A case for server-scale photonic connectivity

Abhishek Vijaya Kumar
Cornell University

Arjun Devraj
Cornell University

Darius Bunandar
Lightmatter

Rachee Singh
Cornell University

Abstract

The commoditization of machine learning is fuelling the demand for compute required to both train large models and infer from them. At the same time, scaling the performance of individual microprocessors to satisfy the demand for compute has become increasingly difficult since the end of Moore's law and Dennard scaling. As a result, compute resources in modern servers are distributed across multiple accelerators on the server board. In this work, we make the case for using optics to interconnect accelerators within a server. A key benefit of on-board chip-to-chip optical connectivity is its ability to dynamically allocate bandwidth between accelerators, where necessary, rather than the common practice of statically dividing bandwidth among links within the topology of a multi-accelerator server, as seen in popular direct-connect architectures. This property prevents bandwidth under-utilization in state-of-the-art rack-scale multi-accelerator deployments. Moreover, server-scale optical connectivity can reduce the blast radius of individual accelerator failures in rack-scale ML deployments. Our early experiments with the prototype of a newly commercialized server-scale photonic interconnect show how the capability of the hardware can enable our vision.

CCS Concepts

• **Hardware** → **Emerging optical and photonic technologies**; • **Networks** → **Physical links**; • **Computer systems organization** → *Fault-tolerant network topologies*; • **Computing methodologies** → *Machine learning*.

Keywords

Silicon photonics, optical networks, reconfigurable networks, collective communication, distributed machine learning

ACM Reference Format:
Abhishek Vijaya Kumar, Arjun Devraj, Darius Bunandar, and Rachee Singh. 2024. A case for server-scale photonic connectivity. In *The 23rd ACM Workshop on Hot Topics in Networks (HOTNETS '24), November 18–19, 2024, Irvine, CA, USA.* ACM, New York, NY, USA, 10 pages. https://doi.org/10.1145/3696348.3696856

1 Introduction

Optical data transmission offers significant advantages over its electrical counterpart, notably in achieving higher data rates and enabling longer communication reach [54]. These benefits stem from the lower loss incurred in transmitting light through media like glass, compared to the higher loss encountered by electrical signals in copper wires. As a result, light has become the primary medium for data transmission across various connectivity scales [45].

Optics in the long-haul and datacenter. The early 2000s marked the commercial adoption of optical communication, with extensive deployment of optical fiber for long-haul Internet connectivity [6, 48]. The subsequent rise of cloud computing spurred the development of cloud datacenters worldwide. To meet the increasing traffic demands in these datacenters, cloud providers interconnected switches using optical fiber in a clos topology [17, 46]. Similarly, growing bandwidth of server network interface cards caused cloud providers to further leverage optical fiber to connect compute servers with top-of-rack switches in datacenter racks [35, 46]. Consequently, the majority of physical connectivity in modern long-haul and datacenter networks is driven by optics.

Decision-making in the optical domain. Physical connectivity has been underpinned by optics in both long-haul and datacenter networks; however, routing decisions in these networks were largely made by electrical packet switches. Recent work has challenged this strict separation of concerns, where optical equipment provides fixed logical connectivity and electrical switches perform dynamic routing. This has resulted in the emergence of *photonic fabrics* in both datacenters [11, 15, 23, 28, 52] and long-haul networks [48, 59] where optical equipment participates in adapting the physical connectivity for performance and fault tolerance of workloads. Most recently, the ability to reconfigure the topology of photonic fabrics was found to be useful in accelerating

HOTNETS '24, November 18–19, 2024, Irvine, CA, USA

ACM ISBN 979-8-4007-1272-2/24/11
https://doi.org/10.1145/3696348.3696856

performance-sensitive distributed machine learning (ML) workloads in cloud datacenters [28, 60].

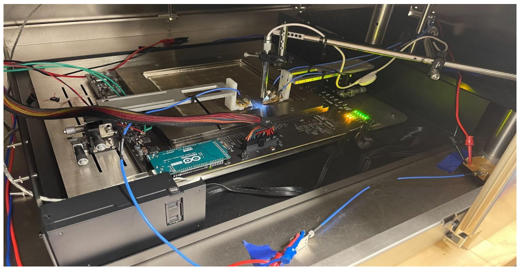


Figure 1: Our 200mm x 200mm server-scale chip-to-chip interconnect prototype wirebonded in a socket.

