TONAK: A Distributed Low-latency and Scalable Optical Switch Architecture

Roberto Proietti*, Christopher J. Nitta, Yawei Yin, Venkatesh Akella, and S. J. B. Yoo*

Department of Electrical and Computer Engineering, University of California, Davis, California, 95616, USA, *rproietti@ucdavis.edu, *sbyoo@ucdavis.edu

Abstract *This paper proposes TONAK, an AWGR-based optical switch with distributed control plane. Simulations results for a 128-port switch show high throughput and low average packet latency for offered loads of up to 75%, while achieving an energy efficiency of ≈ 50pJ/bit.*

Introduction

The ever-growing demand for cloud-based services and high-performance computing is posing major challenges to the scalability and performance of large-scale datacenters and HPC systems [1]. The network performance, measured in terms of throughput, packet latency, as well as the total energy consumption per operation (J/OPs) have become critical metrics due to the limited port-count scalability of high-speed (>10Gb/s) electrical switches [2]. Optical technology can provide a viable solution toward high-throughput switches with both high port counts and low packet latency. Several research projects have already proposed architectures that replace legacy Electrical Packet Switching (EPS) with all-optical packet switching [3] or combination of EPS and slow optical circuit switching [4].

This paper focuses on Arrayed Waveguide Grating Router (AWGR)-based switches, which can exploit optical parallelism to implement output queuing switches (which are very costly and not scalable in electrical domain) that also strongly reduce contention at the switch.

Recently, Ref. [5] demonstrated an All-Optical Token (AO-TOKEN) technique that implements a fully distributed control plane by exploiting the saturation effect in Semiconductor Optical Amplifiers (SOAs) and the wavelength routing in the AWGR. The characteristics of these switches are highly desirable both from scalability and architectural standpoint. This paper proposes a new architecture, TONAK, which combines the token technique from [5] with the AO-NACK technique in [6]. Simulation results for a 128-port switch show that TONAK achieves arbitration latency below 300ns and high throughput for normalized offered load values up to 0.75, greatly outperforming AO-TOKEN. The paper concludes with an analysis on power consumption and scalability.

TONAK Architecture

Fig. 1(a) shows TONAK architecture. Inset 1 of Fig. 1(a) shows the optical transmitter

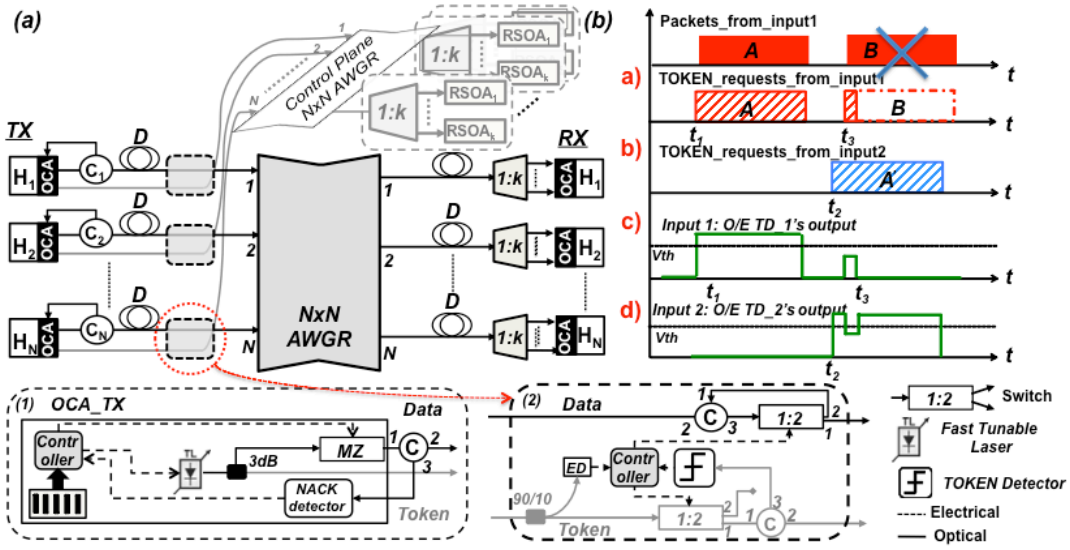

Figure 1. (a) Distributed TONAK architecture. *D* is the distance between hosts (H_i) and AWGRs input ports. AWGR: Arrayed Waveguide Grating Router; OCA: Optical Channel Adapter. Inset 1: host TX interface with ingress buffer queue (I-Q), fast tunable transmitter (TL), NACK detector, and Modulator. Inset 2: 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. (b) Timing diagram explaining how the all-optical control plane can detect contention.

