Scalable Optical Interconnect Architecture Using AWGR-Based TONAK LION Switch With Limited Number of Wavelengths

Roberto Proietti, *Member, IEEE*, Yawei Yin, *Member, IEEE*, Runxiang Yu, Christopher J. Nitta, *Member, IEEE*, Venkatesh Akella, *Member, IEEE*, Christopher Mineo, and S. J. Ben Yoo, *Fellow, IEEE*

***Abstract*—This paper analyzes the scalability in arrayed waveguide grating router (AWGR)-based interconnect architectures and demonstrates active AWGR-based switching using a distributed control plane. First, the paper analyses an all-to-all single AWGR passive interconnection with N nodes and proposes a new architecture that overcomes the scalability limitation given by wavelength registration and crosstalk, by introducing multiples of smaller AWGRs ($W \times W$) operating on a fewer number of wavelengths ($W < N$). Second, this paper demonstrates active AWGR switching with a distributed control plane, to be used when the size of the interconnection network makes the all-to-all approach using passive AWGRs impractical. In particular, an active AWGR-based TONAK switch is introduced. TONAK combines an all-optical NACK technique, which removes the need for electrical buffers at the switch input/output ports, and a TOKEN technique, which enables a distributed all-optical arbiter to handle packet contention. The experimental validation and performance study of the AWGR-based TONAK switch is presented, demonstrating the feasibility of the TONAK solution and the high throughput and low average packet latency for an up to 75% offered load.**

***Index Terms*—AWGR, Datacenter Networking, Optical Interconnects, Optical Switches.**

I. Introduction

THE future high-performance computers (HPC) and/or Data Centers implemented with a vast number of parallel computing units will require balanced processing, memory and bisection bandwidth, according to Amdahl's Other Law [1], [2]. While the 2013 TOP500 supercomputers around the world are approaching tens of PetaFLOP/s performance [3], the bisection bandwidth required is also reaching the PetaByte/s scale, which is more than 46 times the average Internet traffic across the Globe today according to the Cisco Visual Networking Index's (VNI) forecast [4]. It is expected to be increasingly difficult to meet the high bandwidth density and low-latency communication requirements of these petascale computing systems using conventional electrical interconnects. Optical interconnects exploiting inherent wavelength division multiplexing (WDM) parallelism may possibly overcome those limitations.

Among all the proposed and existing optical interconnect architectures for HPC and datacenters, the arrayed waveguide grating router (AWGR) based solutions have attracted much attention due to WDM parallelism, dense interconnectivity and unique wavelength routing capability. For example the architectures of low-latency optical interconnects network switch (LIONS) (previously named as DOS [5]), Petabit [6], and IRIS [7] are all based on AWGR and tunable wavelength converters (TWC).

In principle, the single passive AWGR-based all-to-all interconnection of N nodes in a star topology provides the densest communication pattern that can be imposed in a computer network, but it requires N wavelengths and N^2 transceivers in total, unrealistic when N is a very large number. The number of wavelengths needed for the interconnection can be reduced to W if we consider using $W \times W$ AWGRs in a "Skinny-CLOS" network topology, as explained in Section II. Meanwhile, if we replace W fixed wavelength lasers with one tunable laser (TL) capable of tuning between W wavelengths, the number of transceivers needed can be reduced to M ($M = N/W$) per node. In this paper, we assume a value for W up to 128. Note that, 32-port AWGR is commercially available and 64-port AWGR was demonstrated in 2003 [8]. Moreover, references [9]–[11] demonstrate 128, 256, and 400-port AWGs with 25 GHz spacing. Hence, based on the technology available today, we believe that a 128-port AWGR is a reasonable assumption. The fact that $N \times N$ AWGRs with port count greater than 32 are not commercially available is mainly due to the fact that there is currently no commercial demands for $N \times N$ AWGRs. An alternative approach is the technique recently published in [12], where the authors demonstrated that a large port-count AWGR can be constructed by combining many smaller AWGRs, thus making it possible to scale the AWGR to 128 ports and above.

Commercial TLs can already cover the C-band with 50 GHz channel spacing (tuning across ~90 wavelength channels) and reference [13] demonstrates an even wider tuning bandwidth. Therefore, if we consider using 8 tunable transceivers per node, the 128-port AWGRs can support 1024 nodes when deployed in

Manuscript received July 15, 2013; revised September 18, 2013; accepted October 7, 2013. Date of publication October 15, 2013; date of current version November 27, 2013. This work was supported in part by the Department of Defense under DARPA Inpho program Grant W911NF-12-1-0311.

R. Proietti, R. Yu, C. J. Nitta, V. Akella and S. J. B. Yoo are with the Department of Electrical and Computer Engineering, University of California, Davis, CA 95616 USA (e-mail: rproietti@ucdavis.edu; rxyu@ucdavis.edu; cjnitta@ucdavis.edu; akella@ucdavis.edu; sbyoo@ucdavis.edu).

Y. Yin was with the Department of Electrical and Computer Engineering, University of California, Davis, CA 95616 USA. He is now with NEC Laboratories America Inc., Princeton, NJ 08540 USA (e-mail: yaweiyin@ieee.org).

C. Mineo is with the Laboratory of Physical Sciences, College Park, MD 20740 USA (e-mail: camineo@lps.umd.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JLT.2013.2285883

a "Skinny-CLOS" network topology. However, the reduction in the number of transceivers leads to the need for an active control plane (CP) that can handle the contentions between packets or data flows heading simultaneously toward the same output port, where the number of receivers is insufficient to receive them simultaneously. It has been investigated by the authors that the bottleneck in the scalability of a single LION switch is not only the size and port-count of the AWGR itself, but also the complexity of the loopback buffer and the electrical centralized controller.

This paper first discusses the scalability of the passive AWGR based interconnect system and then proposes a new architecture, which relies on the Skinny-CLOS topology. Further, this paper discusses an active AWGR switch called the TONAK-LION switch. The TONAK-LION switch is an advanced version of the LION switch, combining the distributed all-optical token (AO-TOKEN) CP [14], [15] and the all-optical NACK (AO-NACK) architecture [16]. The TONAK-LION switch combines the advantages of the AO-TOKEN and AO-NACK technologies, so that scalable all-optical switching with a distributed CP becomes possible. The proof-of-concept demonstration of the TONAK-LION switch is presented, validating the feasibility of such technologies. The network performance of a single TONAK-LION switch with 128 ports is also validated through simulation results.

The remainder of the paper is organized as follows: Section II describes the scalable all-to-all interconnection architecture based on multiple AWGRs, focusing on overcoming the AWGR scalability limitations in terms of port-count and crosstalk. We introduce multiples of smaller passive AWGRs in a Skinny-CLOS network topology to support all-to-all communication. Section III investigates active AWGR switches with a distributed CP designed for a large number of interconnection nodes. Section III-A describes the different versions of the active AWGR switch, namely the DOS, LIONS, NACK-LIONS and TOKEN-LIONS, and compares their performance. Section III-B proposes a new switch called TONAK-LIONS, which combines the advantages of TOKEN-LIONS and NACK-LIONS and makes the scalable all-optical switch with a distributed CP possible. In Section IV, we experimentally validate the TONAK-LIONS, using a proof-of-concept demonstration, and we then analyze the network performance of a single TONAK-LION switch through simulations. Section V concludes the paper.