The home stretch: server-scale optical connectivity. While optical connectivity has permeated datacenter racks, interconnects between chips on compute servers remain electrical [32, 34]. This is true for multi-accelerator servers (*e.g.,* Nvidia DGX [7, 20, 33]) that are the building blocks of datacenter clusters and have become workhorses of performance sensitive ML workloads. These servers consist of a handful of ML accelerators (*e.g.,* GPUs, TPUs), connected using on-board electrical interconnects (*e.g.,* PCIe [32], NVLink [34], ICI [23, 60]) either in direct-connect [23, 58] or switched [33, 34] topologies. In this work, we argue that extending optics to the server board by optically interconnecting accelerators within a multi-accelerator server can unlock both improved performance and fault tolerance of ML workloads compared to both direct-connect and switched multi-accelerator servers.

The key advantage of optically connecting accelerators in a server is the ability of the server-scale photonic interconnect to dynamically redirect bandwidth from one accelerator to another. This allows accelerators to transmit at their full egress bandwidth to different neighbors based on the communication pattern of collective primitives in use (*e.g.,* AllReduce) [10]. Using this property, server-scale optical connectivity can achieve better performance compared to:

- **Switched multi-accelerator servers.** Switched electrical multi-accelerator server topologies attempt to leverage the "big-switch" [5, 17, 29] abstraction where the on-board multi-accelerator interconnect has ideal contention-free switches (*e.g.,* NVSwitch). However, inter-accelerator bandwidth within modern servers is already massive — over 300 gigabytes per second in one direction [34]— making it harder to stay true to the ideal switch abstraction. This has resulted in evidence of contention in switched sever-scale interconnects [4, 42]. In contrast, chip-to-chip photonic connectivity enables contention-free networks between accelerators, similar to electrical direct-connect topologies without incurring the disadvantages of direct-connect topologies, namely, underutilized accelerator bandwidth due to fixed inter-accelerator connections.
- **Direct-connect multi-accelerator servers.** A key advantage of on-board chip-to-chip optical connectivity is the ability to dynamically redirect bandwidth from an accelerator along connections where it is needed as opposed to statically partitioning accelerator bandwidth among a subset of inter-accelerator links in the topology of the multi-accelerator server, as is common in direct-connect topologies [23].

Shrinking the blast radius of accelerator failures. Recent work has shown that optical fabrics at the datacenter scale improve the resilience of large ML jobs — jobs that collectively use multiple racks of accelerator chips — in the event of the failure of an accelerator in the direct-connect cluster [23, 60]. For instance, Google's TPU supercomputer migrates a multi-rack ML job away from the rack with a failed TPU chip to a different rack. In addition to migrating the job, the new set of TPU racks are directly connected — without inter-rack electrical packet switching — by reconfiguring the optical circuit switches that link all TPU racks [23, 60]. Even so, reconfigurable datacenter fabrics have an excessively large blast radius, *i.e.,* the extent of impact of the failure of a single accelerator chip. Not only is the migration expensive for the job interrupted by failure, it may also be infeasible to find an entirely unused set of servers for every job with a single failed TPU. We show that server-scale photonics enables routing around TPU chip failures to reduce the blast radius of a single chip failure to only the multi-accelerator server containing the failed chip.

Prototype chip-to-chip optical hardware. To achieve the vision of optical multi-accelerator interconnects, we propose leveraging the recent advances in silicon photonics that have made server-scale photonic interconnect hardware commercially viable [19]. Figure 1 shows our lab prototype of a recent server-scale optical interconnect, Lightpath. Accelerator chips can be stacked on top of *tiles* of the interconnect, forming a grid (Figure 2c). Each tile on the interconnect grid has silicon-based photonic components like lasers, waveguides, switches and photodetectors (Figure 2) to generate, transmit, switch and receive optical signals between chips. We show that Lightpath can connect up to 32 accelerators where each accelerator is 3D stacked on a Lightpath tile equipped with 16 lasers and photodiodes. One wavelength can sustain up to 224 Gbps bandwidth, and programming optical switches on Lightpath can take up to 3.7μs.

