LIONS: An AWGR-Based Low-Latency Optical Switch for High-Performance Computing and Data Centers

LIONS: An AWGR-based Low-latency Optical Switch for High Performance Computing and Data Centers

Yawei Yin, *Member, IEEE,* Roberto Proietti, *Member, IEEE,* Xiaohui Ye, *Student Member, IEEE,* Christopher J. Nitta, *Member, IEEE,* Venkatesh Akella, *Member, IEEE,* S. J. B. Yoo, *Fellow, IEEE*

***Abstract*—This paper discusses the architecture of an Arrayed Waveguide Grating Router (AWGR) based low-latency optical switch called LIONS, and its different loopback buffering schemes. A proof of concept is demonstrated with a 4x4 experimental testbed. A simulator was developed to model the LIONS architecture and was validated by comparing experimentally obtained statistics such as average end-to-end latency with the results produced by the simulator. Considering the complexity and cost in implementing loopback buffers in LIONS, we propose an all-optical negative acknowledgement (AO-NACK) architecture in order to remove the need for loopback buffers. Simulation results for LIONS with AO-NACK architecture and DLB architecture are compared with the performance of the flattened butterfly electrical switching network.**

***Index Terms*—Optical Switches, Optical Interconnections, Optics in Computing, Buffers in Switches**

I. Introduction

WHILE the advances in CMOS technologies continue to support Moore's Law [1] in increasing the number of transistors in a die, microprocessor performance is expected to continue to increase with parallel processing of many cores on-chip. The Amdahl's law [2] suggests that a parallel computing system with balanced processing, memory, and communications performs best across most applications. A balanced system with 100 TeraFLOPs/second computing speed and 100 TeraBytes memory would need 100 TeraByte/second (800 Tb/s) bisection bandwidth. The future high performance computing (HPC) systems and Data Centers implemented with multicore processors will require terabytes/sec of bandwidth for processor to processor and processor to memory. It is expected to be increasingly difficult to meet the scalable, high-bandwidth density and low latency communication requirements of these future large data centers and terascale computing systems using conventional electrical interconnects.

Manuscript received June 1, 2012. This work was supported by the Department of Defense (contract #H88230-08-C-0202) and Google Research Awards

Y. Yin, R. Proietti, X. Ye, C. Nitta, V. Akella and S.J.B.Yoo are with the Department of Electrical and Computer Engineering and Department of Computer Science, University of California, Davis, CA, 95616 USA (email: yyin@ucdavis.edu).

Optical interconnects exploiting the inherent Wavelength Division Multiplexing (WDM) parallelism could overcome those limitations. In the past few years some efforts have been made in designing optical switching architecture by considering the particular challenges faced by the networks supporting datacenters and high-performance computers. Among all the proposed optical switching architectures in literature, the Optical Shared Memory Supercomputer Interconnect System (OSMOSIS) [3, 4], Data Vortex [5] and Low-latency Interconnect Optical Network Switch (LIONS, previously named as DOS) [6] are three pioneering architectures that have attracted a fair amount of attention. The simulation comparison in [6] shows that LIONS provides low-latency and high-throughput switching and does not saturate even at very high (~90%) input load. In addition, the average latency and throughput of LIONS do not change dramatically as the port number increases, which is scalable beyond that of OSMOSIS or Data Vortex.

This paper discusses the architecture of LIONS together with its different loopback buffering schemes. Specifically, this paper presents the hardware demonstration of a 4x4 LIONS testbed, and compares the experimental results with simulation results to verify the correctness of the simulator developed to model the LIONS architecture. The experimental and simulation studies indicate that the loopback buffers in the LIONS architecture will be the main limiting factor for scalability. In order to overcome this limitation, we propose the All-Optical Negative ACKnowledgement (AO-NACK)