II. AWGR All-to-All Interconnects using Limited Number of Wavelengths

All-to-all interconnection using AWGR as a passive component supports the densest possible communication pattern. The all-to-all pattern is the densest communication pattern since there is no bandwidth resources shared on any link (each link is dedicated to one Tx/Rx pair), and therefore no intermediate switching node is required in the all-to-all topology [17]. However, three limiting factors prevent such systems from being deployed in a large scale. First, the port count of a single AWGR is usually restricted by its size, the fabrication constraints and the inter-channel crosstalk. The difficulties arise mainly from the need for high precision control of the channel spacing during fabrication, as well as the need for accurate wavelength registration for all channels after fabrication [18]. Second, the number of wavelengths required in an all-to-all system grows linearly with the number of nodes, but the limited wavelength range leads to high-density channel spacing. Note that, a large number of wavelength channels in an AWGR will lead to significant increases in coherent (in-band) and incoherent (out-of-band) crosstalk [19], which will significantly impair the scalability of an AWGR based interconnection. Therefore, a more scalable architecture imposes the requirement for many smaller AWGRs with a fewer number of wavelengths in order to achieve higher interconnectivity that the original single AWGR system achieved. Fortunately, such an approach is feasible. Fig. 1 shows an example of N nodes interconnecting using W wavelengths and $W \times W$ AWGRs. The transmitters for each node consist of M $(M = N/W)$ groups of fixed wavelength lasers, each group containing W wavelengths and connecting to a separate AWGR input port.

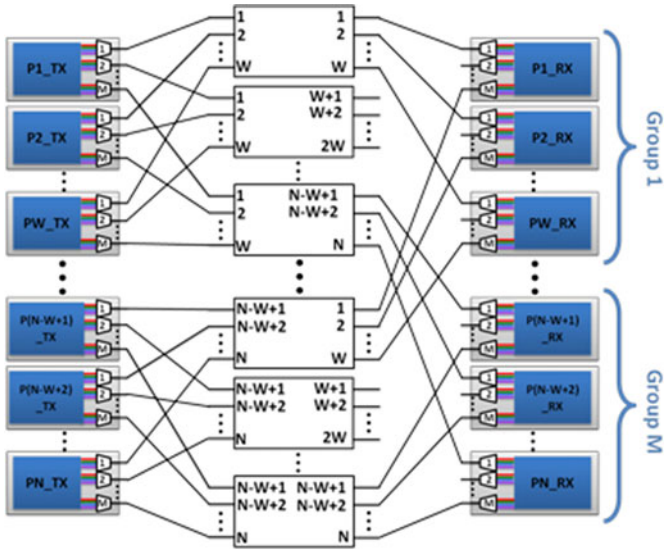

Fig. 1. An example of the scalable optical interconnect architecture using small AWGRS with limited number of wavelengths [20].

Note that the number of wavelengths used in the entire system is reduced by a factor of M, which is the same reduction ratio for the port count requirement on a single AWGR. However, the disadvantage is that the total number of AWGRs increases by a factor of M^2. Smaller AWGRs have far less stringent fabrication and wavelength registration requirements than larger AWGRs do, and they prevent the accumulation of optical crosstalk between a large number of channels.

The above illustrated network topology is named "Skinny-CLOS," since it borrows the idea of the CLOS network [20], which uses small switches to reach large interconnectivity, but it has only one stage of switches (as opposed to the three stages of switches in CLOS) and fewer connections than does the CLOS network. Table I summarizes the parameters needed to build the Skinny-CLOS network with N nodes and compares them with

TABLE I
THE PARAMETERS NEEDED FOR THE DIFFERENT CONFIGURATIONS TO ACHIEVE ALL-TO-ALL INTERCONNECTION

	Directly connect	*NxN* AWGR	Skinny CLOS
# of nodes	*N*	*N*	*N*
# of Tx/Rx	N^2	N^2	N^2
# of WLs	1	*N*	*W*
AWGR port cnt.	*n/a*	*N*	*W*
# of AWGRs	*n/a*	1	N^2/W^2
# of cross talk components	*n/a*	*N*-1	*W*-1
Total fibers	*N(N-1)*	*2N*	$2N^2/W$
Total I/O ports	*n/a*	*N*	N^2/W

other solutions, such as the directly connected all-to-all network using fiber patch cords and the single large AWGR solution. As discussed above, we will consider $W = 128$. Therefore, if we assume $M = 8$, the 128-port AWGRs can support 1024 nodes when deployed in a "Skinny-CLOS" network topology.

In the above, we have focused on using AWGRs as passive interconnect components and targeting an all-to-all system which does not require active optical switching. However, there is the third limiting factor, which is that the number of transceivers needed in the all-to-all network scales exponentially with the number of nodes supported in the network. In other words, every node in the network requires N-1 dedicated transceivers to send to and receive from the other nodes, which yields a total number of $N(N\text{-}1)$ transceivers in the entire network. The $O(N^2)$ is prohibitively large in a real HPC system with tens of thousands of nodes.

The number of transceivers per node can be reduced to M if we use a TL capable of tuning between W wavelengths to replace W different fixed wavelength lasers and, correspondingly, reduce the number of receivers to M. However, this reduction imposes contention between packets trying to reach the same output port since there is no longer a dedicated receiver for each wavelength. Consequently, the optical switches sitting in the middle should be able to handle the contention, meaning that we can no longer use the AWGRs only as passive components and must introduce some active components to arbitrate between the contended packets and grant only a certain number of them. The number of simultaneously granted packets should equal the number of the parallel receivers at each node, and the rest of the contended packets must be buffered or retransmitted.

The LIONS architectures [5], [21] previously proposed by our research team have exploited the use of AWGR as the passive switching fabric as well as the active switching components for handling the arbitration and buffering issues. Section III will briefly review the previous LIONS architectures and introduce a new AWGR based active optical switch, called the TONAK-LION switch.

III. THE AWGR-BASED ACTIVE OPTICAL SWITCH

A. The LION Switch

The low-latency interconnect optical network (LION) switch (previously named DOS) [5], [21] consists of an N port AWGR, one TWC at each input port, an FPGA-based electrical CP, electrical loopback buffers, label extractors, and fiber delay lines. Between the switch and each end-node, an optical channel adapter (OCA) serves as the media interface. The LION switch uses a *forward-and-store* strategy for packets, 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 buffers play an important role in contention resolution for the LION switch. The three proposed loopback buffer architectures in LIONS are referred to as the shared loopback buffer (SLB), the distributed loopback buffer (DLB) and the mixed loopback buffer (MLB). The detailed introduction and performance evaluations of the three buffer architectures are explained in [22]. In short, the SLB requires the most memory I/O bandwidth, while occupying only one additional AWGR input/output port. In contrast, the DLB requires the least memory I/O bandwidth, but it requires more tunable transmitters, and the size of the AWGR must be doubled. While the SLB and the DLB represent the two extremes, the MLB provides a tradeoff between them.

Despite the flexibility and maturity of the electrical loopback buffers, the memory read/write speed becomes a bottleneck. In addition, the electrical loopback buffer requires a large amount of O/E/O conversion, which is power and cost inefficient. We then proposed and implemented the all-optical negative acknowledgement (AO-NACK) technology [16] to eliminate the need for the electrical loopback buffer and improve the line-rate scalability of the LION switch. The AO-NACK technology in the buffer-less LION (NACK-LION) switch uses an optical circulator (OC) at the loopback port of the original LIONS, and exploiting the duplex nature of the AWGR, all of the "dropped" packets from the input ports which failed to win the contention are simultaneously directed to the loopback port by the TWCs and reflected back to their input ports. After being reflected back, the backward travelling (counter propagating) packets are separated from the forward travelling packets at the input ports of the AWGR by OCs and then transmitted back to the hosts. The hosts treat the reflected-back packets as negative acknowledgements (NACKs) and retransmit the corresponding packets in accordance with certain retransmission policies. Compared with the original LIONS, the NACK-LION switch also handles the contended packets through retransmission. The difference is that LIONS retransmits the packets at the site of the switch itself, using the electrical loopback buffer, while the NACK-LIONS retransmits the packets at the site of the hosts. Note that, the retransmission at the host-site leads to more latency. However, the removal of the electrical loopback buffer enables a higher line rate for the packets. In addition, the loopback buffer will also introduce a long queuing delay when the traffic load is high. Our study in [5] shows that, given the short host-switch distances in datacenters, the NACK-LIONS can achieve nearly the same performance in terms of average throughput and latency as can the LIONS with DLB (see Fig. 2). Note that the DLB architecture in LIONS guarantees the best throughput among the three loopback buffer architectures for LIONS, far exceeding the throughput of electronic switch architectures, including the flattened butterfly architecture [21].