A brave new world of optical host interconnects. Our experiments with the lab prototype of a server-scale multi-accelerator interconnect highlight the capability of the technology (*e.g.,* reconfiguration delay, signal loss) and how this capability can enable high-bandwidth and contention-free communication between accelerators. Adoption of server-scale optics will allow interconnect bandwidth to scale beyond the limit of copper wires and open the door to taking a fresh look at several classical networked systems challenges. For instance, server-scale optics will necessitate the development of new host networking software stacks optimized for circuit-switching as opposed to today's packetized data transmission. Moreover, it takes several microseconds to establish optical circuits between chips, therefore, new optical resource allocation algorithms will be needed to arrive at the appropriate trade-off between optical reconfiguration delay and end-to-end server-scale interconnect performance. Our goal is to spark a discussion in the community about the promise and challenges of server-scale optics in next-generation multi-accelerator servers.

We first provide background on multi-accelerator communication in ML (§2). Next, we introduce our prototype of a server-scale photonic interconnect, Lightpath, and discuss its design and capabilities (§3). Then, we present how server-scale photonics (using Lightpath as an example) offers novel opportunities in state-of-the-art rack-scale deployments, from resolving bandwidth under-utilization to limiting the blast radius of individual accelerator failures (§4). Finally, we discuss challenges that this new technology presents (§5) and related work (§6).

2 Multi-accelerator communication in ML

Modern ML models have trillions of parameters, making it infeasible for these models to fit in the memory of a single accelerator [12, 31]. Therefore, it has become essential to distribute the training and inference of large ML models on multiple accelerators (*e.g.,* GPUs, TPUs) with data, model and pipeline parallelism [12, 27, 44]. For instance, OpenAI and Microsoft distributed training of the popular GPT-4 model on thousands of GPUs in a private datacenter cluster [13].

Server-scale multi-accelerator systems (e.g., Intel Gaudi [20], Nvidia DGX [33], Cerebras WSE [7]) are the building blocks of larger clusters and datacenters for distributed ML training and inference. These systems consist of several ML accelerators connected by high-speed on-board electrical interconnects (e.g., PCIe [32], NVlink [34]), supporting up to hundreds of gigabytes per second of bandwidth between pairs of accelerators. Cloud operators connect racks of server-scale multi-accelerator systems into datacenter-scale deployments using a network fabric [17, 23]. This allows cloud customers to lease virtual instances of these systems to deploy their ML models. Depending on the model size, training and inference are distributed across accelerators on the same server or multiple servers in the cloud [23, 35].

Intermediate parameters of the model are accumulated, reduced and transferred over the network between accelerators using *collective communication* primitives [10] like AllReduce in distributed ML. This places collective communication on the critical path of both training and inference of distributed ML models. Recently, researchers have shown that accelerators remain idle during training for large fractions of the time waiting for inter-accelerator communication to complete [2, 8, 9, 14, 25], highlighting the importance of efficient collective communication.

3 Server-scale optical interconnects

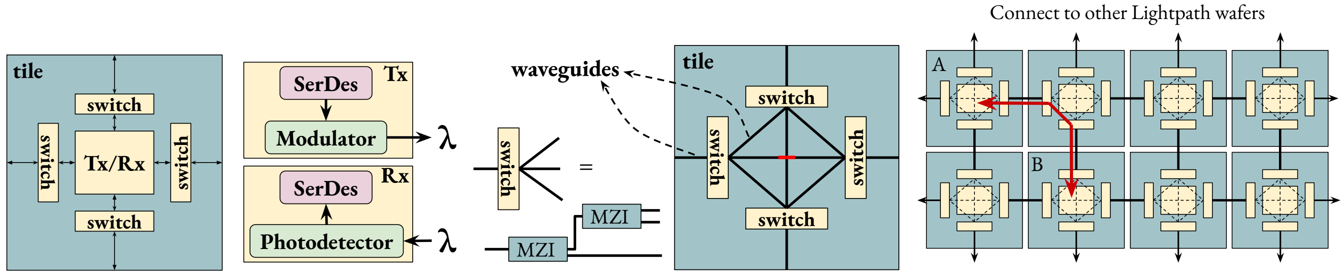
In this section, we first describe the components of a recently commercialized server-scale photonic interconnect, Lightpath. We then set up a lab prototype of Lightpath and experimentally study physical-layer characteristics of the interconnect. Based on the capabilities of the interconnect highlighted in this section, we will show how it can improve collective communication during distributed ML in §4.

