Scalabilitiy and Congestion Control in Oblivious Reconfigurable Networks

Daniel Amir
Cornell University
Ithaca, New York, USA

Tegan Wilson
Cornell University
Ithaca, New York, USA

Vishal Shrivastav
Purdue University
West Lafayette, Indiana, USA

Hakim Weatherspoon
Cornell University
Ithaca, New York, USA

Robert Kleinberg
Cornell University
Ithaca, New York, USA

CCS CONCEPTS

• **Networks** → **Data center networks**; **Network design principles**; **Network simulations**.

KEYWORDS

Datacenter networks, circuit-switched networks, oblivious routing

Traditional datacenter networks have been designed primarily using packet switches. However, due to the end of Moore's Law and Denard Scaling, packet switches face increasing difficulty in scaling to meet network demands without consuming unnecessarily large amounts of power, both within high-density racks[14] and throughout the datacenter[1]. As a result, many emerging network designs have intentionally avoided using packet switches [5, 7, 9, 10, 12, 15, 16]. Circuit switches present an exciting alternative to packet switches due to their reduced power consumption[1, 14], and potential to scale to arbitrary bandwidth (in the case of optical switches). While slow reconfiguration times have historically made circuit switches unable to support low-latency traffic, recent circuit switch design have emerged that are capable of nanosecond-scale reconfiguration times, including both electrical [11] and optical [3, 4, 6] switches. Unfortunately, conventional, dynamically-reconfiguring circuit-switched network designs have inherent latencies both for computing which circuits to deploy and for coordinating switches and nodes, limiting the benefits of this new capability.

Oblivious Reconfigurable Networks (ORNs) are a new network design paradigm that can realize the potential of modern, rapid circuit switches. In ORNs, circuit switches are reconfigured oblivious to traffic, using a predetermined schedule of timeslots. During each timeslot, switch configurations are fixed, connecting nodes according to the connection schedule for long enough to send a single frame. Timeslots are separated by guard bands, during which switches are reconfigured and no data is sent. Guard bands as short as 3.84 ns have been demonstrated for optical ORNs [3], and by using multiple lanes to parallelize the schedule, timeslots can be started every 5.6 ns [14]. To support arbitrary traffic with a fixed schedule, ORNs use an indirect routing scheme to route data to its destination. By co-designing the schedule and routing scheme, good performance can be achieved over all workloads. RotorNet [8], Shoal [14], and Sirius [3] are three designs following the ORN concept that have been demonstrated on physical test-beds. However, all three use schedules based on a single round-robin among all nodes, as shown in fig. 2a. This schedule maximizes throughput at the cost of poor latency scaling (linear in system size), limiting the applicability of these systems to large systems.

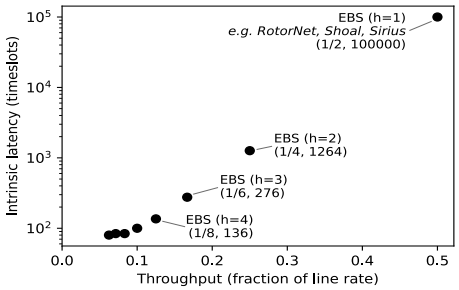


Figure 1: A comparison of the throughput and intrinsic latencies achieved by various tunings of EBS for a 100,000-node network.

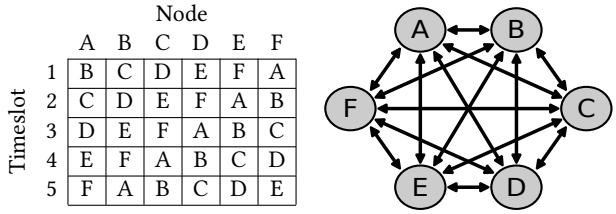
Scalable ORNs. Recent theoretical research [2] has developed a family of ORN designs, known as *EBS*[1], that achieve multiple different tradeoffs between throughput and latency, all of which are Pareto optimal for ORNs (up to a constant factor). These designs generalize the schedule and routing scheme used by existing proposals, and are practical to implement using the same technologies. Rather than participating in a single round-robin with all other nodes, each node instead participates in h smaller round-robins, each of which has only $\sqrt[h]{N}$ participants, as shown in fig. 2b. Compared to existing designs, EBS reduces the latency from $O(N)$ to $O(h\sqrt[h]{N})$. While the throughput is also reduced from $\frac{1}{2}$ to $\frac{1}{2h}$, the tradeoff is Pareto optimal (up to a constant factor) [2]. However, new congestion control mechanisms are needed to achieve these theoretical latencies in practice. Due to the unique properties of ORNs, existing congestion control systems, especially end-to-end mechanisms, are a poor fit.