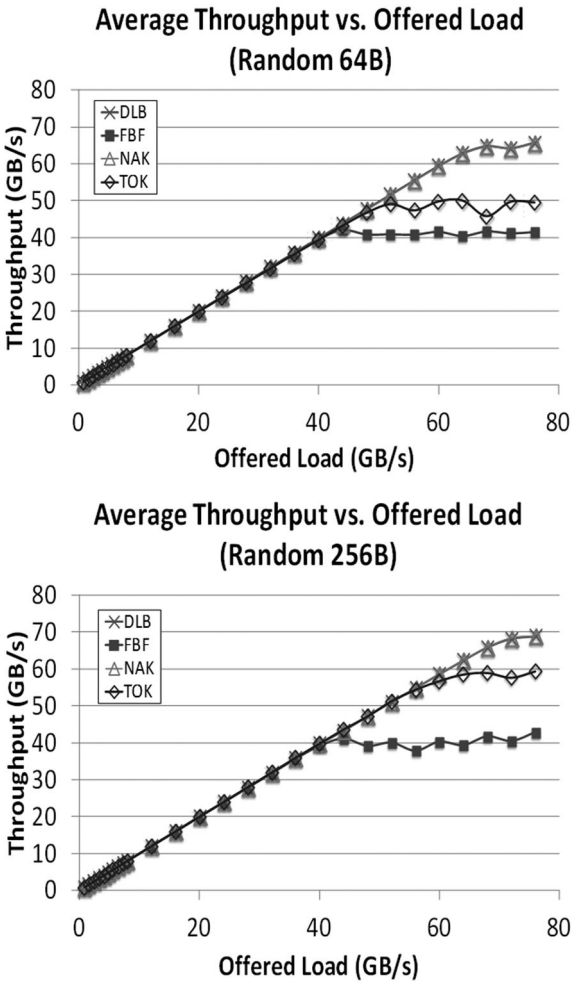

Fig. 2. Performance comparison of DOS with DLB, Electrical switches with Flattened Butterfly (FBF) topology, the LIONS with AO-NACK technology and the LIONS with AO-TOKEN technology, using uniform random traffic and average packet length of 64 bytes (top) and 256 bytes (bottom).

The centralized electrical CP is another major limitation to the scalability of LIONS, since it limits the port count and increases latency. In fact, the maximum number of I/O resources of currently available integrated chips [23] can pose an upper limit to the number of ports that a single CP can handle. Therefore, the all-optical token (AO-TOKEN) technique was introduced in [14] to eliminate the need for a centralized CP by exploiting the saturation effect in the reflective semiconductor optical amplifier (RSOA) [24]. The basic idea is the use of RSOA as the mutual exclusion (mutex) type of arbiter at each output port of the AWGR and the transmission of a packet only after it applies and is granted by its corresponding arbiter. The major advantage of the TOKEN-LIONS technique is that it distributes the contention resolution in the CP without the requirement of a global coordination scheme. This key advantage makes the optical switches scalable. However, we should notice that, since the packets cannot be buffered at the input port of the switch, the delay caused by the wait for the token response can negatively affect the switch performance. Our studies in [14] show how the host-switch distance and the ratio between it and the packet size can significantly impact the switch performance.

Fig. 2 shows a comparison of the switching performance of the prevailing electrical switching architectures using the flattened butterfly (FBF) topology, the LIONS with DLB, the NACK-LIONS and the TOKEN-LIONS. The simulation assumed 64 computing nodes, each node generating packets according to a Bernoulli process. The destination address of the packets follows a uniform random distribution. We assumed 10 Gb/s line rate, 50 ns light propagation time (10 m fiber) from the hosts to the switch, 8 ns wavelength tuning time [13] and 64 ns burst mode RX delay [25]. A small buffer size (40 packets) was assumed in the Tx, Rx, and loopback buffers. As Fig. 2 shows, 256 bytes and 64 bytes average packet sizes were generated, respectively. The LIONS with DLB performs the best, and the performance of the NACK-LIONS is very close to that of the DLB-LION switch. The TOKEN-LIONS performs well when the packet length is large, but starts to saturate early when small packets are used. The link efficiency is reduced because the round-trip-time (RTT) involved in the wait for the token to be granted adds more overhead.

The need to improve the performance of TOKEN-LIONS architecture and eliminate the limitation caused by the waiting time between a TOKEN request and packet transmission led to the design of a new architecture, named TONAK, that can guarantee performance as good as that of AO-NACK while still guaranteeing the advantages of the TOKEN technique.

B. The TONAK-LION Switch

Fig. 3 shows the TONAK-LION switch architecture. Fig. 3, inset *ii)*, shows the optical transmitter, which uses one TL to generate both packets and the corresponding token requests (TRs). The TRs are always generated earlier than are the packets, and the offset time between the two is determined by the RTT time from the TONAK line-card to the RSOA. Inset *i*) shows the line-card that is placed in front of each AWGR input port. This line-card is the key component combining the AO-NACK and AO-TOKEN techniques, since it controls the TR signals and the transmission/reflection of the data packet after its token is granted/denied. The following explains the TONAK's working principle in detail, using the timing diagram illustration in Fig. 4.

When H_1 wants to send a packet to H_N, H_1 will first tune its fast TL to λ_{1N} (the wavelength to reach output N from input 1 in accordance with the AWGR routing table) to generate a TR $\boldsymbol{A}$ which reaches the CP AWGR input port one at $t = t_1$. $\boldsymbol{A}$ is then routed to output N, where it enters in an RSOA to request the token after going through a 1:k optical demultiplexer. We assume k RSOAs are placed at each CP AWGR output port in order to exploit the wavelength parallelism and reduce the contention probability [21]. Note that each node has k RXs to receive up to k simultaneous packets. So, if there are not more than k simultaneous requests, with each request coming from a different contention group [21], there won't be any contentions since each request will go to a different RSOA. However, if there is more than one request coming from the same contention group, there will be more than one request going to the same RSOA, indeed causing contention.

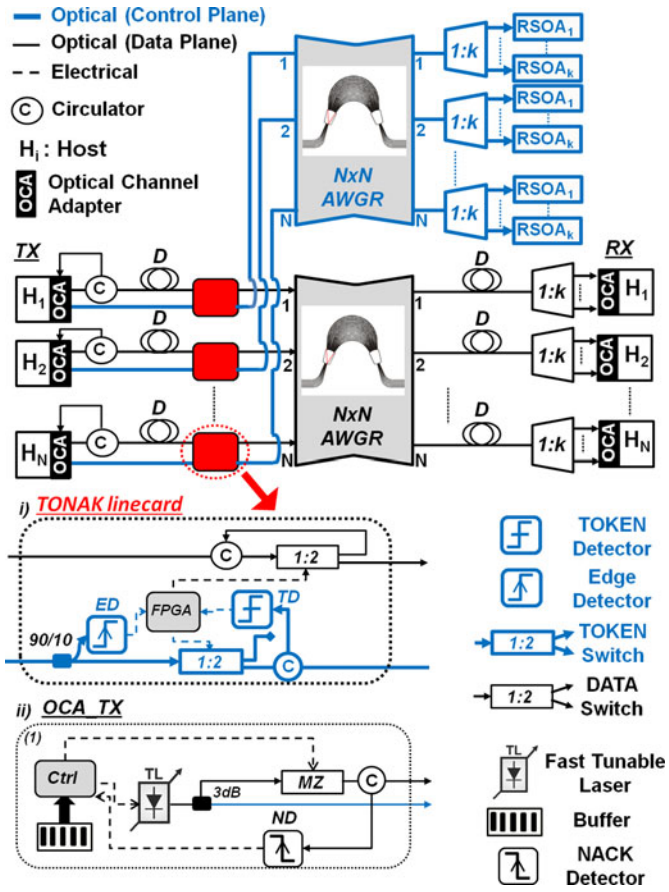

Fig. 3. Distributed TONAK architecture. D is the distance between hosts (Hi) and AWGRs input ports. AWGR: Arrayed Waveguide Grating Router; OCA: Optical Channel Adapter. Inset *i)*: line-card with Token Detector (TD), Token Request Edge Detector (ED), Circulators (C) and Controller. Each control plane AWGR output port connects to an optical demultiplexer and k Reflective SOAs. Each data plane AWGR output port connects to an optical demultiplexer and k burst-mode RXs. Inset *ii)*: host TX interface with ingress buffer queue (I-Q), fast tunable transmitter (TL), NACK detector, and Mach–Zehnder Modulator.

