arbitration occurring prior to the time domain arbitration and space domain arbitration in sWTS algorithm. When the traffic load increases, more forwarding requests resort to the time domain arbitration and space domain arbitration to resolve the contentions as shown in the drop of GR_w with the increase in traffic load in Fig. 8 (a). Comparing *4W-1B*-GR_w and *4W-2B*-GR_w with *2W-1B*-GR_w and *2W-2B*-GR_w, it also shows that when the number of wavelength channels increases from 2 to 4, the wavelength domain GR_w increases greatly because the wavelength domain arbitration may accommodate more forwarding requests.

Time domain arbitration simulation analysis

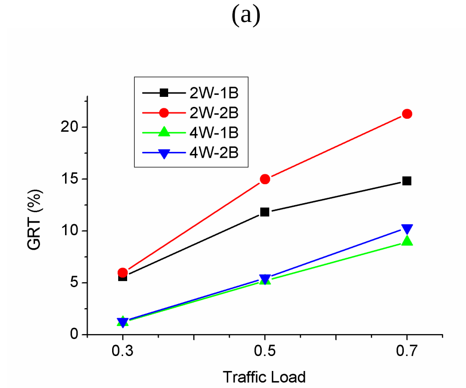
Fig. 8 (b) shows the time domain grant ratio GR_T. At traffic load 0.7, GR_T is less than 20%, which means less than 20% packets are granted in time domain Fig. 8 (b) shows that GR_T increases when the traffic load increases since more forwarding requests resort to the time domain arbitration to resolve the contentions. When the number of wavelength channels increases, GR_T drops because more packets are granted during the previous wavelength domain arbitration. When the number of recirculation buffer port increases, GR_T increases greatly because more recirculation ports may accommodate more requests.

Space domain arbitration simulation analysis

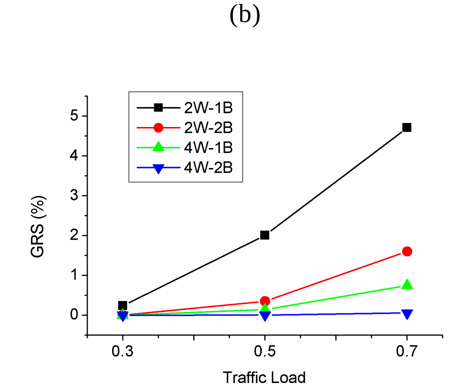
Fig. 8 (c) shows the space domain grant ratio GR_S. Fig. 8 (c) shows that GR_S increases to 5% when the traffic load increases to 0.7, indicating that at most 5% packets are granted in space domain. When the number of wavelength channels increases, GR_S drops. Both the increasing of the number of wavelength channel per port and the recirculation buffer ports leads to a smaller GR_S because more contentions can be resolved in the prior two domains.



(a)



(b)



(c)

Fig. 8. Grant ratios in (a) wavelength domain, (b) time domain, (c) space domain.

B. System Performance Analysis

Packet blocking rate analysis

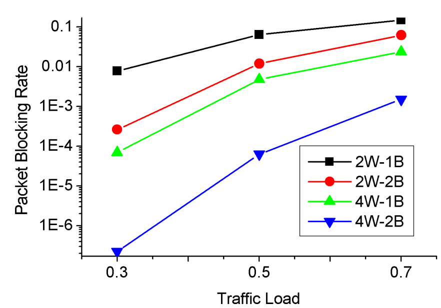
When all the three domains deny the forwarding request, the packet is blocked. Fig. 9 (a) compares the packet blocking rate of the four switch architecture configurations. It shows that the blocking rate drops greatly as the number of wavelength channels per port and the number of recirculation buffer ports increase.

Average packet forwarding latency and jitter analysis

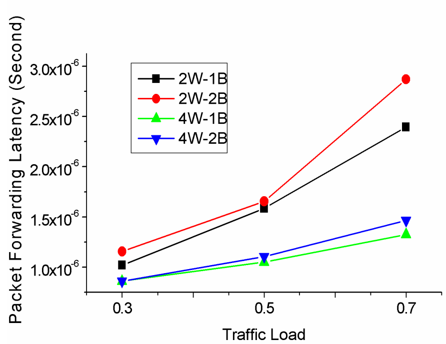
The forwarding delay in each node is mainly introduced when the packets go through the recirculation buffer delay line ports since most packet contentions are resolved in wavelength and time domains in our simulation network according to Fig. 8. Fig. 9

(b) compares the per-node forwarding delays of the four switch architecture configurations. It shows that the forwarding latency per node drops greatly when the number of wavelength channels per fiber increases from 2 to 4 because more contentions are resolved in wavelength domain without introducing delays. Also, the figure shows that as the number of recirculation buffer ports increases from 1 to 2, the forwarding latency increases because more contentions are resolved by fiber delay lines ports, which leads to a larger average packet forwarding latency. Fig. 9 (c) compares the timing jitter of the four switch architecture configurations.