The proof of the Pareto optimality of EBS (as developed in [2]) implies several properties ORNs must have to achieve good performance. First, in order to guarantee throughput across arbitrary workloads, they must load-balance across manyindirect paths for each flow, up to $O(N)$. Second, in order to achieve scalable latency characteristics, they must use sufficiently long path lengths.

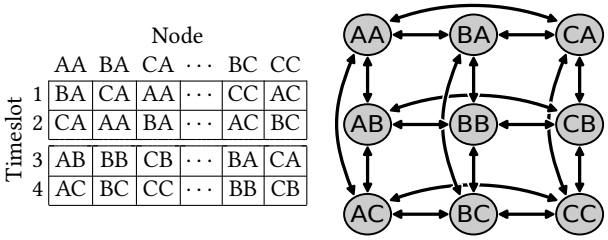
[1]Elementary Basis Scheme, named for a mathematical construction used to define it

ACM SIGCOMM '23, September 10, 2023, New York, NY, USA

ACM ISBN 979-8-4007-0236-5/23/09.
https://doi.org/10.1145/3603269.3610862



(a) Single round-robin schedule for six nodes.



(b) EBS schedule with $h = 2$ for nine nodes. Here, each node is labeled with two letters, and participates in round robins with nodes differing by only one letter.

Figure 2: Sample schedules for both existing ORN designs and EBS with $h = 2$, with visualizations of all connections formed over the course of each schedule.

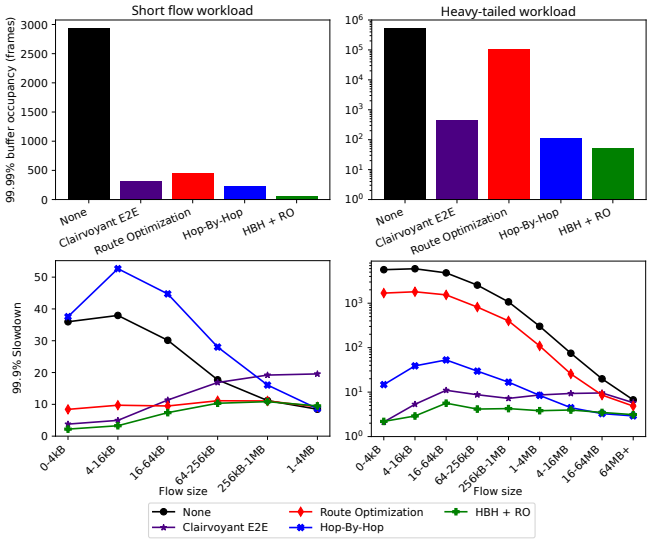


Figure 3: Buffer occupancies and tail flow completion time slowdowns for 4096-node EBS with $h = 4$.

Congestion in ORNs. There are two primary causes of congestion in ORNs. The first, *egress congestion*, occurs when frames bound for the same destination accumulate in queues leading to their destination. In ORNs, as in other contexts, incast can cause this type of congestion. However, because each flow must use many paths in an ORN, it is possible for two frames from the same flow to experience egress congestion. Unadressed, egress congestion can be long-lived, and is promoted by heavy-tailed workloads in which most bytes are sent in long flows.

In the second form of congestion, *path-collision congestion*, frames with unrelated destinations happen to be enqueued at the same node to be sent to the same next hop. Due to the use of indirect paths, path collisions can occur between many different source destination pairs. While path-collision congestion is unlikely to be sustained on its own, it can magnify the impact of egress congestion when both occur simultaneously. Path-collision congestion can occur spontaneously for all workloads.