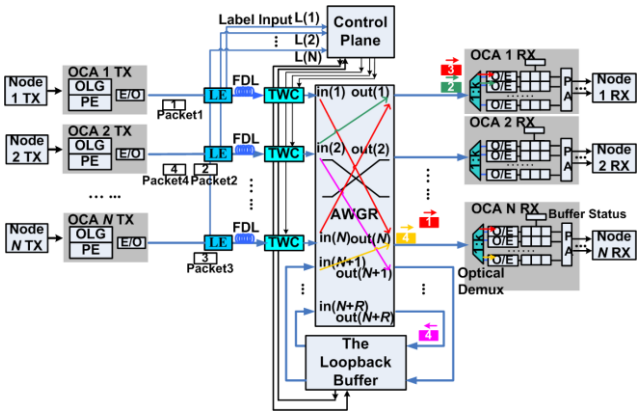


Figure 1 LIONS Architecture

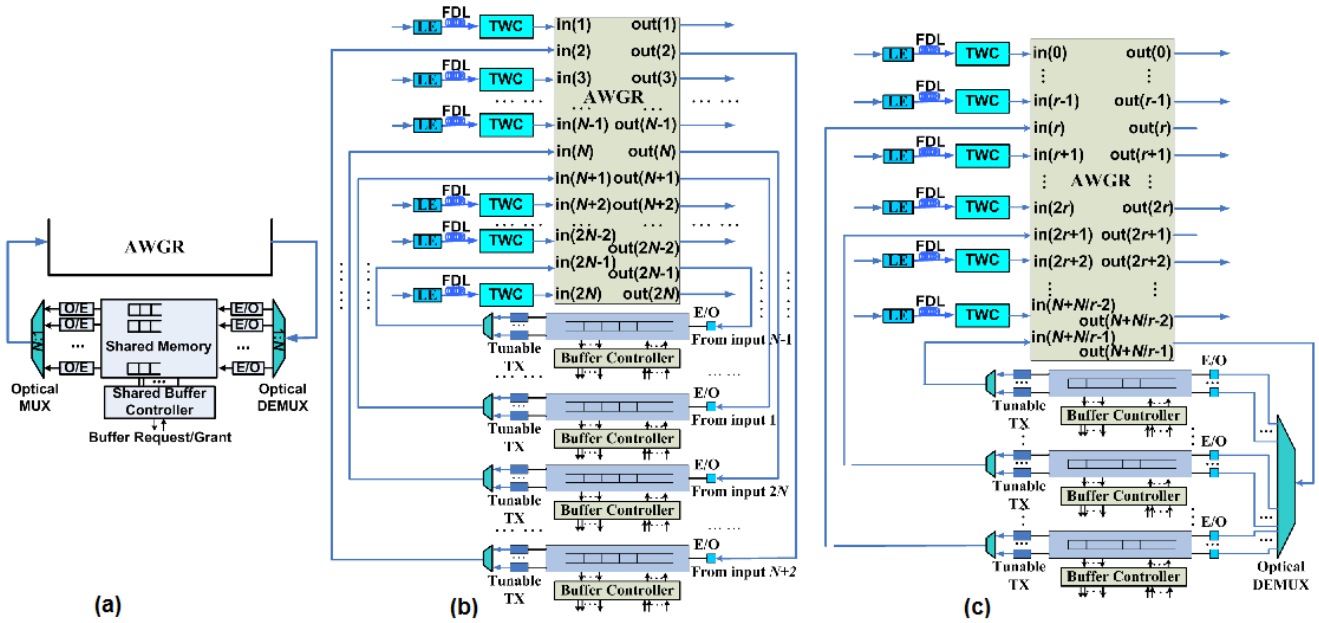


Figure 2. LIONS with (a) shared loopback buffer, (b) distributed loopback buffer, and (c) mixed loopback buffer

architecture. Moreover, the synthetic traffics as well as GUPS benchmarks are used to evaluate the performance of AO-NACK architecture in comparison with traditional LIONS with loopback buffers and the electrical switching network using a Flattened Butterfly (FBF) topology.

The remainder of this paper is organized as follows. Section II describes the comprehensive architecture of LIONS together with the different loopback buffer strategies. Section III presents the proof of concept testbed demonstration of a 4x4 LIONS with shared loopback buffer architecture, and compares the experimentally collected statistics with the simulation results in order to verify the accuracy of our simulator. In Section IV introduces the LIONS with AO-NACK architecture which removes the scalability limitation of the loopback buffer. The results of performance simulations comparing the different architectures on both synthetic and benchmark workloads are also presented here. Section V concludes this paper.

II. LIONS With Different Loopback Buffering Schemes

It is well known that traditional electrical switch output queuing is capable of achieving lower switch latency and higher throughput than input queuing [7] if the necessary speedup (x*N*) is incorporated in the output queue. However, when the line rate and the switch radix increase, output queuing becomes difficult to implement (primarily because the output queues with high aggregated bandwidth are difficult to realize). Therefore, electrical switch designs focus on complicated input queuing structures, such as Virtual Output Queue (VOQ), and associated multi-stage arbitration schemes. AWGR switching fabric does not suffer from this limitation as they have the unique strength of wavelength parallelism, which allows optical wavelengths to cross over and propagate in parallel.

Figure 1 shows the general architecture of LIONS, which has at its core an AWGR, Tunable Wavelength Converters (TWCs), an electrical control plane, electrical loopback buffers, label extractors, and Fiber Delay Lines (FDLs). Between the switch and each end-node, there is an Optical Channel Adapter (OCA) that serves as the media interface[6]. In particular, the AWGR based switching fabric can easily realize the output queue, provided that a 1:*N* optical DEMUX with *N* receivers is available at each AWGR output. However, requiring *N* receivers at each output may not be practical or scalable since this requires a total of N^2 receivers for the whole switch. We assume that each output is equipped with a 1:*k* optical DEMUX and *k* receivers with $k < N$, thus realizing an output queuing with a speedup of *k*. We define a *wavegroup* as a set of wavelengths that will emerge from the same output port of the 1:*k* optical DEMUX.

In an optical switch, the store-and-forward mechanism cannot be applied due to the lack of feasible optical buffers. In the proposed LIONS switch, the electronic buffers are placed in the loopback path, referred to here as the loopback buffer. Note that the LIONS switch uses a *forward-store* strategy, as opposed to the store-and-forward strategy employed in an electrical switch. Only the contended packets that fail to get grants from the arbiter are stored. The loopback buffer in LIONS plays an important role in contention resolution. In this section, we compare the three proposed loopback buffer architectures in LIONS, referred to as the shared loopback buffer (SLB), the distributed loopback buffer (DLB) and the mixed loopback buffer (MLB).

A. Shared Loopback Buffer

The SLB has the structure shown in Fig.2(a). A 1:*N* optical DEMUX and *N* receivers are necessary, because the SLB may receive delayed packets from different inputs concurrently on different wavelengths. A *N*:1 optical MUX and *N* transmitters are also required to allow the SLB to send delayed packets to different outputs on different wavelengths concurrently. In the SLB, the optical DEMUX and MUX can both be realized by a 1:*N* AWG. The packets received on different wavelengths will

be copied to the shared memory in parallel. All delayed packets will be stored in the shared memory before transmission. The queuing structure in the shared memory is organized based on outputs, since the packet destined for the same output should be sent out serially. The transmission for a delayed packet can start before the entire packet arrives at the SLB. When a grant is given to the SLB for a particular output, the SLB sends out all delayed packets going to that output serially.

The benefits of the SLB are the following. First, the total size of the buffer can be small, as one input port can cause contention at only one output, but not at multiple outputs, at any time. Second, the SLB can have a simple buffer controller; since packets are stored based on outputs, no further scheduling among delayed packets is required. Nevertheless, the main drawback of the SLB is that the shared memory limits the scalability of the optical hybrid switch, since the required memory I/O bandwidth is proportional to both the switch radix and the data rate on each wavelength.

B. Distributed Loopback Buffer

The loopback buffer will become more scalable if the queues can be organized based on the input ports instead of on the output ports. The input-based buffer can be realized in a distributed manner with multiple separate buffers. For each buffer, the required memory I/O bandwidth can be reduced by a factor that is proportional to the number of separate buffers. Therefore, the loopback buffer can support a switch with a higher port count and a higher data rate.

The proposed DLB has N separate memory units to realize N separate queues, with each unit serving delayed packets from one particular switch input. In the simplest case, each queue has one transmitter and does not adopt VOQ. However, the buffer controller design will become more complicated, as contention may occur among queues for different inputs. Moreover, the head-of-line (HOL) blocking may occur, and end-to-end latency may increase.

