Bufferless Optical Clos Switches for Data Centers

H. Jonathan Chao and Kang Xi

Polytechnic Institute of New York University, Brooklyn, New York 11201
{chao,kxi}@poly.edu

Abstract: A bufferless optical Clos switch called PetaX is designed to provide high-bandwidth data exchange for data center networks. PetaX combines the best features of electronics and optics and achieves good performance with low complexity.

OCIS codes: (060.4259) Networks, packet-switched; (060.6719) Switching, packet

1. Introduction

Data centers are critical infrastructure of the Internet, providing data- and/or computing-intensive services for a large variety of applications. Data centers are the only platform that can support large-scale cloud computing applications, such as Microsoft Azure, Amazon Elastic Compute Cloud (EC2), Google search, and Facebook. The scale of data centers has been growing steadily, from tens of thousands of servers to hundreds of thousands of servers in a single facility, requiring high-capacity, high-performance data center networks (DCNs) for data exchange.

Traditional DCNs have a multi-tier architecture [2, 6]. Servers are mounted to racks and connected to one or two top-of-rack (ToR) switches, which are then connected to access switches to form clusters. The clusters are interconnected using a small number of high-capacity aggregation routers. At the top, a few core routers interconnect the aggregation routers. Currently, it is common practice to introduce bandwidth oversubscription at each layer for cost savings. This architecture does not perform well at high loads due to bandwidth bottleneck.

Research has been conducted to build high-performance and scalable DCNs. One approach is to use commodity switches or servers to scale out the network, such as Portland [8], VL2 [3], and DCell [4]. The other approach is to exploit optical devices to build high-capacity switches, such as Data Vortex [5] and OSMOSIS [7]. We present a switch architecture that can scale up to petabit capacity, called PetaX. PetaX features a high-capacity bufferless optical switch fabric and low-complexity line cards.

2. PetaX Architecture

PetaX combines the best features of electronics and optics. Packets are buffered and processed electronically on the line cards, and then switched in an all optical bufferless switch fabric that is controlled by an electronic scheduler. This achieves ultra high capacity with flexible processing and control. The switch fabric has a three-stage Clos network [1] including input modules (IMs), central modules (CMs), and output modules (OMs) (refer Figure 1). Each IM/CM/OM uses an arrayed waveguide grating router (AWGR) as the core switch module. Each input port of the CMs and OMs has a tunable wavelength converter (TWC) to control the routing path. The IMs do not need TWCs because the line cards have tunable laser sources. PetaX design exploits the recent advances in optical devices, including tunable laser, TWC, AWGR and integrated optical devices.

Figure 2 shows a 4×4 switch fabric using 2×2 AWGRs and a path from input 3 to output 2. To establish the path, we configure the tunable devices so that the line card transmits at λ_0, the TWC in the CM keeps λ_0 unchanged, and the TWC in the OM converts λ_0 to λ_1. Amplifiers can be added between stages wherever necessary. Currently 128×128 AWGRs are available [9], so it is feasible to build a switch with 10,000 ports, each with 100 Gb/s line rate, achieving petabit scale capacity.

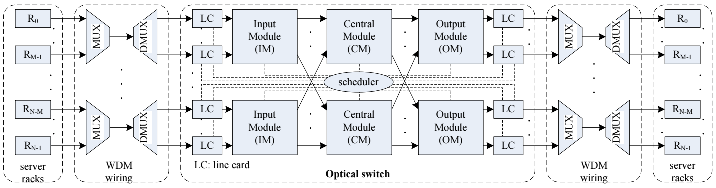

Fig. 1. Switch architecture.

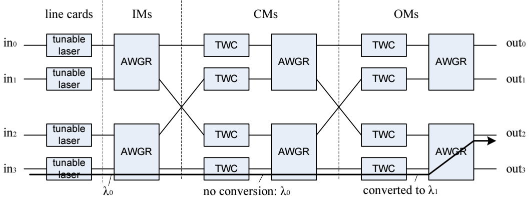

Fig. 2. A 4×4 switch fabric using 2×2 AWGRs and TWCs.