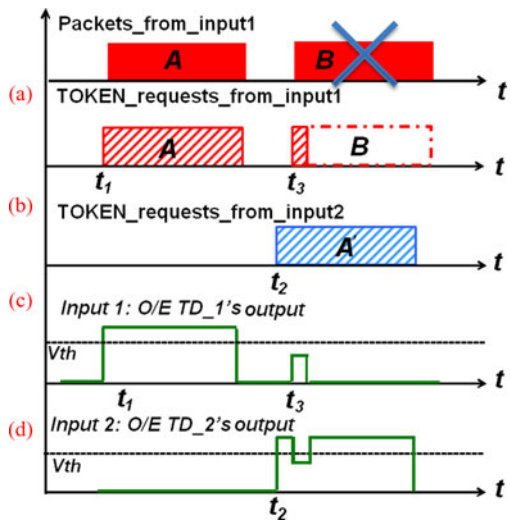

Fig. 4. Timing diagram explaining how the all-optical control plane can detect contention.

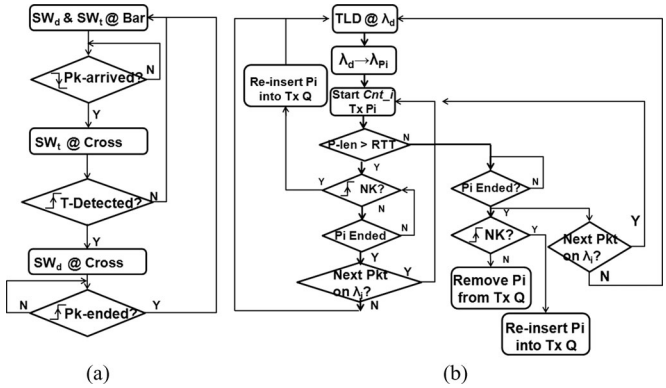

Fig. 5. Flow chart of the (a) TONAK line-card and (b) the OCA Tx in the TONAK-LION switch. SWd: data switch; SWt: token switch; Pk: packet; T: token; λd: default lambda; Cnt: counter; Tx: transmit; Pi: packet i, P_len: packet length; NK: NACK, Tx Q: the buffering queue at the transmitter.

The RSOA amplifies TR ***A*** and reflects it back to the AWGR input port, where TR A is extracted by an OC placed on the token path (inset i)) and converted in the electrical domain by a token detector (TD). The TD (simply an O/E converter followed by a threshold comparator) generates an electrical signal with a voltage V_{p1} proportional to the optical power (P_{TO1}) of the reflected TR. V_{p1} being greater than V_{th} means that output N is available. Then the controller (FPGA or ASIC) sets the 1:2 $LiNbO_3$ switch (switching time $<$1 ns) in the data path to the *cross* state so that packet A can be switched on-the-fly to the desired output port of the data-plane AWGR. Note that the TR stays active for the entire packet transmission time to hold the token and to prevent collision. The 1:2 $LiNbO_3$ switch in the token path is set to the *cross* state whenever the edge detector (ED) senses an incoming TR.

The scenario described above represents the case when a packet is not experiencing contention. However, as shown in Fig. 4, H_2 generates a TR and packet A' directed to the same output N. This request reaches the CP AWGR input port one at $t = t_2$. Then at $t = t_3$, when the transmission of packet A' has not yet been completed, another TR coming from H_1 and being directed to output N arrives. This time, the RSOA at output N is already saturated with the TR A' at λ_{2N}. Therefore, the RSOA amplifies and reflects back the new TR B at λ_{1N}, with a lower power. Given that TR B reaches the TD with optical power $\mathrm{P_{TO3}}$, the TD will generate an electrical signal with V_{p3}. Due to the gain saturation effect [14], $\mathrm{P_{TO3}}$ will be $\approx$ Psat/2 and V_{p3} will be $\approx V_{p1}/2$, where Psat is the saturation output power of the RSOA. With V_{th} set between V_{p1} and $V_{p1}/2$, the controller can recognize that the token for output N is not available. Upon the failed token application, the controller then sets the 1:2 switch in the data path to the *bar* state. In this case, the incoming packet B is blocked and sent back to the Tx, where it is extracted by an OC and acts as AO-NACK. The controller also sets the 1:2 switch for the token path to the *bar* state, which immediately blocks the denied TR B.

The complete workflow of an OCA Tx and a TONAK line-card is shown in the flow chart in Fig. 5. Note that, at the

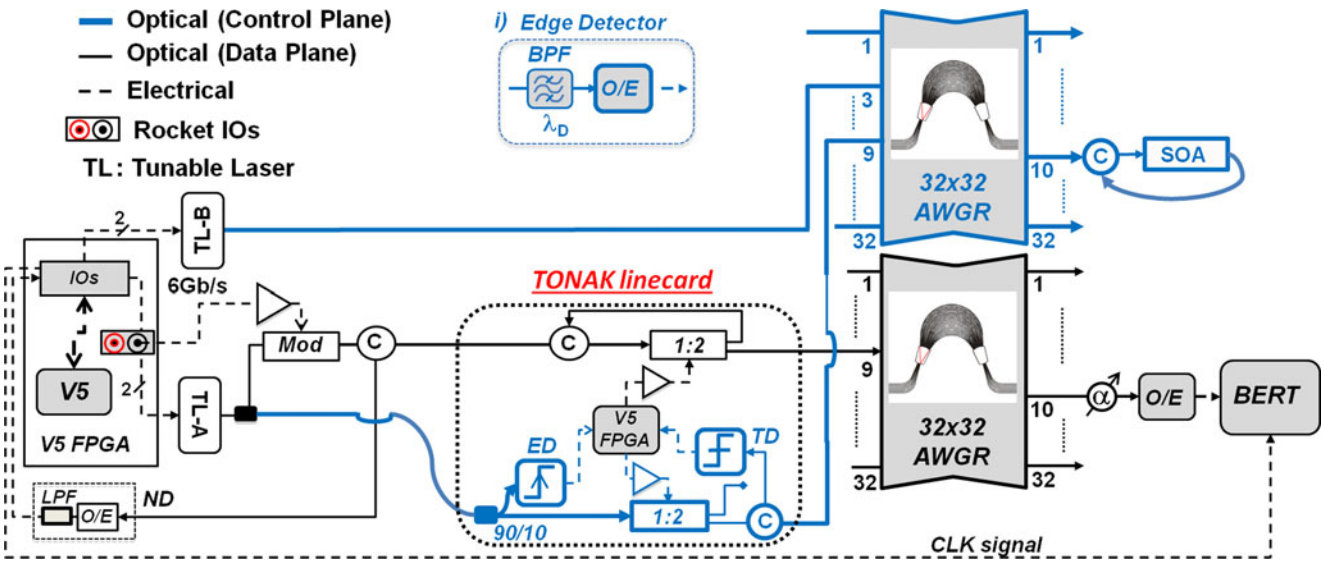

Fig. 6. Experimental setup of the TONAK-LION switch testbed. V5 FPGA: Virtex 5 Field Programmable Gate Array; ND: Nack Detector; LPF: Low Pass Filter; Mod: Modulator; C: Optical Circulator; ED: Edge Detector; TD: Token Detector; AWGR: Arrayed Waveguide Grating Router; SOA: Semiconductor Optical Amplifier; α:Variable Optical Attenuator; BERT: Bit Error Rate Tester.

transmitters, the flow is divided into two sub-flows based on the ratio between the RTT time and the packet length [16].