Instead of using VOQs with complicated arbitration to alleviate the effect of the HOL blocking, we can also exploit the intrinsic wavelength parallelism to do so and keep the arbitration relatively simple. We can deploy multiple transmitters for each queue, thus making it capable of sending multiple packets to different switch outputs on different wavelengths concurrently. Considering scalability and cost, only a fixed number of transmitters should be deployed for each queue. The packets that are waiting for transmission at the head of each VOQ should gain the grant of the transmitters first, and then the one who wins one transmitter can make the request to the arbiters in the control plane. In this case, with multiple transmitters, multiple VOQs in each queue can make more than one request at a time to the control plane arbiter, and thus make more efficient use of the resources and avoid HOL blocking.

The DLB uses tunable transmitters, because packets stored at one queue may go to different AWGR outputs. Therefore, each queue in the DLB should be connected to a separate AWGR input. Then a $2*N \times 2*N$ AWGR is needed to connect to N end nodes and the DLB, as shown in Fig. 2(b). The blocked packets from a single AWGR input port are all directed to a certain AWGR output that is connected to the dedicated queue in DLB. Each input queue in DLB corresponds to an AWGR input port. Couplers are used to realize the optical MUX if each queue has multiple transmitters. We can interleave the inputs connecting with the end nodes and the inputs connecting with the loopback queues so that, in each contention group, half of the inputs connect with end nodes and half of the inputs connect with loopback queues. With this connection, k wavegroups for one output will be used equally. Multiple delayed packets going to the same output can be sent out concurrently from different queues on different wavelengths, thus reducing the end-to-end latency. Moreover, through the careful association of the inputs connected to end nodes with the inputs connected to queues, the waiting time for a packet at the DLB can be further reduced. For example, we can connect an end node to the input i and connect its corresponding DLB queue to the input j ($0 \leq i, j \leq 2*N$). Assuming there are at least two contention groups, then if i and j satisfy the relation $j=mod(i+N+1, 2*N)$, they belongs to different contention groups. Therefore, if a packet first reaches the input i but the request is rejected, it will make a request to a different contention group when it comes out of the queue in DLB.

C. Mixed Loopback Buffer

The DLB can achieve lower end-to-end latency than can the SLB. The I/O bandwidth requirement for each memory unit increases only when the data rate increases and not as the port count increases. However, the DLB occupies N AWGR ports to support the queues for N end nodes, while the SLB occupies only one AWGR port. The DLB also requires more transmitters than does the SLB in order to alleviate the HOL blocking. To achieve the benefits from both the SLB and the DLB while mitigating their disadvantages, we propose the Mixed Loopback Buffer, as shown in Fig. 2(c). The MLB still occupies multiple AWGR inputs to support multiple separate queues, so that each queue occupies one AWGR input. Unlike the DLB, in which each queue serves only one end node, each queue in the MLB serves r end nodes and connects to r outputs of the 1:N optical DEMUX. Therefore, the MLB occupies N/r AWGR inputs and one output if the switch connects to N end nodes. Note that in principle, the MLB could also use N/r output ports on AWGR, but that would require N/r of 1:r optical DEMUX at the input of each queue in MLB. Again, each queue in the MLB can have multiple transmitters to alleviate the HOL blocking. Since the MLB has only N/r queues, even if each queue has r tunable transmitters, the MLB has N tunable transmitters in total, which is much fewer than the number of tunable transmitters required by the DLB. All the couplers used in LIONS are assumed to be *2:1* couplers for simplicity. Therefore, for MLB with r tunable transmitters at each queue, (r-*1*) couplers are needed for each queue, and since there are (N/r) queues in total, (r-*1*)*(N/r) couplers are required in total. In the MLB, we can adopt scheduling and arbitration similar to those used in the DLB to schedule the transmission for delayed packets. At each queue, a delayed packet must first gain a transmitter and then make a request to the control plane. Since the MLB occupies multiple AWGR inputs, as in the

DLB, the MLB can send multiple delayed packets to the same AWGR output if we interleave the ports connecting with the end nodes and those connecting with the MLB. Furthermore, through the careful assignment of the inputs to the MLB and association of the inputs connected to end nodes with the inputs connected to the queues, a delayed packet can make a request to a different contention group upon arriving at the MLB. For example, a set of AWGR inputs $\{j|\text{mod}(j, r+1)=0, 0\le j<N+N/r\}$ can be used to connect with N/r queues of the MLB; the queue connected to a particular input j can be used to serve delayed packets from a input set of $\{i|i=\text{mod}(j+N/2+N/(2*r)+h, N+N/r), 1\le h\le r\}$, so that the set of inputs and the input connected to the corresponding queue always belong to different contention groups. When r is a fixed small number, although the memory used in the MLB requires more I/O bandwidth than the memory used in the DLB does, the former requires much less I/O bandwidth than the memory used in the SLB does. Overall, the MLB solution is a tradeoff between the DLB solution and SLB solution in terms of required I/O bandwidth.

D. Comparison of Different Loopback Buffers

TABLE I
Comparison of Different Loopback Buffers

	SLB	MLB	DLB
Occupied AWGR ports	1	N/r	N
Number of Receivers	N	N (r for each queue)	N (1 for each queue)
Number of Transmitters	N (fixed)	N (tunable, r for each queue)	$N*m$ (tunable, m for each queue)
Number of Optical DEMUX	One 1:N optical DEMUX	One 1:N optical DEMUX	None
Number of Optical MUX	One N:1 optical MUX	None	None
Number of couplers	None	$(r\text{-}1)*(N/r)$	$N*(m\text{-}1)$
Memory Write/Read Bandwidth	$N*B$ (read) $N*B$ (write)	$r*B$ (read) $r*B$ (write)	$m*B$ (read) B (write)
Number of Memory	1	N/r	N

Table I presents a detailed comparison of the three different loopback buffers. The SLB requires the most memory I/O bandwidth, while it occupies only one additional AWGR input. On the other hand, the DLB requires the least memory I/O bandwidth, but it requires more tunable transmitters, and the size of the AWGR must be doubled to support the DLB. While the SLB and the DLB represent the two extremes, the MLB provides a tradeoff between the SLB and the DLB.

