Software-Defined Networking Control Plane for Seamless Integration of Silicon Photonics in Datacom Networks

Yiwen Shen, Payman Samadi, Ziyi Zhu, Alexander Gazman, Erik Anderson, David Calhoun, Maarten Hattink, Keren Bergman

Lightwave Research Laboratories, Electrical Engineering Department, Columbia University, U.S.A.
ys2799@columbia.edu

Abstract *We present a scalable Software-Defined-Networking (SDN) control-plane to integrate Silicon Photonics (SiP) with conventional Ethernet/InfiniBand networks and simultaneously perform packet and circuit switching. Experimental evaluations demonstrate this unique solution with 224 microseconds control plane latency for data-center and high-performance-computing platforms.*

Introduction

The ever-increasing growth in the scale of Data Centers (DC) and High Performance Computing (HPC) systems has escalated the importance of the efficiency and scalability of the interconnection network. The main challenge is providing a fast and non-blocking switching platform that supports high network concurrency, scalability, and energy efficiency. In today's DCs and HPC systems however, conventional networks employ a fixed physical layer, which leads to both data-starved processors connected by oversubscribed links, and wasted bandwidth allocated on links with minimal or no traffic. Moreover, for HPC systems, many scientific applications have well-defined traffic patterns between only a small number of racks in the entire network, creating a mismatch between the network topology and the application's traffic pattern [1].

Software-Defined Networking (SDN) can provide advanced network reconfiguration capabilities through modification of the flow tables in electronic switches. Optical circuit switching introduces new degrees of freedom in the network by enabling data rate transparent physical layer reconfiguration that provides opportunities for significant system performance improvement through more efficient network resource utilization. By leveraging Silicon Photonics (SiP) for optical circuit switching, we take advantage of its small area footprint, low power consumption, CMOS compatibility with low fabrication cost at large scales, and its potential for nanosecond range dynamic connectivity.

Researchers have been discovering various approaches to integrate SiP switching in DC and HPC networks, including architectural [1] and scalabilty [2,3] studies. The majority of these works were either focused on the network architecture or controlling the SiP element itself [4,5]. However, there has been little to no study on system-level integration of SiP switch with the SDN DC or HPC packet-switched networks control plane.

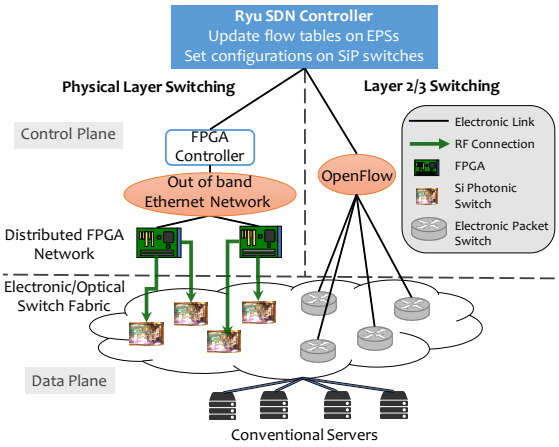


Fig. 1: Control plane architecture for seamless integration of SiP switches to DC and HPC networks.

In this work, we present an SDN approach to seamlessly integrate SiP switches with conventional networks control plane. Our proposal i) flexibly scales to manage several SiP switches, ii) works with any standard SDN controller such as Ryu and OpenDaylight, iii) is optimized and synchronized with Ethernet for minimal packet drop, and iv) is adaptable to HPC InfiniBand networks. We fully implemented the control plane and built a DragonFly [6] network testbed for evaluations. Experimental results show end-to-end control plane latency of 224 µs.

SDN Control Plane Architecture

Fig. 1 shows the control plane architecture, consisting of the physical layer switching module for the SiP switches and L2/3 switching module for electronic packet switches (EPSs). The SDN controller is developed using the Ryu SDN frame-

work, and manages the behavior of both the EPS and SiP switches in a synchronized and simultaneous manner. For the L2/3 switching module, the SDN controller adds/deletes flow rules to the flow table of the EPSs through the OpenFlow protocol, controlling the path that the traffic takes after the SiP switches have changed the physical topology.

In the physical layer switching module, the SDN controller sends the switch configuration commands to each SiP switch through a distributed FPGA network. The network is a 1G out-of-band Ethernet network that can scale in a fat-tree topology. An in-house developed C++ application interface translates the configuration messages to Ethernet packets. The connection between this interface and the SDN controller is a persistent TCP socket connection. Once the FPGA receives the configuration, per-defined voltages/currents are applied to the SiP switch using Digital-to-Analog Converters (DACs).