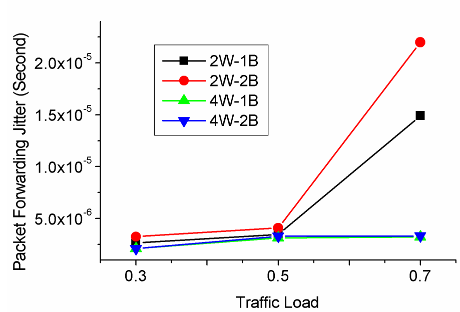
Fig. 9 (a), (b) and (c) show that increasing the wavelength domain arbitration capability is an effective way to drop the arbitration blocking rate and decrease the forwarding delay and jitter. Increasing the number of recirculation ports may decrease the packet loss rate at the price of increasing average forwarding delay and jitter.



(a)



(b)



(c)

Fig. 9. System performance: (a) arbitration blocking rate, (b) packet forwarding latency, (c) packet forwarding jitter.

VI. PORT ARBITER HARDWARE IMPLEMENTATION

Fig. 10 shows the port arbiter implementation framework: the mixed-tree optical packet switch arbiter architecture (MTA). The MTA arbiter determines which input requests are granted and which output wavelength channel is assigned to each granted input request. The port status base is a data structure to keep track of the status of each wavelength on each port. To support variable length packets it consists of a counter that keeps track of the amount of transmission time left. The port arbiter module uses this information to decide the number of requests granted during an arbitration cycle. MTA consists of two phases: the request aggregation phase and the token distribution phase.

1) Request Aggregation Phase

The Request Aggregation Phase adopts the binary tree structure to aggregate the input requests in order to save the aggregation time. Each leaf node computes the number of forwarding requests from one input port stores it and propagates it to the internal parent node. Each internal node stores the two input sums from its two child nodes respectively, adds them, and outputs it to its parent node.

2) Token Distribution Phase

The Grant Generation Phase uses another binary round-robin tree structure to distribute the available wavelength channels to each input port fairly.

To increase the arbitration speed, the request aggregation phase can be pipelined with the token distribution phase, which leads to a 2-stage pipelined architecture. The token distribution phase can be further logically divided into two sub-phases: the inter-port distribution sub-phase and the intra-port distribution sub-phase, which leads to a 3-stage pipelined implementation architecture.

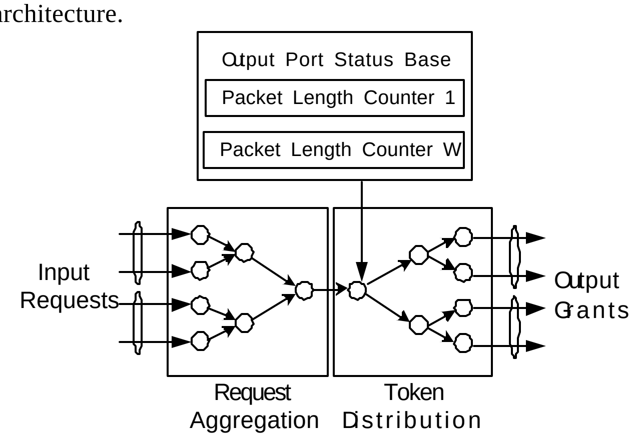


Fig. 10. Mixed-tree arbiter framework.

We implemented our designs on a Xilinx 1000E FPGA chip. We assume 8 input/output fibers. Fig. 11 shows the un-pipelined (MTA), 2-stage pipelined (MTA-2S), and the 3-stage pipelined (MTA-3S) implementations. For each design, the port carries 2, 4, 8, 16, 32 wavelength channels (W=2, 4, 8, 16, 32). The clock period denotes the arbitration period, rate at which new input requests can be processed. The 3-stage pipelined design may efficiently decrease the clock cycle and increase the system frequency.

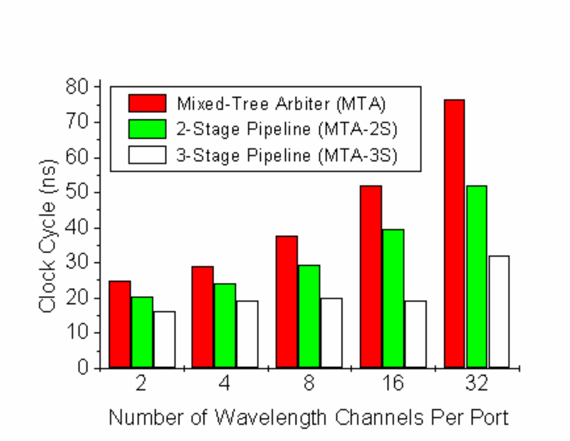


Fig. 11. Comparison of various pipelined implementations.