The performance evaluation of LIONS with three kinds of loopback buffers are presented in [8]. Overall, LIONS adopting the proposed DLB and MLB can provide better performance compared with the switch using the SLB, as the DLB and the MLB use more AWGR ports for transmitting buffered packets. Therefore, the buffered packets can be sent out on different wavelengths even for the same destination, and the delayed packets do not make requests to the same contention group as when they firstly arrived at the switch. In addition to the gain in performance, the DLB and the MLB require much less memory I/O bandwidth than does the SLB. Although the DLB with multiple transmitters per queue performs a little bit better than MLB, the MLB occupies fewer AWGR ports and requires fewer tunable transmitters.

III. Hardware Demo of LIONS and Experimental Verification of The Simulator

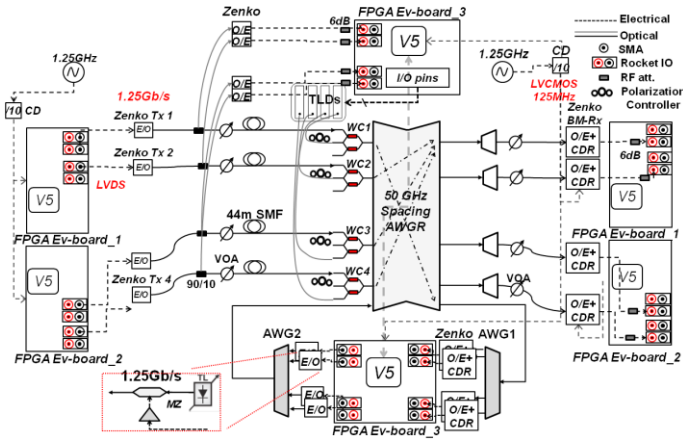


Figure 3. 4x4 LIONS Testbed with shared loopback buffer

Although the performance of LIONS with DLB and MLB is much better than with SLB, the complexity and cost of these two architectures restrain them from a laboratory-scale implementation. Moreover, other than studying the feasibility, the major purpose of implementing LIONS is to verify the correctness of the simulator we developed. No matter which architecture of the loopback buffers is chosen, an accurately modeled LIONS in simulator should have similar or even identical performance with the actually implemented testbed. As far as one of the LIONS architectures can be modeled correctly in a reasonable scale, we will be able to verify that the library of different modules in LIONS have been established accurately in the simulator. Therefore, the projection of LIONS to higher port count or the projection to other buffer architectures become reasonably easy, since it's only a matter of adjusting the parameters or adjusting the buffer arrangements in the simulator. In light of this, we choose the most feasible SLB architecture to implement among all the three architectures of LIONS loopback buffers.

The 4x4 LIONS testbed is depicted in Fig. 3. As shown, a 32x32 50 GHz-spacing AWGR constitutes the core of the switch architecture. It also includes wavelength converters (WCs) based on cross-phase modulation (XPM) in a semiconductor optical amplifier Mach-Zehnder interferometer (SOA-MZI). Each WC accepts one continuous wave (CW) input signal from a tunable laser diode (TLD) board. The TLD guarantee nanosecond switching time over the C band with a wavelength accuracy of 0.02 nm. By reading the 5-bit parallel control signals coming from the FPGA-based control plane, each TLD board tunes its wavelength according to a routing

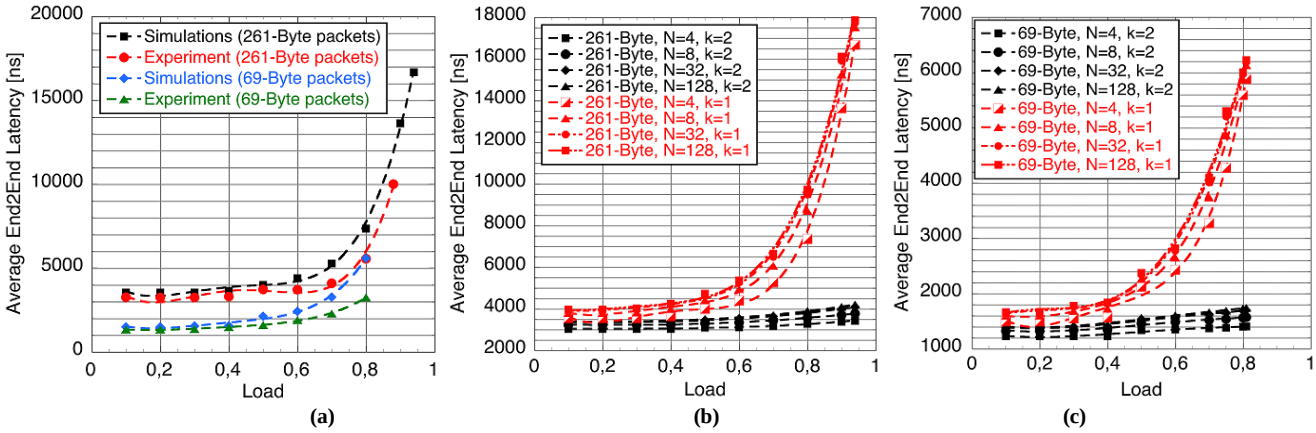


Figure 4 (a) comparison of simulation results with experimental results; (b) projection of simulation results to high port count with large packet size and k=1, 2; (c) projection of simulation results to high port count with small packet size and k=1, 2

table stored on a CPLD chip mounted on the board itself. An optical path is established between each AWGR input and output on a packet basis, according to the destination address carried by each packet label.

The control plane and loopback buffers are implemented using a Xilinx Virtex 5 FPGA ML523 Characterization Platform, which is capable of instantiating 8 high speed RocketIO GTP tile transceivers connected to 16 pairs of differential SMA connectors. Four pairs of the transceivers are used as the 4 control plane channels, each of which receives the labels from its label extractor (i.e. the 90/10 splitter). The labels are encapsulated in the packet headers in the time domain for simplicity. Note that the labels can be wavelength multiplexed to the payload in order to completely separate control and data plane, which is done in the simulator. The control plane reads the label and generates the 5 bit control signals to TLD boards after arbitration. The contended packets that fail to win arbitrations are directed to the inputs of the loopback buffer. The I/O ports of loopback buffers are implemented using another 4 pairs of RocketIO transceivers. The Zenko™ Burst Mode Transceivers (BM-Tx/Rxs) are widely used in the testbed beside the RocketIO™ interfaces of end-hosts, control plane and loopback buffers, and the Burst Mode Clock and Data Recovery (BM-CDR) modules are used at the receivers connected to the output ports of the AWGR, since the WC will turn the optical power on and off while switching.