Lightpath. Lightpath is a photonic fabric in a hybrid Complementary Metal Oxide Semiconductor (CMOS) photonics process [21]. Figure 1 shows our lab prototype of Lightpath. A Lightpath wafer consists of 32 tiles that can interconnect 32 chips by bonding or stacking one chip (e.g., GPU, TPU) per tile. Chips are stacked to tiles on Lightpath, forming a grid (Figure 2c). Each tile on the interconnect grid has silicon-based photonic components like lasers, waveguides, switches, and photodetectors (Figure 2) to generate, transmit, switch, and receive optical signals between chips.

Modulators and Photodetectors. At the center of each Lightpath tile is a Transmitter and a Receiver (Figure 2a). The optical transmitter (Tx) converts data from the chip to optical signals by modulating a wavelength of light. In Lightpath, we use micro-ring modulators (MRRs) to modulate light signals with data. The optical receiver (Rx) demultiplexes multiple wavelengths of light, converts modulated wavelengths back to electric signals using photodetectors, and sends them to the Serializer/Deserializer (SerDes).

Light sources and waveguides. Each Lightpath tile has 16 wavelength-multiplexed lasers. Optical waveguides transport optical wavelengths generated by the lasers across tiles. Waveguides form edges of a two-dimensional grid that connects Lightpath tiles. While each waveguide can support multiplexed wavelengths that carry data, the number of connections that can be made by one Lightpath tile is limited by the number of SerDes ports available in the electrical chip.

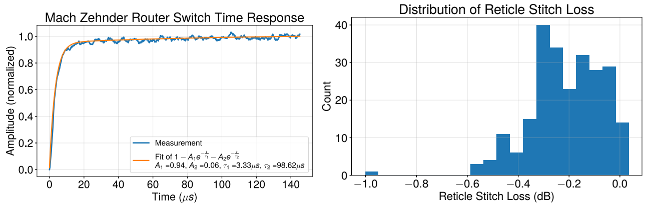
Optical switches. Each Lightpath tile is equipped with four optical switches; each switch has a degree of 1 × 3. We construct these optical switches using Mach-Zehnder Interferometers (MZIs) [57] as shown in Figure 2b. Each switch connects to inter-tile optical waveguides and the three remaining optical switches on the same Lightpath tile. We can program the MZIs to route wavelengths arriving at a tile to one of three neighboring tiles via an optical switch.



(a) Tx/Rx on a Lightpath tile. (b) Optical switches on a Lightpath tile. (c) $2X4$ tiles on a Lightpath wafer.

Figure 2: Figure 2a shows one tile on Lightpath. Each tile has a Tx/Rx block and four switches. The Tx/Rx block (zoomed in) modulates data from chips on to wavelengths of light (λ) in the transmit direction and detects bits from the modulated light in the receive direction. Figure 2b hides the Tx/Rx block to reveal the optical waveguide connectivity between switches and across tiles in Lightpath. The red bar shows optical crossings. Each switch on the tile has a degree of $1X3$ and is constructed using Mach-Zehnder Interferometers (MZIs). Figure 2c shows $2X4$ tiles from a Lightpath wafer. In addition to waveguides connecting switches on a tile (dashed lines), waveguides also connect tiles (solid lines). Optical switches on tiles can configure circuits like the one between tile A and B (red). Optical fibers (solid lines with arrows) connect one Lightpath wafer to others.

This routing behavior is reconfigurable, allowing us to create on-demand optical circuits between GPUs on Lightpath.



(a) Reconfiguration latency. (b) Stitching loss.

Figure 3: We use the setup in Figure 1 and connect two tiles on the Lightpath wafer to a Xilinx FPGA through amplifiers. We use the FPGA to generate traffic that is sent to Tile 1 on Lightpath. Modulators on the tile transmit the traffic to Tile 2. We measure characteristics (*e.g.,* bit error rate) using this transfer. We use an Arduino controller to program the Lightpath device through the JTAG interface [1].

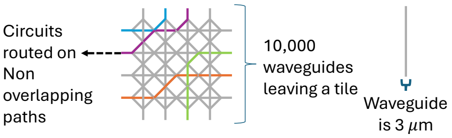


Figure 4: MZI switches and waveguides are arranged in a grid on a tile to allow 10,000 waveguides. Transceivers of the tile are connected to one of these switches.