Like in AO-TOKEN, TONAK does not require a centralized CP and the acquisition of the token is handled in a fully distributed fashion. However, the main differences between TONAK-LIONS and TOKEN-LIONS are that (a) the TD is placed at the input port of the switch in TONAK, as opposed to the TD being placed on the distant Tx side; (b) the TONAK uses the AO-NACK technique to notify the senders of any packets experiencing contention, as opposed to holding the packet at the Tx until it wins the contention. In this way, the offset time between the TR and its packet can be dramatically reduced, mitigating a performance bottleneck in the TOKEN-LION switch, as described above.

IV. Experimental Validation and Performance Study of the TONAK Switch

A. Experiment Validation

Fig. 6 illustrates the testbed used for the proof-of-concept experimental demonstration of TONAK architecture. A Virtex5 FPGA evaluation board generates TRs and related packets. The packets are generated through a rocket IO. GTX interface, which limits the line-rate used in this experiment to 6 Gb/s. Standard user IOs pins are used to control and tune two fast TLs [13], [26], i.e., TL-A and TL-B. The FPGA tunes TL-A (TL-B) sending two control signals named txa_tld_bit (txb_tld_bit) and txa_tld_en (txb_tld_en). TL-A is the laser for transmitter A (TX-A), which connects to the TONAK switch input port 9 through a TONAK linecard. TX-A generates a sequence of TRs and related packets, as explained in detail later. TL-A connects to a 3-dB power splitter. One splitter output connects to a Mach Zehnder (MZ) modulator for data packet modulation (a 10 GHz electrical amplifier drives the modulator with the data generated by rocket IO interface). The modulator connects to the data input of the TONAK linecard through an OC, which extracts the counter-propagating AO-NACK messages, as explained in the previous section. The AO-NACK messages are then detected by a NACK detector (ND) connected to one FPGA IO pin. The ND in this experiment is implemented with a simple 1.25 GHz O/E converter (with limiting amplifier) and a 400 MHz low-pass filter. The second splitter output (blue color) connects directly to the TOKEN input of the TONAK linecard. The TONAK line-card has two inputs, as explained above, and two outputs. The DATA output (black) connects to the DATA plane AWGR, while the TOKEN output (blue) connects to the TOKEN plane AWGR. The DATA path (black) contains an OC followed by a 1:2 MZ switch. Its default position is in *bar* state (output connected to OC). If the TOKEN response coming from the distributed CP is positive (TOKEN detector output is "1"), a V5 FPGA changes the MZ switch to *cross* state (output connected to AWGR DATA plane) to let the incoming packet going to the AWGR DATA plane input and reach the desired output. In case the response to a TOKEN request is negative (TOKEN detector output is "0") the incoming packet is reflected back to the TX, where it gets detected by the ND.

TABLE II
Wavelength Values Used in the Experiment

	$\lambda_{DEFAULT}$ (tx_tld_bit)	λ_{SIGNAL} (tx_tld_bit)
TL-A	1547.85 nm (0)	1550.1 nm (1)
TL-B	1547.35 nm (1)	1547.6 nm (0)

The TONAK linecard TOKEN path (blue) contains a 90/10 splitter, a 1:2 MZ switch and a circulator. Default state for the MZ switch is *bar* (idle output). The power splitter taps 10% of the optical power of an incoming TR to feed an ED. Since a TOKEN request is initiated with a change of TL wavelength from $\lambda_{DEFAULT}$ to λ_{SIGNAL} (see Table II), the ED is composed by a passband filter centered at $\lambda_{DEFAULT}$ and an O/E converter

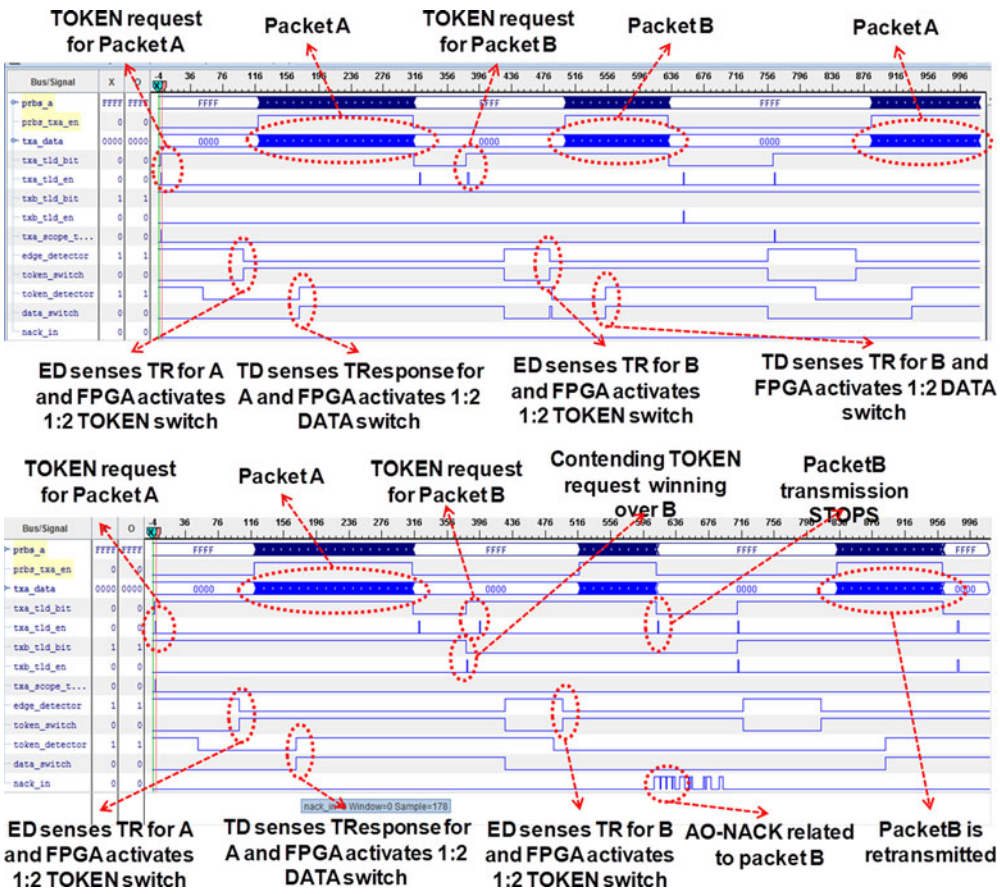

Fig. 7. ChipScope experimental timing diagram demonstrating the TONAK technique in case of contention (bottom) and no contention (top).

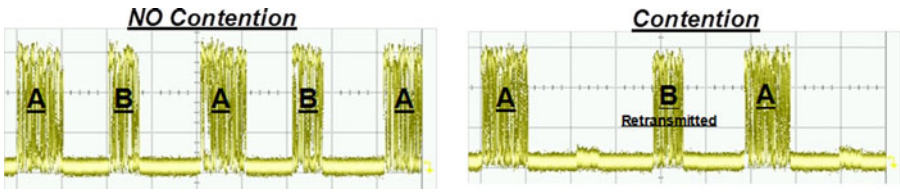

Fig. 8. Experimental oscilloscope traces for packets at AWGR output 10 in case of NO contention (left) and contention (right).

(Fig. 6 inset *i*)). When the ED senses an incoming TOKEN request, the V5 FPGA, triggered on the falling edge of the ED output, changes the 1:2 MZ switch to *cross*-state to let the request reaching the TOKEN-plane AWGR input and then the SOA at the desired AWGR output. If the token response is positive, the 1:2 MZ switch stays in *cross*-state for the entire packet duration. In case of a negative token response (desired TONAK switch output is not available), the FPGA change the MZ switch state back to *bar* state.

Two 50 GHz-spacing 32×32 AWGR with uniform insertion loss of 8 dB and cyclic frequency characteristic (ULCF AWGR [27]) represent the core of the TONAK switch architecture. The bottom AWGR (black) acts as the data plane switch fabric, while the top AWGR (blue) implements, together with a SOA, the distributed all-optical TOKEN-based CP described above. Because an RSOA was not available, we emulated the RSOA function with a SOA and an OC.