The LIONS testbed uses two Virtex 5 FPGA platforms to emulate the multiprocessor parallel computing system. Four of MicroBlaze Soft Processor Cores [9] were instantiated on Virtex 5 FPGA with MPI interfaces capable of doing Remote Direct Memory Access (RDMA) operations. The generated data are firstly written to the BRAM block on FPGA and then moved into the RocketIO transmitter output queue using direct memory access (DMA) operation. Then the packets are encapsulated and de-serializd by RocketIO at the 1.25 Gbps output line rate. On the Rx side, the received data packets are directly moved from the input queue to the DDR2 SDRAM memory on board using DMA operation.

The end-to-end latency is one of the important performance metrics to the switch. A synthetic traffic model is used in the testbed. The data streams at each host are encapsulated into fixed size packets with uniform random destination address. Each packet is with 5 byte header (2 byte preamble, 1 byte destination address, 1 byte source address and 1 byte packet length). Different offered load can be achieved by changing the guard time between packets. Note that a minimum guard time of 17 byte has to be guaranteed due to the hardware constraints (i.e. worst case TLD tuning time, burst mode receiver settling time and comma alignment delay in SERDES etc.) Since the traffic is uniform randomly distributed, the end-to-end latency statistics can be collected at any of the output ports. In the experiment, only host 2 was used to collect data.

Figure 4(a) shows the comparison of statistic results from both simulations and experiments. The black and red line shows the 4x4 experimental and simulation data respectively, with k=1 and packet size of 256 bytes plus 5 byte header, while the blue and gray lines shows the same results with packet size of 64 bytes. Here k means the number of parallel wavelengths can be received by the same host simultaneously from one output port of the switch [6]. As shown, the comparison of the results shows a close match between the experimental data and the simulation data, which verifies the correctness and accuracy of the simulator we developed. The other curves in Fig. 4(b) and (c) show the projection of the results to high port count, and also to k=2 case. As depicted, the increase of the LIONS radix does not significantly affect the end to end latency, while k=2 can dramatically reduce it since it reduces the contention probability at each output port.

The testbed implementation was limited by the available off-the-shelf components. Therefore, the line-rate of the testbed was limited because the commercially available Zenko™ BM-CDR modules are running at 1.25 Gbps. Despite the low line rate of the demo, the match between the simulation and experiment is still meaningful considering that the simulation at 10Gbps or higher line rate will only change the packet transmission time. It is important to note in our architecture, the control plane can operate at a lower data rate because the label associated with each packet can be modulated on a separate wavelength. So, even with a FPGA, 10Gbps data rate can be supported with 1.25Gbps label rate and an ASIC based control

plane can be used to go beyond that. . The arbitration process stays the same no matter what line rate is used in the simulation. The scalability of the arbitrations in LIONS switch has been addressed in our previous papers. In particular [6] showed the arbitration logic (which is the bottleneck to scalability to larger number of ports) can be implemented in a distributed manner and scales beyond the traditional electrical switch.

IV. Bufferless LIONS With AO-NACK

Although LIONS can support low latency switching under high input loads, its loopback buffers require complex and fair amount of memory and TX/RX components running at the line rate, which becomes a bottleneck for implementation. The all-optical negative acknowledgement (AO-NACK) technique is capable of eliminating this bottleneck by promptly notifying the end nodes whenever one of their packets cannot reach the desired output due to contention.

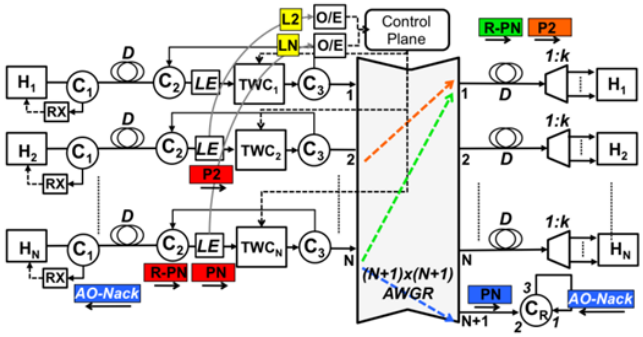


Figure 5. Interconnect architecture using AO-NACK . H: host; C: optical circulator; LE: label extractor; TWC: tunable wavelength converter; AWGR: array waveguide grating router; RX: AO-NACK receiver.

Figure 5 shows an architecture using the AO-NACK technique. N hosts connect to an $(N+1)\times(N+1)$ AWGR by means of a fiber of length D, which represents the host-switch distance. Tunable wavelength converters (TWCs) used at each of the N input ports perform the switching function in the optical domain. Each input port is also equipped with two optical circulators (OCs) to separate the on-the-fly packets (the packets travelling toward the AWGR input ports) from the counter propagating (traveling backwards) AO-NACKs. Label extractors (LEs) separate the low-speed label signals from the high-speed payloads and send the labels to the low-speed electronics control plane (CP). After O/E conversion, the CP processes the label and sends the control signals to the TWCs to switch the packets according to their destination. As explained in detail in [6], optical parallelism in AWGR can be used to reduce contention probability--k inputs can reach the same output port using different wavelengths. Then a 1:k optical demux at each host receiver-side separates the different signals traveling simultaneously on the same fiber. For simplicity let us assume that k=1. If two packets (*e.g.* P2 and PN) from different inputs are contending for the same output (output1), the CP switches one packet (*e.g.* P2) to the desired output, while the other packet (PN) is switched to the N+1 port (reflective port). An OC used as shown in Fig. 5 reflects the packet (PN) back to its sender (H_N). An OC at the host-site (C1) extracts the counter-propagating packet, which now acts as AO-NACK. A dedicated receiver is then used to detect the AO-NACK and trigger the retransmission. If $L/2D \geq 1$, where L is the packet length (in meters), the AO-NACK reaches the sender while the transmission for the related packet is still happening or it has just finished. In this case a simple edge detector is sufficient to detect the AO-NACK since there is no ambiguity about which packet the AO-NACK refers to. If $L/2D < 1$, the received AO-NACK is related to a packet of which the transmission is completed. Since there may be several on-the-fly packets, an edge detector can be still used, but the sender needs to use a time-stamp for each on-the-fly packet. If the counter expires (the time counter value can be fixed since the AO-NACK arrival time is deterministic), the sender can then assume that the associated packet has reached the desired output. Otherwise, packet retransmission is triggered. Another solution could consist of including in the packet header an on-the-fly packet sequence number field of a few bits and then receiving and reading only the first few bytes of the AO-NACK messages. In this case, it is necessary that the AO-NACK technique preserve the packet content. Note that the passive nature of AWGR and OC (C_R) guarantees that this technique can intrinsically reflect multiple packets simultaneously without any crosstalk effect. This aspect, together with the fact that an AO-NACK cannot contend with other packets or other AO-NACKs, makes this technique robust.