generating both packets and token requests (TRs). Fig. 1(b) illustrates the token-based contention resolution. Assume that host_1 sends a packet to host_N. Host_1 will first tune its fast tunable laser (TL) to λ_{1N} (the wavelength to reach output *N* from input 1 according to AWGR routing table) to generate a TR ***A*** which reaches the Control Plane (CP) AWGR input port 1 at t=t_1. ***A*** is then routed to output *N*, where it enters in a Reflecting SOA (RSOA) after going through a 1:*k* optical demultiplexer. In general we assume *k* RSOAs for each CP AWGR output in order to exploit the wavelength parallelism and reduce the contention probability [3]. The RSOA amplifies and reflects the TR ***A***, which is extracted by an optical circulator placed on the token path right at the AWGR input (see inset 2 of Fig. 1(a)), and converted in the electrical domain by a token detector (TD). The TD (an O/E converter followed by a threshold comparator) generates an electrical signal with $V_p = V_{TO1}$ proportional to the optical power (P_{TO1}) of the reflected TR, and above a certain threshold V_{th}. This condition means that output *N* is available. A controller (FPGA or ASIC) sets the 1:2 $LiNbO_3$ switch [7] (switching time < 10ns) in the data path in position 1 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 to hold the token and to prevent collision. A 1:2 $LiNbO_3$ switch in the token path is set to position 1 anytime the Edge Detector (ED) senses an incoming TR.

The same situation described above arises when host_2 generates a TR and packet ***A'*** directed to output *N*. The reader should take note of the behavior at t=t_3, when the transmission of packet ***A'*** has not yet completed, but when host_1 wants to transmit another packet to output *N*. The RSOA at output *N*, already saturated with the TR ***A'*** at λ_{2N}, amplifies and reflects back the new TR ***B*** at λ_{1N}, which reaches the TD with optical power P_{TO3}. The TD generates an electrical signal with $V_p = V_{TO3}$. Assume that the RSOA has been saturated by the still active TR ***A'***. Due to the gain saturation effect [5], P_{TO3} will be ≈ Psat/2 and V_{TO3} will be ≈ $V_{TO1}/2$, where Psat is the output saturation power of the RSOA. By setting V_{th} between V_{TO1} and $V_{TO1}/2$, it becomes possible for the controller at input 1 to recognize that the token for output 1 is not available. The controller then sets the 1:2 $LiNbO_3$ switch in the data path to position 2. In this way, the incoming packet ***B*** is blocked and sent back to the TX, where it is extracted by an optical circulator and acts as AO-NACK (see [6] for more details on AO-NACK technique). The controller also sets the 1:2 $LiNbO_3$ switch for the token path to position 2; this immediately stops the denied TR ***B***, minimizing the number of unresolved collisions (see [5] for more details). 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 with AO-TOKEN are: (a) the TD is now placed at the switch input; (b) the use of AO-NACK technique to notify the senders of any contention. In this way, the delay between a TR and packet transmission, which was the limiting factor of AO-TOKEN, can almost be nullified.

Simulation Results

We developed a 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 in [6]. 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).

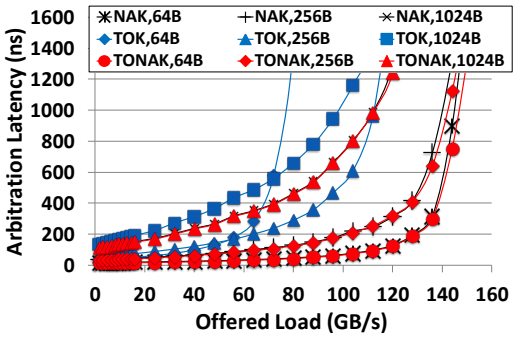

Figure 2. Arbitration latency as function of the offered load for uniform random traffic distribution.

Fig. 2 and 3 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 4m, and average packet sizes of 64B, 256B, and 1024B were simulated. Line-rate was 10Gb/s. TONAK significantly outperforms AO-TOKEN (TOK) for the reasons mentioned above. Performance is also slightly better than AO-NACK (NAK) architecture because TONAK does not require a guard-time (due to tunable laser tuning time) between consecutive packets with the same destination. Fig. 4 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.

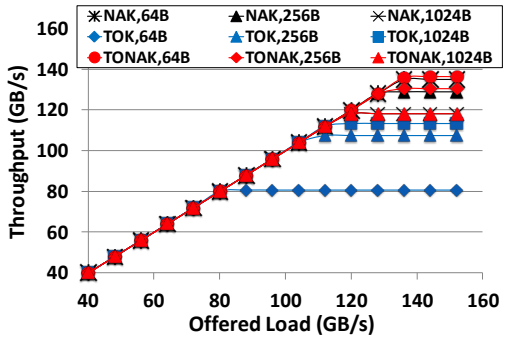

Figure 3. Throughput as function of the offered load for uniform random traffic distribution.