Table III shows the FPGA implementation results. The area is shown in Virtex slices and 4-input LUTs. Note that XCV1000E has 24576 LUTs, so we can easily fit the 8 arbiters required in our design on one XCV1000E chip.

TABLE III
FPGA AREA AND TIMING RESULTS

Design	Slices	LUTs	Clock Cycle (ns)
16×16	239	435	21.608
32×32	639	1148	23.905

Table IV shows the ASIC implementation of our designs on a 0.25-micron commercial standard library. The area is expressed in square microns.

TABLE IV
ASIC AREA AND TIMING RESULTS

Design	Area (square micron)	Clock Cycle (ns)
16×16	114451	4.82
32×32	198460	5.55

VII. CONCLUSION

This paper proposes the novel sWTS algorithm to solve the all-optical packet switch fabric arbitration problem and shows both the simulation and hardware implementation results. The proposed sWTS algorithm combines contention resolution in wavelength, time, and space domains. The simulation results with self-similar traffic demonstrates that the new arbitration algorithm effectively improve the overall system performance in term of the packet blocking rate, packet forwarding delay and jitter as the numbers of wavelength channels per port and the number of recirculation buffer ports increase. This paper further presents a new hardware arbiter design called mixed-tree arbiter (MTA) suitable for high-performance optical packet switches. We verify and implement the MTA on both Xilinx FPGA and ASIC. Our results show that the 3-stage pipelined MTA achieves a high arbitration rate for FPGA implementation. The performance is further improved (~five-fold improvement) with an ASIC implementation.

VIII. REFERENCE

[1] S. Yao, B. Mukherjee, and S. Dixit, “Advances in Photonic Packet Switching: An Overview,” *IEEE Commun. Mag.*, vol. 38, no. 2, pp. 84–94, Feb. 2000.

[2] S. L. Danielsen, P. B. Hansen, and K. E. Stubkjear, “Wavelength conversion in optical packet switching,” *J. Lightwave Tech.*, vol. 16, pp. 2095–2108, Dec. 1998.

[3] I. Chlamtac *et al.*, “CORD: contention resolution by delay lines,” *IEEE J. Select. Areas Commun.*, vol. 14, pp. 1014–1029, Jun. 1996.

[4] D. K. Hunter, M. C. Chia, and I. Andonovic, “Buffering in optical packet switches,” *J. Lightwave Tech.*, vol. 16 , pp. 2081–2094, Dec. 1998.

[5] F. Forghierri, A. Bononi, and P. R. Prucnal, “Analysis and comparison of hot-potato and single-buffer deflection routing in very high bit rate optical mesh networks,” *IEEE Trans. Commun.*, vol.43, pp. 88–98, Jan. 1995.

[6] N. McKeown, “The iSLIP scheduling algorithm for input-queued switches”, *IEEE/ACM Transactions on Networking*, Vol. 7, No. 2, pp. 188-201, April 1999.

[7] M. Yang and S Q. Zheng, “The kDRR scheduling algorithms for multi-server packet switches”, in Proc. *the ISCA 15th International Conference on Parallel and Distributed Computing Systems (PDCS)* 2002, pp. 78-83.

[8] P. Gupta and N. Mckeown, “Designing and implementing a fast crossbar Scheduler,” *IEEE Micro*, Vol. 19, No. 1, pp. 20-29, 1999.

[9] H. J. Chao, C. H. Lam, and X. Guo, “A fast arbitration scheme for terabit packet switches”, *Proc. of Globecom*, pp. 1236-1243, 1999.

[10] K. Lee, S. Lee, and H. Yoo, “A distributed on-chip crossbar switch scheduler for on-chip networks”, *IEEE Custom Integrated Circuits Conference (CICC)*, pp. 671-674, Sep. 2003. San Jose, CA.

[11]S. Q. Zheng, M. Yang, and F. Masetti-Placci, “Hardware switch scheduling in high-speed, high capacity IP routers,” *Proc. 14th Conference on Parallel and Distributed Computing and Systems (PDCS)*, pp. 636-641, Nov. 2002.

[12].Y.-S. Yeh, M. G. Hluchyj, and A. S. Acampora, “The knockout switch: a simple, modular architecture for high-performance packet switching,” IEEE J. Select Areas Commun., vol. 5, no. 8, pp. 1274-1283, Oct, 1987.

[13] S. J. B. Yoo, G. K. Chang, “High-throughput, low-latency next generation Internet using optical-tag switching,” *U.S. Patent* 6,111,673, 1997.