A proof of concept demonstration of AO-NACK technique can be found in [10], where a host-switch distance of ≈ 20m and a packet length of 204.8ns are used. The notification of AO-NACKs messages and successful packet retransmission is demonstrated with error-free operation at 10 Gb/s and 40 Gb/s.

In order to evaluate the performance of AO-NACK architecture, the simulator has been developed to correctly model the DLB, AO-NACK and the Flattened Butterfly (FBF) architecture [11]. The DLB architecture was chosen since it represents the best performance among all the three loopback buffer architectures. The simulated FBF consists of a four by four grid of routers with four nodes connected to each router in a 20m by 20m square. Each node is connected to every other node in each row and column. The radix of each router is 10. The distance from each node to its "local" router is 1.75m, with each router 5m from its nearest neighbor. In the DLB and AO-NACK architecture, the distance between each node and the centralized switch is assumed to be 10m. Note that k=4 means that the receiver bandwidth for each AO-NACK node is four times that of the FB nodes – this may appear to be an unfair comparison, however, remarkably, the FBF network has 2.5 times as many links as the AO-NACK [12].

Both synthetic traffic model and benchmarks are used in the simulator to evaluate the performance of LIONS, and the results are compared with the FBF switching network.

The synthetic traffic simulations consisted of uniform random and hot-spot traffic with packet sizes of 256 and 64 bytes. Figure 6 through Fig. 9 shows the throughput and latency for uniform random traffic with 256 and 64 byte packets.

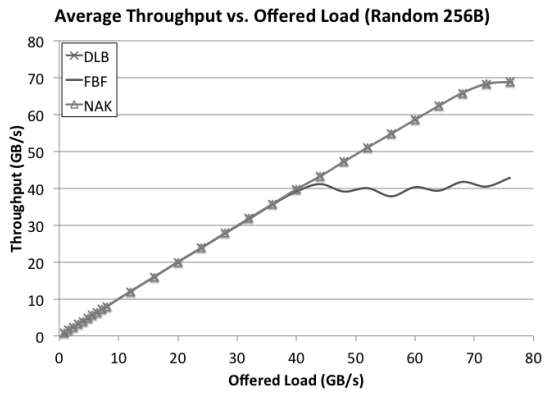


Figure 6. DLB, FBF, and NAK Network Throughput vs. Offered Load for 256B Packets on Uniform Random Traffic

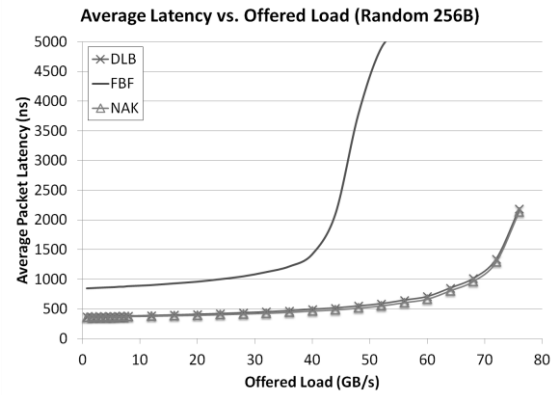


Figure 7. DLB, FBF, and NAK Network Latency vs. Offered Load for 256B Packets on Uniform Random Traffic

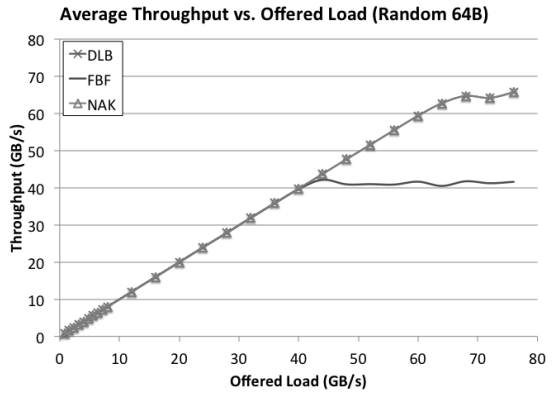


Figure 8. DLB, FBF, and NAK Network Throughput vs. Offered Load for 64B Packets on Uniform Random Traffic

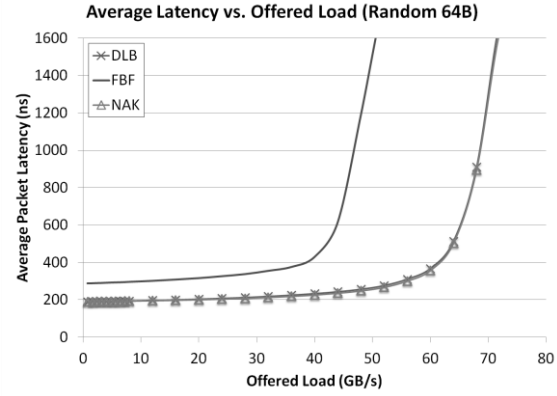


Figure 9. DLB, FBF, and NAK Network Latency vs. Offered Load for 64B Packets on Uniform Random Traffic

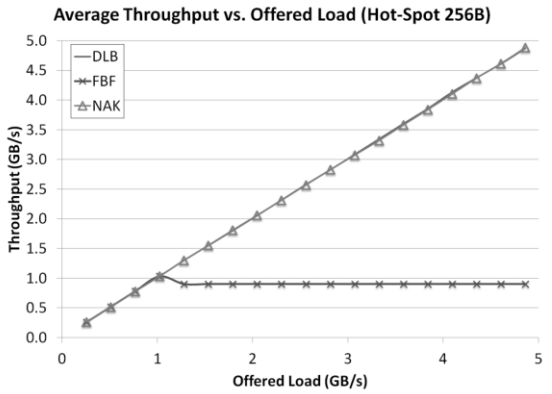


Figure 10. DLB, FBF, and NAK Network Throughput vs. Offered Load for 256B Packets on Hot-Spot Traffic