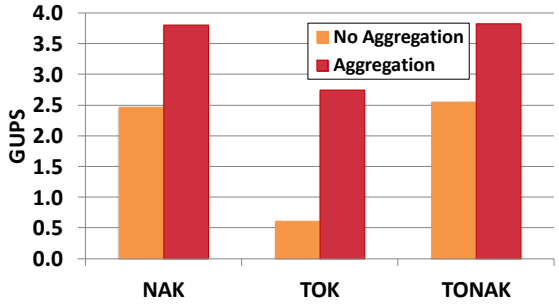

Figure 4. Giga Updates per Seconds (GUPS) benchmarking results.

The results shown in Fig. 4 are for both aggregation of requests and replies into larger packets and for requests/replies being sent independently.

Power consumption and scalability

Unlike electrical switches, in TONAK, power consumption and port count are not affected by the line-rate. With k RSOAs per output port, and each RSOA consuming typically ≈ 200mW, the power consumption scales as $0.2 \times k \times N$ Watt (W). Fig.5 shows how the switch energy efficiency improves with line-rate.

Scalability of single-AWGR architecture is limited by coherent crosstalk (≤128) [8]. In order to extend the port count beyond what can be obtained with a single AWGR, the architecture in Fig.6 can be used. With M^2 single-AWGR-based W-port switches, and with the assumption that each node has M transmitters and k receivers, an architecture which is able to interconnect $N = M \times W$ nodes in a single hop is possible. The use of a smaller size AWGR ($W \times W$ instead of $N \times N$) can simplify the wavelength registration problem (W instead of N wavelength values can be used), and more importantly, reduces the amount of optical crosstalk (crosstalk between W number of wavelengths in M^2 separate AWGRs instead of N number of wavelengths in one AWGR).

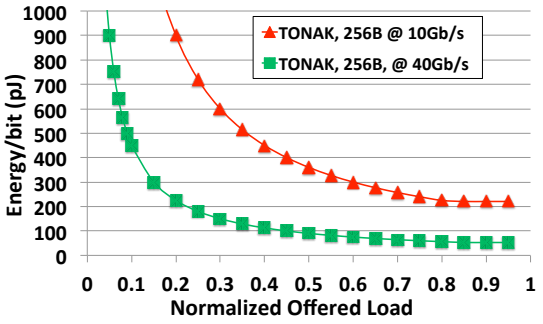

Figure 5. TONAK Energy/bit consumption for 10 Gb/s and 40 Gb/s line-rate.

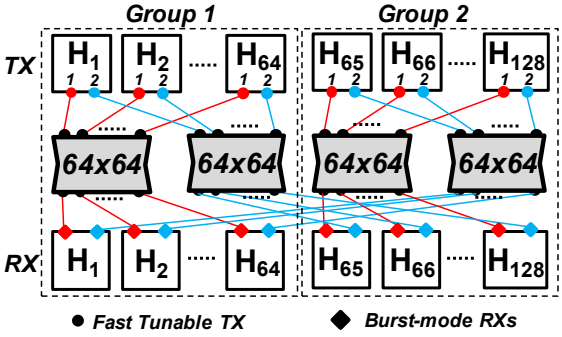

Figure 6. Example of 128-port switch built using multi-AWGR architecture composed by M^2 (4) AWGRs with W (64) ports.

Conclusions

The close host-switch distance typical of datacom networks enables TONAK, a scalable AWGR-based switch architecture that combines the advantages of AO-TOKEN (distributed control plane, modulation format transparency, no guard-time in between consecutive packets with the same destination) and AO-NACK (fast physical layer contention notification) techniques to guarantee low latency and high-throughput at high traffic load. Power consumption scales linearly with port-count and does not depend on the line-rate.

References

[1] L. A. Barroso, et al., Synthesis Lectures on Computer Architecture, **4** (1),1-108 (2009).

[2] F. Abel, et al., ACM Transactions on Networking, **15** (6), 1603-1615, (2007).

[3] X. Ye, et al., ACM/IEEE Symposium on Architectures for Networking and Communications Systems, 1-12 (2010).

[4] N. Farrington, et al., J SIGCOMM Comput. Commun. Rev., **40**, (4), 339-350 (2010).

[5] R. Proietti, et al., Journal of Selected Topics in Quantum Electronics, in press (2013).

[6] R. Proietti, et al., Photonics Technol. Letters, **24** (5), 410-412 (2012).

[7] http://www.eospace.com/switches.htm

[8] Takahashi, H. et al., J. of Lightwave Technol., **14** (6), 1097 – 1105 (1996).