In this proof-of-concept demonstration TX-A generates endlessly two packets, A and B (Fig. 7), directed to the TONAK switch output 10. The TL-B, connected to input 3 of the TOKEN plane AWGR, generates only a periodic contending TR that causes contention for packet B. As a result of this contention event, packet B is reflected back to TX-A and retransmitted at a later time, as shown in Fig. 7 and Fig. 8. The SOA is placed at output 10 of the TOKEN plane AWGR.

Table II shows the wavelength values used in the experiment and related values for the control bit signals (*txa_tld_bit* and *txb_tld_bit*). The wavelength values named as $\lambda_{\mathrm{SIGNAL}}$ are determined by the AWGR routing table. In particular, 1550.1 nm and 1547.6 nm are the wavelength values to reach AWGR output 10 from AWGR inputs 9 and 3, respectively. The values named as $\lambda_{\mathrm{DEFAULT}}$ do not belong to the AWGR grid so that the optical power from TL-A and TL-B is blocked when no packets have to be transmitted. This is important in an actual implementation to avoid crosstalk at the switch outputs.

The following describes the experiment. Fig. 7 illustrates two timing diagrams showing traces acquired with Xilinx ISE ChipScope tool, which allows capturing the electrical signals at the different FPGA IOs during the experiment. The top timing diagram is for a case in which no contending TRs are generated by TLB. The bottom timing diagram shows the case when the CP detects contention for packets B, which are then retransmitted. Note that the numbers at the top of each timing diagram represent the time evolution in clock cycle (2.67 ns/clk in this experiment). Clock cycle "0" corresponds to the trigger event given by the enable pulse for TL-A, which determines the beginning of a TOKEN request for packet A. So, when the FPGA generates a pulse enable signal on "*txa_tld_en*" IO pin and sets the "*txa_tld_bit*" pin output to "1", TL-A tunes its wavelength from $\lambda_{\mathrm{DEFAULT}}$ to $\lambda_{\mathrm{SIGNAL}}$ (see Table II). After a certain amount of clock cycles, the TOKEN request A reaches the ED that triggers the FPGA (edge_detector signal on Chip_scope goes "low") in TONAK linecard, which then sets the MZ TOKEN switch in *cross* state (ChipScope *token_switch* signal goes "high"). In this way, the TOKEN request can reach the SOA at output 10 of the TOKEN plane AWGR. After approximately 80 clock cycles (equivalent to the round-trip time for the TOKEN request to reach the SOA, being reflected back and reach the TOKEN detector), the TOKEN detector senses the reflected TOKEN request. Since the SOA was not saturated (which means that the target TONAK switch output is available), the optical power is enough to trigger the TOKEN detector (Chip_scope *token_detector* signal is "high"). Then, FPGA sets the MZ DATA switch to *cross*-state (*data_switch* signal on Chip_scope goes "high"), allowing the incoming packet A to enter the AWGR data plane and being routed to the desired output port 10. Note that packet A transmission (*txa_data* on Chip_scope) starts with a certain delay compared to the related TOKEN request. This delay is to account for the latency in the TL board and the ED to TD round-trip time.

When transmission of packet A has been completed, FPGA sets *txa_tld_bit* at "0" and generates an enable pulse on *txa_tld_en* to return TL-A to $\lambda_{\mathrm{DEFAULT}}$. Transmission of packet B follows exactly the same process described above. The only difference is the length of packet B, which is 2/3 of packet A length. Both packet A and B contains a different portion of a PRBS 2^{15} -1.

Let us now analyze the case with contention illustrated in Fig. 7 (bottom). The contention happens for packet B (for packet A, the situation is exactly the same explained above). Note that, a few clock cycles before the generation of TOKEN request for packet B, *txb_tld_bit* is set at "0" and an enable pulse is generated on *txb_tld_en*. This means that TL-B tunes from its default position to 1547.6 nm, the wavelength values to reach and saturate the SOA. This time, the TOKEN request for packet B finds the SOA already saturated and reaches the TD with an optical power value too low to trigger the TD. The FPGA, not seeing the *token_detector* signal going "high" when expected, understands that the TOKEN request for packet B is not successful (desired AWGR output is not available). Then, the MZ data switch is left in default position (*bar* state) and the incoming packet B gets reflected back to TX-A, where it gets detected by the ND (see *nack_in* Chipscope signal going "high"). As response to the detection of an AO-NACK message, TX-A stops immediately transmission of packet B (which was still under transmission, brings TL-A back to its default wavelength, and schedules retransmission of packet B at a later time (around clock cycle number 886). This time, retransmission of packet B is successful. Fig. 8 shows oscilloscope traces for packets at AWGR output 10 for the case without (left figure) and with (right figure) contention.

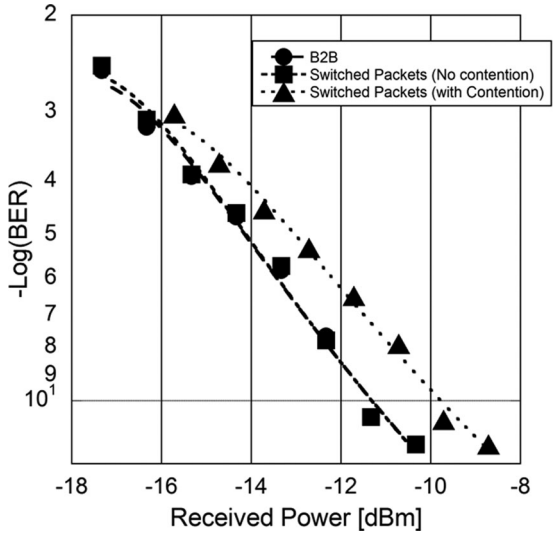

Fig. 9. BER measurements: Back to Back (circle); Switched packets at AWGR output 10 without contention (squares) and with contention for packets B (triangles).

Fig. 9 shows BER measurements for the packets at AWGR output 10 for the contention-less (squares) and contention (triangles) scenarios described above. In both cases the BER reaches error-free condition. To account for the different duty cycles of the signals (see Fig. 8), BER measurements have been plotted as function of the peak received power. There is some penalty associated with the switched packets in case of contention and retransmission of packet B. This penalty is given by the limited extinction ratio (<20 dB) of the 1:2 MZ switch used in the experiment (there is some optical power in between packet A and retransmitted packet B - see Fig. 8). Note that devices with higher extinction ratio (>30 dB) are available (http://www.eospace.com/switches.htm). Actually, the finite ER of the 1×2 switches can cause out-of-band crosstalk. It is then important to maximize the ER at the 1:2 MZ switch output. The worst case for this type of out-of-band crosstalk is when N-1 inputs are trying to send data to the same output simultaneously. Assuming k RSOA per output port, k requests would be granted and N-1-k packets would be rejected, causing N-1-k sources of out-of-band crosstalk. Since each node has a 1:k demux and k receivers, each receiver would only see (N-1-k)/k crosstalk terms. For $N = 128$ and $k = 4$, this translates in about 30 crosstalk terms (a factor of 14.7 dB). So, in the worst case, the signal to crosstalk ratio would be 30-14.7 = 15.3 dB. This value, according to the paper [28] gives negligible penalty.