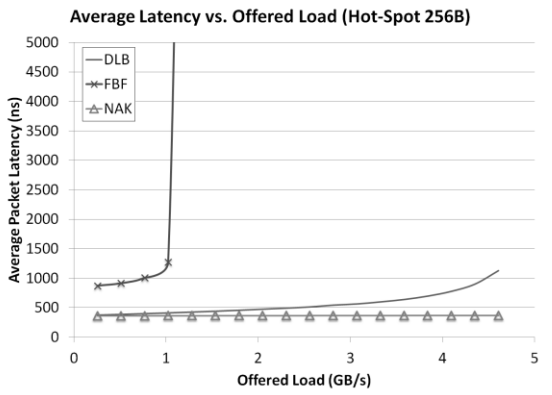


Figure 11. DLB, FBF, and NAK Network Latency vs. Offered Load for 256B Packets on Hot-Spot Traffic

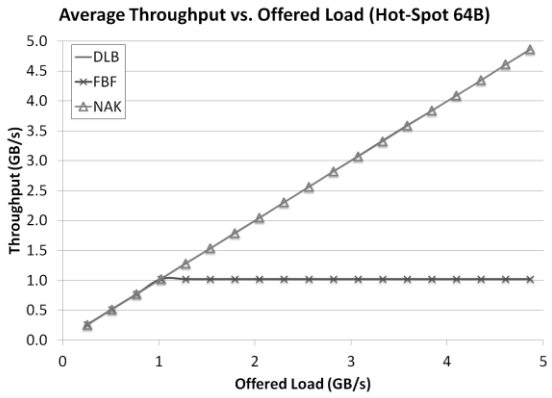


Figure 12. DLB, FBF, and NAK Network Throughput vs. Offered Load for 64B Packets on Hot-Spot Traffic

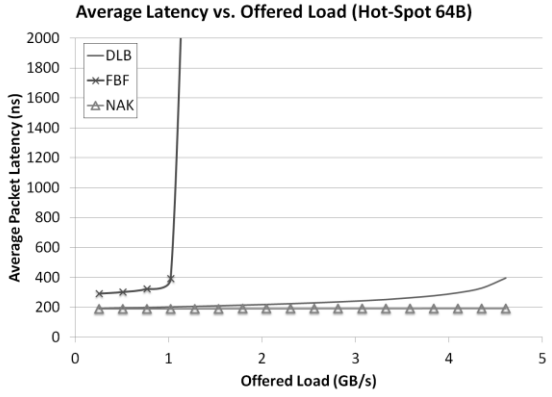


Figure 13. DLB, FBF, and NAK Network Latency vs. Offered Load for 64B Packets on Hot-Spot Traffic

The performance of DLB and the NAK is almost identical for all synthetic traffic patterns – the difference between the two is almost indistinguishable in the figures. The DLB and NAK architectures greatly outperform the FBF for larger packet sizes, but as the packet size decreases the difference between the AWGR based networks and the FBF becomes less apparent. The diminished performance of the AWGR based networks is primarily due to the time required to tune the wavelengths.

The throughput and latency results for hot-spot traffic are shown in Fig. 10 through Fig. 13. Note that the offered load is limited to 5GB/s since the networks are theoretically limited to four 10Gbps receivers. The maximum throughput of the AWGR based networks is four times that of the FBF since K was set to four for these simulations. The AWGR networks are fairly insensitive to the packet size on the hot-spot traffic pattern due to the relatively low total load being offered.

The GUPS benchmark is of particular interest in high performance computation and typical of in-memory database applications that implements *transactional* nature of query processing. Each "update" requires a node to read a random memory location, modify the value and then write back to the same memory location. The GUPS benchmarking for the three networks simulated a 64 node shared memory system with a 64-bit address space. The updates were of 64-bit data values and 1024 outstanding requests were allowed per node. Each non-local update required a 64-bit read request, 128-bit read reply (64-bit address and 64-bit data), and a 128-bit write (64-bit address and 64-bit data). The simulations were run allowing outstanding requests to be aggregated into larger packets and also without aggregation.

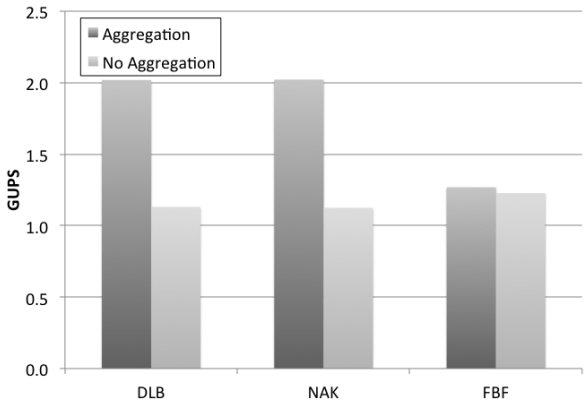


Figure14. DLB, NAK, and FBF in GUPS benchmark.

The LIONS networks performed poorly without aggregation due to the small average packet size and the tuning time discussed previously. The LIONS networks greatly outperformed the FBF when aggregation was allowed, and actually approached the theoretical maximum GUPS for the network configuration, as shown in Fig. 14. These results show that the tuning time for the AWGR based networks is critical to overall system performance, and that data aggregation can dramatically improve performance.

V. Conclusion

This paper discusses the architecture of LIONS together with its different loopback buffer schemes. A proof of concept demonstration of a 4x4 LIONS testbed is presented. A simulator is developed to model the LIONS architecture and was calibrated and verified by the experimental results. Although LIONS with DLB performs, in theory, the best in terms of latency and throughput, the AO-NACK architecture can eliminate the complex and costly loopback buffers while maintaining the same level of performance as DLB. The LIONS outperforms FBF in synthetic traffic, whereas it outperforms FBF in GUPS benchmark when aggregation is allowed. As the next step, we are currently developing the interface for the FPGA boards and 10 Gbps BM-CDRs for the testbed.