Congestion control in scalable ORNs. ORNs pose a unique environment for congestion control. First, ORNs exhibit heavy reliance on *multi-pathing* in order to ensure good throughput across arbitrary traffic. Traditional congestion control mechanisms, such as the TCP-suite of algorithms, expect packets from a given flow to take a single paths, or sometimes a small handfull of paths. However, ORNs often require $O(N)$ paths in order to achieve good performance. Short flows may never have two frames traverse the same path, and difficult-to-untangle *fate sharing* makes it impractical to apply congestion information from one path to another without becoming too conservative. The *long path lengths* necessary to achieve scalability mean that congestion can occur far from both the sender and the receiver of a frame. Finally, because *queues empty slower than line-rate*, sending only one frame per schedule iteration, ORNs are especially sensitive to even low levels of queuing delay. All of these factors suggest that end-to-end congestion control is a poor fit in the ORN context, motivating the development of novel congestion control mechanisms for this context.

While existing ORN proposals have congestion control mechanisms, they are enabled by the short, two-hop paths used by these designs. This prevents them from being used with scalable ORNs such as EBS, which must use longer paths. We have developed two congestion control mechanisms that together provide impressive performance for EBS. The first design is a hop-by-hop design that effectively addresses egress congestion. This design uses Push-In-Extract-Out queues [13] to pause sending frames to congested destinations without affecting traffic to other destinations. The second is a routing optimization that takes advantage of flexibility in the first half of paths in EBS. For those hops, rather than sending via a randomly selected next hop, nodes instead send via a next hop that has a short local send queue, reducing path collisions. While this design deviates slightly from EBS's routing scheme, experiments have shown that it does not violate the throughput guarantees of EBS in practice. Together, these designs proactively address all forms of congestion as soon as they occur within the network.

Figure 3 shows the results of packet-level simulations of a 4096 node network using EBS with $h = 4$. In these simulations, the combination of both of our congestion control mechanisms is able to outperform an idealized, clairvoyant end-to-end congestion control for both a short-flow and a heavy-tailed workload.

Acknowledgments

This work was supported in part by NSF grants CSR-1704742, CHS-1955125, DBI-2019674, and CAREER-2239829; Cisco Research Award #23089533; a Google Research Scholar award; and a Microsoft Investigator Fellowship.

REFERENCES

[1] Slavisa Aleksic. 2010. Electrical Power Consumption of Large Electronic and Optical Switching Fabrics. 95 – 96. https://doi.org/10.1109/PHOTWTM.2010.5421958

[2] Daniel Amir, Tegan Wilson, Vishal Shrivastav, Hakim Weatherspoon, Robert Kleinberg, and Rachit Agarwal. 2022. Optimal Oblivious Reconfigurable Networks. In *Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing (STOC 2022)*. Association for Computing Machinery, New York, NY, USA, 1339–1352. https://doi.org/10.1145/3519935.3520020

[3] Hitesh Ballani, Paolo Costa, Raphael Behrendt, Daniel Cletheroe, Istvan Haller, Krzysztof Jozwik, Fotini Karinou, Sophie Lange, Kai Shi, Benn Thomsen, et al. 2020. Sirius: A Flat Datacenter Network with Nanosecond Optical Switching. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 782–797.

[4] Q. Cheng, A. Wonfor, J. L. Wei, R. V. Penty, and I. H. White. 2014. Demonstration of the feasibility of large-port-count optical switching using a hybrid Mach–Zehnder interferometer–semiconductor optical amplifier switch module in a recirculating loop. *Opt. Lett.* 39, 18 (Sep 2014), 5244–5247. https://doi.org/10.1364/OL.39.005244

[5] Paolo Costa, Hitesh Ballani, Kaveh Razavi, and Ian Kash. 2015. R2C2: A Network Stack for Rack-scale Computers. In *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication (SIGCOMM '15)*. ACM, New York, NY, USA, 551–564. https://doi.org/10.1145/2785956.2787492

[6] M. Ding, A. Wonfor, Q. Cheng, R. V. Penty, and I. H. White. 2017. Scalable, low-power-penalty nanosecond reconfigurable hybrid optical switches for data centre networks. In *2017 Conference on Lasers and Electro-Optics (CLEO)*. 1–2.