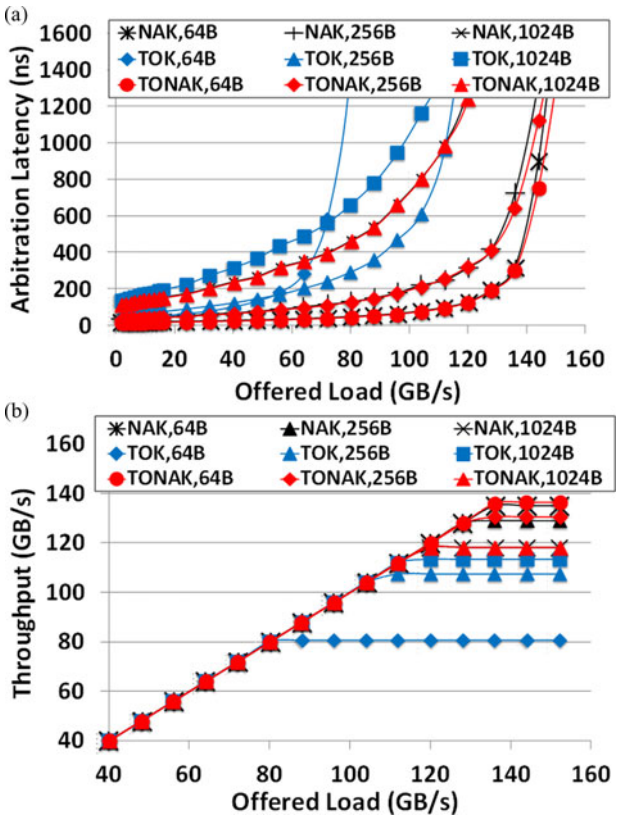

Fig. 10. (a) Arbitration latency as function of the offered load for uniform random traffic distribution. (b) Throughput as function of the offered load for uniform random traffic distribution.

B. Performance Study of the TONAK Switch Architecture

We developed a clock-cycle-accurate architecture level simulator and simulated a TONAK switch with 128 ports. TONAK performance is compared against AO-TOKEN and the centralized AO-NACK architecture. The number of receivers per output port (k) was chosen to be 4. We simulated both synthetic uniform random traffic and GUPS (Giga-Updates per Second). Fig. 10 (a) and (b), respectively, show the performance of the three architectures in terms of average arbitration latency and throughput as a function of the offered load. The host-switch distance was fixed to 4 meters, and average packet sizes of 64B, 256B, and 1024B were simulated. Line-rate was 10 Gb/s. TONAK significantly outperforms AO-TOKEN (TOK) for the reasons mentioned above.

TONAK performance is also slightly better than AO-NACK (NAK) architecture because TONAK does not require a guard-time (due to TL tuning time) between consecutive packets with the same destination.

Fig. 11 shows results for GUPS benchmarking, which is of particular interest in high performance computation. Traffic in GUPS is typical of in-memory database applications that implement transactional 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 simulated a 64-bit address space distributed across 128 nodes. Each update was applied to 64-bit data values and each node was allowed up to 1024 outstanding requests.

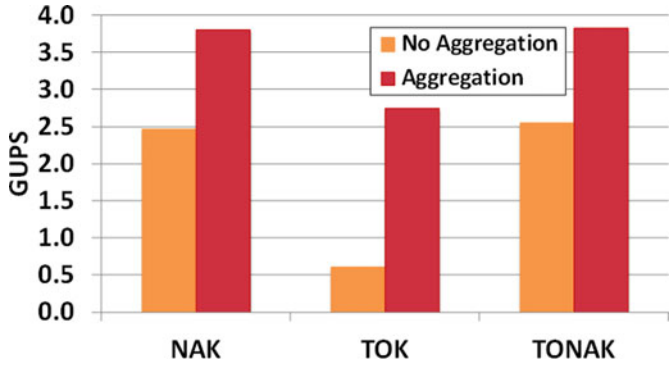

Fig. 11. Giga updates per seconds (GUPS) benchmarking results.

The results shown in Fig. 11 are for both aggregation of requests and replies into larger packets and for requests/replies being sent independently.

V. Conclusion

This paper investigated the scalability issues in the AWGR based interconnect architectures and pursued the active AWGR switch architecture with distributed CP. We started with the all-to-all interconnection using one AWGR, N wavelength and N^2 transceivers to interconnect N nodes, and analyzed its scalability limitations in terms of crosstalk, fabrication and wavelength registration. These limitations can be overcome by using W wavelengths and $W \times W$ AWGRs in the "Skinny-CLOS" network topology at the price of using more (N^2/W^2) of such small AWGRs. However, the non-scalable N^2 transceiver problem cannot be overcome easily. When the number of transceivers is reduced, the network has to introduce active switching components to handle the contention between packets. The AWGR-based active optical switches, named as DOS/LIONS, are reviewed. DOS/LIONS still have the scalability limitation in the electrical CP and loopback buffers. The most advanced all optical versions of LIONS, such as the AO-TOKEN and AO-NACK technique can remove the limitations in the electrical CP and loopback buffers, respectively. In order to combine the benefits of both AO-TOKEN and AO-NACK and completely build an all-optical, distributed and scalable switch, the TONAK architecture is proposed and implemented. Through experimental demonstration, we validated the working principle and feasibility of the switch and, through simulation based benchmarking experiments, we studied the performance of the TONAK switch and compared it with the previous TOKEN and NACK switches. The GUPS bench simulation results show that the TONAK switch performs at similar level as the NACK-LIONS switch, and significantly outperforms the TOKEN-LIONS switch. The main advantage given by the TONAK architecture against the NACK architecture is that the TONAK switch has a distributed CP which eliminates the need of TWCs at the switch input ports. This advantage is fourfold. First, the TONAK architecture is compatible now with advanced modulation formats, which could be used to increase the switch line-rate beyond the

limitation given by the AWGR channel bandwidth. Second, since the TL element becomes the TX laser, the total number of lasers in the system is reduced. Third, consecutive packets with the same destination address can be transmitted without the minimum guard time required by the previous architecture with centralized CP (minimum guard time is given by the laser tuning time). Lastly, the distributed architecture removes the complexity and I/O limitations given by the electrical wiring needed between the centralized CP and the TLs.

REFERENCES

[1] D. Cohen, F. Petrini, M. D. Day, M. Ben-Yehuda, S. W. Hunter, and U. Cummings, "Applying Amdahl's other law to the data center," *IBM J. Res. Dev.*, vol. 53, pp. 5:1–5:12, 2009.

[2] G. Bell, J. Gray, and A. Szalay, "Petascale computational systems," *Computer*, vol. 39, pp. 110–112, 2006.

[3] (2013, Jul. 14). TOP 500 SUPERCOMPUTER. [Online]. Available: http://www.top500.org/lists/2013/06/

[4] CISCO (2013, Jul. 4). Cisco Visual Networking Index: Forecast and Methodology, 2012–2017. [Online]. Available: http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/white_paper_c11-481360.pdf

[5] Y. Yin, R. Proietti, X. Ye, C. Nitta, V. Akella, and S. Yoo, "LIONS: An AWGR-based low-latency optical switch for high performance computing and data centers," *IEEE J. Sel. Topics Quantum Electron.*, vol. 19, no. 2, art. no. 3600409, Mar./Apr. 2012.

[6] K. Xi, Y.-H. Kao, M. Yang, and H. J. Chao, "Petabit optical switch for data center networks," Polytechnic Ins., New York Univ., New York Univ., New York, NY, USA, Tech. Rep. 2010.

[7] J. Gripp, J. E. Simsarian, J. D. LeGrange, P. Bernasconi, and D. T. Neilson, "Photonic terabit routers: The IRIS project," presented at the Opt. Fiber Commun. Conf., San Diego, CA, USA, 2010.

[8] S. Kamei, M. Ishii, M. Itoh, T. Shibata, Y. Inoue, and T. Kitagawa, "64 × 64-channel uniform-loss and cyclic-frequency arrayed-waveguide grating router module," *Electron. Lett.*, vol. 39, pp. 83–84, 2003.

[9] K. Okamoto, K. Syuto, H. Takahashi, and Y. Ohmori, "Fabrication of 128-channel arrayed-waveguide grating multiplexer with 25 GHz channel spacing," *Electron. Lett.*, vol. 32, pp. 1474–1476, 1996.

[10] Y. Hida, Y. Hibino, M. Itoh, A. Sugita, A. Himeno, and Y. Ohmori, "Fabrication of low-loss and polarisation-insensitive 256 channel arrayed-waveguide grating with 25 GHz spacing using 1.5% Δ waveguides," *Electron. Lett.*, vol. 36, pp. 820–821, 2000.