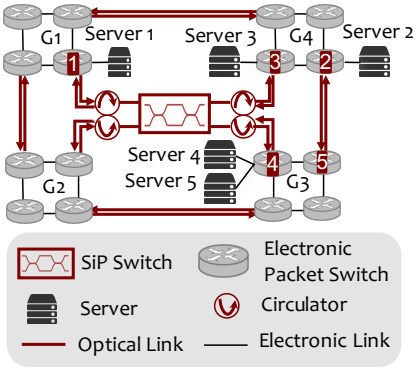
Testbed

We built a DragonFly network testbed consisting of 32 nodes and 4 groups as shown in Fig. 2(a) to evaluate the performance metrics of the control plane and demonstrate the potential benefits on a real network. Each group consist of 4 EPSs with 2 servers. The intra group connections is through 10G Direct-Attached copper cables and the inter group connections are optical links with 10G DWDM SFP+ transceivers with 24 dB power budget. All EPSs are OpenFlow enabled and connected to a separated SDN controller server. There is an out-of-band 1G Ethernet network for the OpenFlow and FPGA networks.

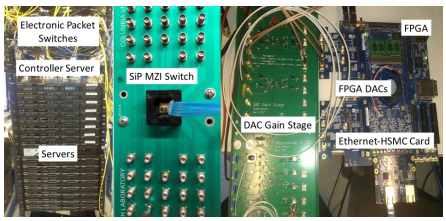
The SiP switch connects the four groups of the DragonFly and is a re-arrangeably non-blocking Mach-Zehnder Interferometer (MZI) based 4×4 Benes topology. For this particular demonstration, the switch is biased to either bar or cross state, performing as a 2×2 switch. Measured dynamic extinction ratios are 10 – 15 dB.

Experimental Results

We evaluated our proposed control plane by measuring the end-to-end latency for a synchronous circuit and packet switching experienced by packets on the network. The switching pipeline consists of software and hardware latencies. The hardware latency includes the switching time of the SiP switch and the EPS while updating the flow tables. We measured the SiP switching time by measuring the time that the RF signal is applied to the switch and the optical output reaches



(a)



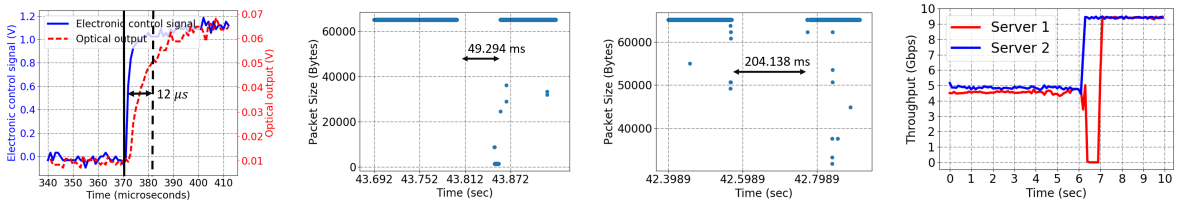
(b)

Fig. 2: (a) Experimental testbed of a DragonFly topology and a SiP switch for inter-group reconfiguration, (b) Snapshot of the testbed.

90% of the amplitude, shown in Fig. 3(a) as 12 µs. Carrier injection MZI switches are able to perform ns switching. However, in our testbed this switching speed was slower due to the DAC's sub-megahertz range sampling speed. The EPS latency was measured on the testbed by performing a data transfer between servers 2 and 3 on an indirect path and changing it to a direct path. Fig. 3(b) shows the results as 49.294 ms for L3 flow insert.

The software latencies include 1) the flow insertion latency, 2) the SDN controller to FPGA controller latency, and 3) the FPGA network latency. We measured the flow insertion latency on a 10G Pica8 switch by simultaneously inserting 800 flows using multiprocessing and achieved on average 78.5 µs per flow. Latency 2 is the latency for a TCP socket connection. We measured 223 µs that is half of the Round Trip Time (RTT) for a SiP switch reconfiguration command. The latency of the FPGA network (from the time the configuration packet is sent to the time that the DAC applies voltage) was measured to be 0.19 µs, which occurs simultaneously on multiple DACs.