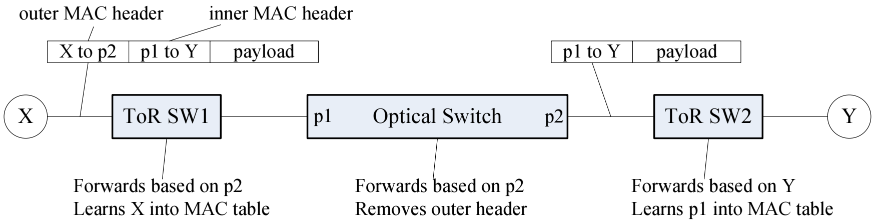

Fig. 3. Packet forwarding in 4SDCN.

Packets are assembled into fixed-size frames for switching. The bufferless optical switch modules are reconfigured synchronously. Ideally, arriving frames must be synchronized and precisely aligned. We relax this constraint by inserting a guard time between consecutive frames and using configurable fiber delay lines to compensate for the propagation delay difference. The guard time is also used to tolerate the switch reconfiguration time.

Fiber delay lines are installed at the input of each IM, CM, and OM so that frames from different line cards are aligned at the IMs and kept to be aligned when arriving at the CMs and OMs. Each delay line is configured based on the corresponding propagation delay with at most one frame period.

3. Forwarding

With PetaX a DCN is flattened in that all the ToR switches are directly connected to a single giant switch. To reduce complexity, PetaX avoids table lookup by letting servers encapsulate the input and output port IDs in headers. In Figure 3, server X performs MAC-in-MAC encapsulation when sending a packet to server Y (in-rack communication uses regular MAC frames). The outer header indicates the packet is from server X to port p2 of the optical switch. The inner header indicates the packet is from port p1 of the optical switch to server Y. ToR switches treat the port ID in the packet header just like a regular MAC address, while the optical switch uses the explicit output port ID to forward frames to avoid expensive table lookup. In the example, SW1 performs address learning and forwarding based on the outer MAC header. The optical switch uses p2 for forwarding at the ingress and then removes the outer header when assembling packets to frames. SW2 forwards the packet based on address Y and performs address learning by storing p1 in its MAC table. This scheme requires a centralized controller and modification of the address resolution protocol (ARP), which are feasible and have been adopted in related work [8, 3].

4. Scheduling Algorithm

The objective of scheduling is to find a bipartite match from the input ports to the output ports and assign a CM for each match such that the throughput is maximized. Existing algorithms have high implementation complexity, thus cannot scale to support a large number of ports. We present an iterative frame scheduling algorithm that is practical and scalable. Corresponding to each IM, CM and OM we have an scheduling module, which we call scheduler at IM (SIM), scheduler at CM (SCM) and scheduler at OM (SOM), respectively. The algorithm completes in H iterations as described below.

- **Request**: Each input port chooses H VOQs using round robin and sends the requests to the SIM that is corresponding to the IM with a connection to the requesting input port.
- **Iteration**: repeat the following steps for H cycles. In the hth cycle ($h = 1, 2, ..., H$), we only consider the hth requests from each input port.
  1. **Request filtering**: If an input port received a grant in the previous iterations, no further processing is done to its requests. For the remaining requests, if an SIM receives multiple requests pointing to the same output port, it chooses one randomly.
  2. **CM assignment**: Each SIM randomly assigns an available CM to each request and sends the requests to the corresponding SCMs.
  3. **CM arbitration**: If a SCM receives multiple requests pointing to the same OM, it chooses one using round robin. The selected requests are sent to the corresponding SOMs.
  4. **OM arbitration**: For each output port, the corresponding SOM grants the first request using round robin if the output port did not grant any requests in the previous iterations and is not overloaded.

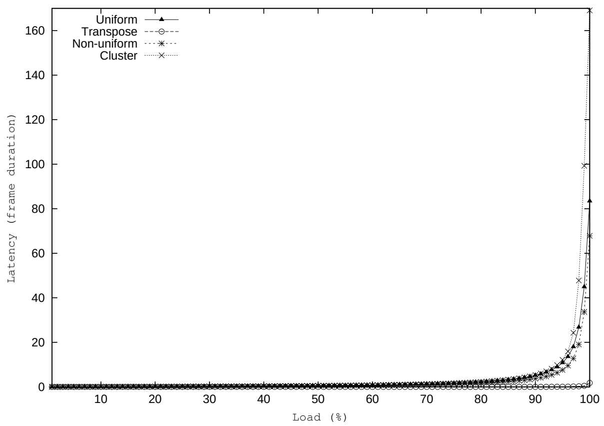

Fig. 4. Average latency (in frame duration) under various loads.