[11] Y. Hida, Y. Hibino, T. Kitoh, Y. Inoue, M. Itoh, T. Shibata, A. Sugita, and A. Himeno, "400-channel arrayed-waveguide grating with 25 GHz spacing using 1.5% Δ waveguides on 6-inch Si wafer," *Electron. Lett.*, vol. 37, pp. 576–577, 2001.

[12] T. Niwa, H. Hasegawa, K. Sato, T. Watanabe, and H. Takahashi, "Large port count wavelength routing optical switch consisting of cascaded small-size cyclic arrayed waveguide gratings," *IEEE Photon. Technol. Lett.*, vol. 24, no. 22, pp. 2027–2030, Nov. 2012.

[13] S. Matsuo and T. Segawa, "Microring-resonator-based widely tunable lasers," *IEEE J. Sel. Topics Quantum Electron.*, vol. 15, no. 3, pp. 545–554, May/Jun. 2009.

[14] R. Proietti, C. J. N. Y. Yin, R. Yu, S. J. B. Yoo, and V. Akella, "Scalable and distributed contention resolution in AWGR-based data center switches using RSOA-based optical mutual exclusion," *IEEE J. Sel. Topics Quantum Electron.*, vol. 19, no. 2, art. no. 3600111, Mar./Apr. 2012.

[15] R. Proietti, Y. Yin, R. Yu, C. Nitta, V. Akella, and S. J. B. Yoo, "An all-optical token technique enabling a fully-distributed control plane in AWGR-based optical interconnects," *J. Lightw. Technol.*, vol. 31, no. 3, pp. 414–422, Feb. 2013.

[16] R. Proietti, Y. Yin, R. Yu, X. Ye, C. Nitta, V. Akella, and S. J. B. Yoo, "All-optical physical layer NACK in AWGR-based optical interconnects," *IEEE Photon. Technol. Lett.*, vol. 24, no. 5, pp. 410–412, Mar. 2012.

[17] J. Bruck, H. Ching-Tien, S. Kipnis, E. Upfal, and D. Weathersby, "Efficient algorithms for all-to-all communications in multiport message-passing systems," *IEEE Trans. Parallel Distributed Syst.*, vol. 8, no. 11, pp. 1143–1156, Nov. 1997.

[18] J. Ing-Fa and L. San-Liang, "Simple approaches of wavelength registration for monolithically integrated DWDM laser arrays," *IEEE Photon. Technol. Lett.*, vol. 14, no. 12, pp. 1659–1661, Dec. 2002.

[19] E. L. Goldstein, L. Eskildsen, and A. F. Elrefaie, "Performance implications of component crosstalk in transparent lightwave networks," *IEEE Photon. Technol. Lett.*, vol. 6, no. 5, pp. 657–660, May 1994.

[20] Y. Yin, K. Wen, R. Proietti, C. J. Nitta, V. Akella, C. Mineo, and S. J. B. Yoo, "AWGR-based all-to-all optical interconnects using limited number of wavelengths," in *Proc. IEEE Opt. Interconnects Conf.*, 2013, pp. 47–48.

[21] C. Clos, "A study of non-blocking switching networks," *Bell Sys. Tech. J.*, vol. 32, pp. 406–424, 1953.

[22] X. Ye, P. Mejia, Y. Yin, R. Proietti, S. J. B. Yoo, and V. Akella, "DOS—A scalable optical switch for datacenters," in *Proc. ACM/IEEE Symp. Architectures Netw. Commun. Syst.*, 2010, pp. 1–12.

[23] Xiaohui Ye, Roberto Proietti, Yawei Yin, S. J. B. Yoo, and V. Akella, "Buffering and flow control in optical switches for high performance computing," *J. Opt. Commun. Netw.*, vol. 3, pp. A59–A72, Aug. 2011.

[24] F. Abel, C. Minkenberg, I. Iliadis, T. Engbersen, M. Gusat, F. Gramsamer, and R. P. Luijten, "Design issues in next-generation merchant switch fabrics," *IEEE-ACM Trans. Netw.*, vol. 15, no. 6, pp. 1603–1615, Dec. 2007.

[25] H. C. Shin, J. S. Lee, H. I. Kim, I. K. Yun, S. W. Kim, and S. T. Hwang, "Reflective semiconductor optical amplifier," Google Patents, Patent US8149503 B2, 2006.

[26] Y. Yin, R. Proietti, X. Ye, R. Yu, V. Akella, and S. J. B. Yoo, "Experimental demonstration of LIONS: A low latency optical switch for high performance computing," in *Proc. Int. Conf. Photon. Switching*, 2012, pp. 1–2.

[27] C. K. Chan, K. L. Sherman, and M. Zirngibl, "A fast 100-channel wavelength-tunable transmitter for optical packet switching," *IEEE Photon. Technol. Lett.*, vol. 13, no. 7, pp. 729–731, Jul. 2001.

[28] K. Okamoto, T. Hasegawa, O. Ishida, A. Himeno, and Y. Ohmori, "32 × 32 arrayed-waveguide grating multiplexer with uniform loss and cyclic frequency characteristics," *Electron. Lett.*, vol. 33, pp. 1865–1866, 1997.

[29] L. Buckman, L. Chen, and K. Lau, "Crosstalk penalty in all-optical distributed switching networks," *IEEE Photon. Technol. Lett.*, vol. 9, no. 2, pp. 250–252, Feb. 1997.

Roberto Proietti received the M.S. degree in telecommunications engineering from the University of Pisa, Pisa, Italy, in 2004, and the Ph.D. degree in electrical engineering from Scuola Superiore Sant'Anna, Pisa, Italy, in 2009.

He is currently a Project Scientist with the Next Generation Networking Systems Laboratory, University of California, Davis, CA, USA. 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.

Yawei Yin received the B.S. degree from the Department of Applied Physics, National University of Defense Technology, Changsha, China, in 2004, and the Ph.D. degree in electrical engineering from the Beijing University of Posts and Telecommunications, Beijing, China, in 2009.

He was a Postdoctoral Research Scientist in the Next Generation Networking Systems Laboratory, University of California, Davis, CA, USA, from 2009 to 2013. He is currently with NEC Laboratories America, Inc., Princeton, NJ, USA. His research interest includes low-latency and scalable optical interconnects for data centers and high performance computing and elastic optical networks.

Runxiang Yu received the B.Eng. degree in electrical engineering from Peking University, Beijing, China, in 2007. He is currently working toward the Ph.D. degree at the Department of Electrical and Computer Engineering, University of California, Davis, CA, USA.

His research focuses on advanced switching technologies and system level integration for next-generation optical networks.

Christopher J. Nitta received the Ph.D. degree in computer science from the University of California, Davis, CA, USA, in 2011. He is currently an Adjunct Professor of Computer Science 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 the Ph.D. in computer science from the University of Utah, Salt Lake City, UT, USA, in 1992.He is currently a Professor of Electrical and Computer Engineering at University of California, Davis. 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.

Christopher Mineo received the B.S. degree in electrical and computer engineering from Rutgers University, New Brunswick, NJ, USA, in 2003, and the M.S. and Ph.D. degrees in computer engineering from North Carolina State University, Raleigh, NC, USA, in 2005 and 2010, respectively. He has been part of the custom digital circuit design and chip-level timing teams in IBM's PowerPC development group, in Research Triangle Park, NC, USA. He has also worked in the optical proximity correction group at IBM in Hopewell Junction, NY, USA. He is currently a Researcher for the US Government at the Center for Exceptional Computing. His research interests include network-on-chip optimization, low-power circuit design, architectures for massively parallel systems, and optical inter-chip communication for high performance computing).

S. J. Ben Yoo (S'82–M'84–SM'97–F'07) received the B.S. degree in electrical engineering with distinction, the M.S. degree in electrical engineering, and the Ph.D. degree in electrical engineering with a minor in physics, all from Stanford University, Stanford, CA, in 1984, 1986, and 1991, respectively.

He currently serves as a Professor of Electrical Engineering at University of California at Davis (UC Davis). His research at UC Davis includes high-performance all-optical devices, systems, and networking technologies for 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, and is 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).