[7] Monia Ghobadi, Ratul Mahajan, Amar Phanishayee, Nikhil Devanur, Janardhan Kulkarni, Gireeja Ranade, Pierre-Alexandre Blanche, Houman Rastegarfar, Madeleine Glick, and Daniel Kilper. 2016. ProjecToR: Agile Reconfigurable Data Center Interconnect. In *Proceedings of the 2016 ACM SIGCOMM Conference (SIGCOMM '16)*. Association for Computing Machinery, New York, NY, USA, 216–229. https://doi.org/10.1145/2934872.2934911

[8] Soudeh Ghorbani, Zibin Yang, P. Brighten Godfrey, Yashar Ganjali, and Amin Firoozshahian. 2017. DRILL: Micro Load Balancing for Low-Latency Data Center Networks. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication (SIGCOMM '17)*. Association for Computing Machinery, New York, NY, USA, 225–238. https://doi.org/10.1145/3098822.3098839

[9] Navid Hamedazimi, Zafar Qazi, Himanshu Gupta, Vyas Sekar, Samir R. Das, Jon P. Longtin, Himanshu Shah, and Ashish Tanwer. 2014. FireFly: A Reconfigurable Wireless Data Center Fabric Using Free-Space Optics. In *Proceedings of the 2014 ACM Conference on SIGCOMM (SIGCOMM '14)*. Association for Computing Machinery, New York, NY, USA, 319–330. https://doi.org/10.1145/2619239.2626328

[10] Sergey Legtchenko, Nicholas Chen, Daniel Cletheroe, Antony Rowstron, Hugh Williams, and Xiaohan Zhao. 2016. XFabric: A Reconfigurable In-Rack Network for Rack-Scale Computers. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. USENIX Association, Santa Clara, CA, 15–29. https://www.usenix.org/conference/nsdi16/technical-sessions/presentation/legtchenko

[11] Macom M21605 Crosspoint Switch 2015. Macom M21605 Crosspoint Switch. www.macom.com/products/product-detail/M21605/. (2015).

[12] George Porter, Richard Strong, Nathan Farrington, Alex Forencich, Pang Chen-Sun, Tajana Rosing, Yeshaiahu Fainman, George Papen, and Amin Vahdat. 2013. Integrating Microsecond Circuit Switching into the Data Center. In *Proceedings of the ACM SIGCOMM 2013 Conference on SIGCOMM (SIGCOMM '13)*. Association for Computing Machinery, New York, NY, USA, 447–458. https://doi.org/10.1145/2486001.2486007

[13] Vishal Shrivastav. 2019. Fast, Scalable, and Programmable Packet Scheduler in Hardware. In *Proceedings of the ACM Special Interest Group on Data Communication (SIGCOMM '19)*. Association for Computing Machinery, New York, NY, USA, 367–379. https://doi.org/10.1145/3341302.3342090

[14] Vishal Shrivastav, Asaf Valadarsky, Hitesh Ballani, Paolo Costa, Ki Suh Lee, Han Wang, Rachit Agarwal, and Hakim Weatherspoon. 2019. Shoal: A Network Architecture for Disaggregated Racks. In *16th USENIX Symposium on Networked Systems Design and Implementation (NSDI 19)*. USENIX Association, Boston, MA. https://www.usenix.org/conference/nsdi19/presentation/shrivastav

[15] Ankit Singla, Atul Singh, and Yan Chen. 2012. OSA: An Optical Switching Architecture for Data Center Networks with Unprecedented Flexibility. In *Presented as part of the 9th USENIX Symposium on Networked Systems Design and Implementation (NSDI 12)*. USENIX, San Jose, CA, 239–252. https://www.usenix.org/conference/nsdi12/technical-sessions/presentation/chen_kai

[16] Meg Walraed-Sullivan, Jitendra Padhye, and David A. Maltz. 2014. Theia: Simple and Cheap Networking for Ultra-Dense Data Centers. In *Proceedings of the 13th ACM Workshop on Hot Topics in Networks (HotNets-XIII)*. ACM, New York, NY, USA, Article 26, 7 pages. https://doi.org/10.1145/2670518.2673885