References

[1] G. E. Moore, "Cramming more components onto integrated circuits," *Electronics*, vol. 38, 1965.

[2] G. Amdahl, "Validity of the Single Processor Approach to Achieving Large-Scale Computing Capabilities," in *AFIPS Conference Proceedings*, 1967, pp. (30): 483–485.

[3] C. Minkenberg, F. Abel, P. Muller, R. Krishnamurthy, M. Gusat, P. Dill, I. Iliadis, R. Luijten, R. R. Hemenway, R. Grzybowski, and E. Schiattarella, "Designing a Crossbar Scheduler for HPC Applications," *IEEE Micro*, vol. 26, pp. 58 - 71, 2006.

[4] R. Hemenway, R. R. Grzybowski, C. Minkenberg, and R. Luijten, "Optical-packet-switched interconnect for supercomputer applications," *Journal of Optical Networks*, 2004.

[5] O. Liboiron-Ladouceur, A. Shacham, B. A. Small, B. G. Lee, H. Wang, C. P. Lai, A. Biberman, and K. Bergman, "The Data Vortex Optical Packet Switched Interconnection Network," *Journal of Lightwave Technology*, vol. 26, July 2008.

[6] X. Ye, P. Mejia, Y. Yin, R. Proietti, S. J. B. Yoo, and V. Akella, "DOS - A scalable Optical Switch for Datacenters," in *ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)*, 2010.

[7] C. Shang-Tse, A. Goel, N. McKeown, and B. Prabhakar, "Matching output queueing with a combined input/output-queued switch," *Selected Areas in Communications, IEEE Journal on*, vol. 17, pp. 1030-1039, 1999.

[8] Xiaohui Ye, Roberto Proietti, Yawei Yin, S. J. B. Yoo, and V. Akella, "Buffering and Flow Control in Optical Switches for High Performance Computing," *Journal of Optical Communications and Networking*, vol. 3, pp. A59-A72, August 2011.

[9] M. Hubner, K. Paulsson, and J. Becker, "Parallel and Flexible Multiprocessor System-On-Chip for Adaptive Automotive Applications based on Xilinx MicroBlaze Soft-Cores," in *Parallel and Distributed Processing Symposium, 2005. 19th IEEE International*, 2005.

[10] Roberto Proietti, Yawei Yin, Runxiang Yu, Xiaohui Ye, Christopher NItta, Venkatesh Akella, and S. J. B. Yoo, "All-optical Physical Layer NACK in AWGR-based Optical Interconnects," *IEEE Photonics Technology Letters*, vol. 24, pp. 410-412, March 2012.

[11] J. Kim, W. J. Dally, and D. Abts, "Flattened butterfly: a cost-efficient topology for high-radix networks," in *the 34th annual international symposium on Computer architecture*, 2007, pp. 126 - 137.

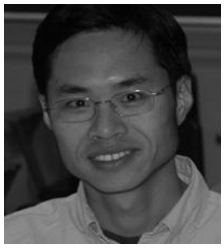
[12] Roberto Proietti, Christopher Nitta, Xiaohui Ye, Yawei Yin, Venkatesh Akella, and S. J. B. Yoo., "Performance of AWGR-based Optical Interconnects with Contention Resolution based on All-Optical NACKs," in *Optical Fiber Communications Conference (OFC)*, 2012.

Yawei Yin [M'11] received his B.S. degree in applied physics from National University of Defense Technology (NUDT), Changsha, China, in 2004 and Ph.D degree in Electrical Engineering from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 2009.

He is currently a postdoctoral researcher with the Next Generation Networking Systems Laboratory, University of California, Davis, His research interests include optical switching technologies and architectures for supercomputing and data center applications, as well as flexible bandwidth elastic optical networking algorithms, simulations and experiments.

Roberto Proietti [M'11] received his M.S. degree in Telecommunications Engineering from University of Pisa, Italy, in 2004 and the PhD in Electrical Engineering from Scuola Superiore Sant'Anna, Pisa, Italy in 2009

He is currently a postdoctoral researcher with the Next Generation Networking Systems Laboratory, University of California, Davis. His research interests include optical switching technologies and architectures for supercomputing and data center applications, high-spectrum efficiency coherent transmission systems and elastic optical networking.



Xiaohui Ye received his B.S.and M.S. degree in electrical engineering from Tsqinghua University, Beijing, China, in 2002 and 2005 respectively.

He is a Ph.D candidate with the Next Generation Networking Systems Laboratory, University of California, Davis. His research interests are in various topics related to heterogeneous networks, including optical switching technologies and architectures for supercomputing and data center applications.

Christopher J. Nitta received his PhD in Computer Science from University of California, Davis in 2011

He is a postdoctoral researcher and lecturer at University of California, Davis. His research interests include network-on-chip technologies, embedded system and RTOS design, and hybrid electric vehicle control.

Venkatesh Akella received his PhD in Computer Science from Univer- sity of Utah in 1992 and has been a professor of Electrical & Computer Engineering at University of California, Davis since 1992.

His current research encompasses various aspects of embedded systems and computer architecture with special emphasis on embedded software, hardware/software codesign and low power system design. He is member of ACM and received the NSF CAREER award.

S. J. B. Yoo [S'82, M'84, SM'97, F'07] received his B.S. degree in electrical engineering with distinction, his M.S. degree in electrical engineering, and his Ph.D. degree in electrical engineering with minor in physics, all from Stanford University, California, in 1984, 1986, and 1991, respectively.

He currently serves as a professor of electrical engineering at the University of California at Davis (UC Davis). His research at UC Davis includes optical switching devices, systems, and networking technologies for the future computing and communications. Prior to joining UC Davis in 1999, he was a senior research scientist at Bellcore, leading technical efforts in optical networking research and systems integration. He participated ATD/MONET testbed integration and a number of standardization activities including GR-2918-CORE, GR-2918-ILR, GR-1377-CORE, and GR-1377-ILR on dense WDM and OC-192 systems. He is a Fellow of the Optical Society of America (OSA), a Fellow of IEEE, and a recipient of the DARPA Award for Sustained Excellence (1997), the Bellcore CEO Award (1998), the Outstanding Mid-Career Research Award (UC Davis, 2004), and the Outstanding Senior Research Award (UC Davis, 2011).