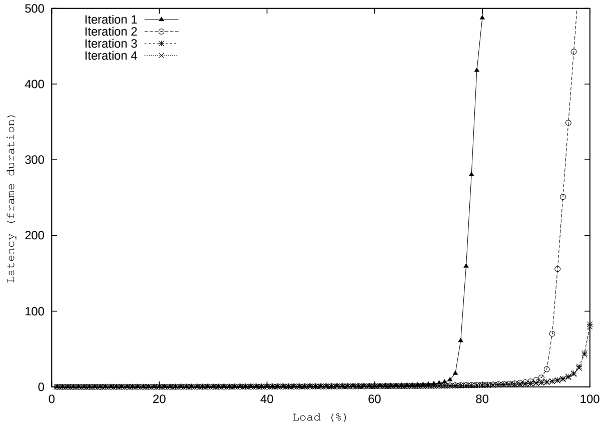

Fig. 5. Average latency (in frame duration) with various iterations.

In our switch there are two contention places: at CMs and at output ports. We combine multiple approaches to resovle the contention. We introduce randomness to reduce the contention probability. We allow each input to generate in H requests. (Note that generating N requests would be good but incurrs high communication overhead.) We also employ multiple iterations and speedup. Our simulation shows that with three iterations and 1.6 speedup, the scheduling algorithm achieves nearly 100% throughput under various traffic distributions and switch sizes.

5. Performance Evaluation

We use simulation to evaluate the scheduling algorithm. Our simulation shows that PetaX can achieve 99.6% throughput with three iterations and a speedup factor of 1.6 when the port number is 10,000. When the port number increases from 1024 to 10,000, at 90% load, the average latency changes only from 5.13 to 5.19 frames. This shows that PetaX has excellent scalability. Figure 4 shows that PetaX achieves near 100% throughput consistently under various traffic patterns (the port number is 1024). Figure 5 shows that three iterations are sufficient to achieve good performance.

6. Conclusions

A large-scale switch is designed to meet the requirements of data center networks for high capacity data exchange. The switch combine the best of optics and electronics to reduce the complexity. A scalable and practical scheduling algorithm is developed and its performance is verified using simulations.

7. Acknowledgement

This research is supported by CACI Technologies under contract S09-096600.

References

1. H. J. Chao and B. Liu. *High Performance Switches and Routers*. Wiley-IEEE Press, 2007.
2. Cisco. Cisco Data Center Infrastructure 2.5 Design Guide. Cisco Systems, Inc., Dec. 2007.
3. A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta. VL2: a scalable and flexible data center network. In *SIGCOMM '09: Proceedings of the ACM SIGCOMM 2009 conference on Data communication*, pages 51–62, New York, NY, USA, 2009. ACM.
4. C. Guo, H. Wu, K. Tan, L. Shi, Y. Zhang, and S. Lu. DCell: A scalable and fault-tolerant network structure for data centers. In *SIGCOMM '08: Proceedings of the ACM SIGCOMM 2008 conference on Data communication*, pages 75–86, New York, NY, USA, 2008. ACM.
5. C. Hawkins, B. A. Small, D. S. Wills, and K. Bergman. The Data Vortex, an All Optical Path Multicomputer Interconnection Network. *IEEE Trans. Parallel Distrib. Syst.*, 18(3):409–420, 2007.
6. Juniper. Juniper Networks Data Center LAN Connectivity Design Guide, June 2008.
7. R. Luijten and R. Grzybowski. The OSMOSIS Optical Packet Switch for Supercomputers. In *Optical Fiber Communication Conference (OFC)*, 2009.
8. R. Niranjan Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya, and A. Vahdat. PortLand: a scalable fault-tolerant layer 2 data center network fabric. In *SIGCOMM '09: Proceedings of the ACM SIGCOMM 2009 conference on Data communication*, pages 39–50, New York, NY, USA, 2009. ACM.
9. F. Xue and S. Ben Yoo. High-capacity multiservice optical label switching for the next-generation Internet. *IEEE Communications Magazine*, 42(5):S16 – S22, may 2004.