Tab. 1 is a summary of all the components with labeled parallel actions. The overall control plane latency is 223.19 µs. Having evaluated all pieces of the switching latency, we performed a switching on the testbed and monitored the receiving packet rate. Data was sent from server 1 to server 4 in an indirect path with a bar configuration on the SiP



(a) (b) (c) (d)

Fig. 3: (a) MZI SiP switching time, (b) L3 flow insertion time on a OpenFlow electronic packet switch measured by monitoring packet transfer using tcpdump, (c) End-to-end switching time measured by monitoring packet transfers using tcpdump, (d) Demonstration of network optimization by SiP switch, improving the server throughput by redirecting the traffic to an unused link.

Tab. 1: Breakdown of control plane and hardware latencies

Control Plane Latency		Hardware Latency	
SDN controller per flow update	78.5 µs (parallel 1)	EPS L3 switching	49.294 ms (parallel 2)
SDN controller to FPGA controller	223 µs (parallel 1)	SiP switching	12 µs
FPGA controller to FPGA DACs	0.19 µs		
Total control plane		223.19 µs	
Transceiver locking		203.194 ms (parallel 2)	
Total end-to-end switching		204.138 ms	

switch through EPS 2, 3 and 5 and then through a direct path by switching to cross state on the SiP switch. Fig. 3(c) shows the tcpdump traces with 204.138 ms switching time. The extra 203.194 ms latency is due to the optical tranceiver locking latency after reconfiguration. We used commercial SFP+ modules that are not optimized for reconfiguration; however there exists transceivers with ns lock time[7].

Finally, to show the benefits of our control plane in practice, we show network optimization through bandwidth steering. In the testbed, servers 1 and 2 are transmitting data to servers 4 and 5. The SiP switch is in bar configuration so they need to share a 10G link. Fig. 3(d) shows the throughput. At second 6, we steered the bandwidth by reconfiguring the SiP switch to cross and providing an extra inter group connection between G1 and G3. In this state, servers 1 and 2 each have their own dedicated inter group link and can thus transmit with the full 10G capacity.

Conclusion and Discussion

While electronic packet switches are important and widely used in both HPC and DC networks, employing SiP for optical circuit switching can vastly improve their performance. However, integration of SiP with current SDN packet switched networks requires control planes capable of synchronized switching with minimal packet drop. Analyzing our results, with close to 50 ms latency on EPS flow update and 203 ms transceiver lock time, the SiP switch in μs switching time is sufficiently fast for physical layer reconfiguration and ns optical switching is not a major requirement. Developing commercial transceivers with μs locking and EPS with μs flow insert time will reduce the overall switch time to sub-milliseconds range that is equivalent to a few hundred Kb packet drops on a 10G link for online switching. Our proposed SDN control plane with sub-millisecond latency to reconfigure a network of SiP switches is a major required piece towards integration of SiP in Datacom networks. Our future work include demonstrating SDN SiP management at scale and integration of the control plane in an InfiniBand network.

Acknowledgments

This work was partly supported by the AIM Datacom and U.S. Department of Energy (DoE) Advanced Simulation and Computing (ASC) program through contract PO 1319001 with Sandia National Laboratories.

References

[1] K. Wen et al., “Flexfly: Enabling a Reconfigurable Dragonfly through Silicon Photonics,” Proc. SC, No. 15 (2016).

[2] D. Nikolova et al., ”Scaling silicon photonic switch fabrics for data center interconnection networks,” Opt. Exp. Vol. **23**, p. 1159-1175 (2015).

[3] K. Ishii et al., ”Toward exa-scale optical circuit switch interconnect networks for future datacenter/HPC,” Proc. SPIE Vol. **10131** (2017).

[4] R. Aguinaldo et al., ”Wideband silicon-photonic thermo-optic switch in a wavelength-division multiplexed ring network,” Opt. Exp. Vol. **22**, p.8205-8218 (2014).

[5] A. Gazman et al., ”Software-defined control-plane for wavelength selective unicast and multicast of optical data in a silicon photonic platform,” Opt. Exp. Vol. **25**, p. 232-242 (2017).

[6] J. Kim et al., ”Technology-Driven, Highly-Scalable Dragonfly Topology,” ISCA (2008).

[7] A. Rylyakof et al., ”22.1 A 25Gb/s burst-mode receiver for rapidly reconfigurable optical networks,” proc. ISSCC (2015).