Chip-to-chip optical circuits. Figure 2c shows an optical circuit constructed on Lightpath by directing signals through a series of horizontal and vertical bus waveguides. A group of MZIs are configured so that a pair of bus waveguides, one from A to B and vice versa, directly connect a transceiver from tile A to a transceiver from tile B. Figure 2c shows 4 waveguides entering a tile for simplicity. In reality, Lightpath can support over 10,000 waveguides per tile since each waveguide and MZI has a pitch of 3μm (Figure 4).

Fiber connectivity between Lightpath wafers. One Lightpath wafer connects to others using attached fibers. With attached fibers, we can cascade several Lightpath wafers to create a rack-scale photonic interconnect. The waveguides leaving a tile at the edge of a server are attached to fibers, enabling circuit switching across servers. Fibers can be attached vertically to the tiles to build 3D topologies.

Microsecond reconfiguration. We fabricated a testbed that demonstrates the performance of optical devices on Lightpath in GlobalFoundries [16] (Figure 1). Using this testbed, we show (Figure 3a) that MZIs of optical switches on Lightpath can be reconfigured within 3.7μs. The fast reconfiguration of optical switches enables the quick establishment of optical circuits between a pair of chips bonded on to Lightpath. By design, optical circuits between chips eliminate contention at intermediate hops on the path.

Measuring signal loss. The circuit from A to B in Figure 3a crosses two tile boundaries. To quantify feasibility of routing within a server, we measured the loss of signal quality at the crossing in Figure 3b. The low-loss (0.25dB) optical crossings enable routing within the same active silicon device layer.

4 Opportunities

All modern multi-accelerator servers (*e.g.,* Nvidia DGX [33], Intel Gaudi [20]) use electrical interconnects (*e.g.,* ICI [23], Nvidia's NVlinks [34], PCIe [32]) to network the on-board accelerators. Larger ML deployments of rack-scale or datacenter-scale, use a network fabric to interconnect multi-accelerator servers. Most cloud providers use the traditional datacenter technology to network multi-accelerator servers [17, 35]. In this design, optical fiber connects multi-accelerator servers in a rack to a top-of-rack packet switch. Racks are connected in a datacenter clos topology that is underpinned by optics but still electrically packet-switched (*e.g.,* leaf-spine architecture).

While these deployments are scalable, ML jobs across multiple racks in this design encounter high communication overheads, reducing the training and inference throughput [36]. Even specialized rack-scale multi-accelerator interconnects, recently announced by Nvidia [51], rely on implementing an ideal contention-free switch to connect up to 72 GPUs [51].

State-of-the-art large ML deployments. Recently, Google has developed an ML supercomputer that uses optics to construct large direct-connect accelerator deployments [23, 28, 60]. The supercomputer has 64 racks, where each rack is a 3D torus. Within each rack, there are 16 multi-accelerator servers, each with 4 TPU chips (*i.e.,* ML accelerators) onboard. The on-board interconnect is electrical. Even within the rack, the bulk of the connectivity is electrical, except for *wrap-around* links that optically connect opposite faces of the rack cubes via optical circuit switches. The resulting TPU rack forms a three-dimensional torus [23]. The optical circuit switches can be programmed to directly connect multiple racks or cubes together into larger tori (Figure 5a).

This topology is designed to work with multi-dimensional bucket ring algorithms [23, 39] for various collective communication primitives [40]. Note that ring-based algorithms require an accelerator to communicate with only two other accelerators at a given time, making communication in a ring on a direct-connect torus congestion free. However, this architecture leads to two issues: (1) direct-connect tori under-utilize the bandwidth of each accelerator in multi-tenant systems and (2) this architecture leads to a large blast radius in case of the failure of a single TPU. This section explores how server-scale multi-accelerator photonic interconnects, like Lightpath, can alleviate these issues. We use Google's TPU cluster [23] in the rest of this section since it is the largest ML accelerator cluster in the cloud built with a reconfigurable fabric tailored for ML jobs. Using Lightpath (§3), the TPUs within a server are connected via waveguides and TPUs across the server are connected with fibers.

4.1 Resolve bandwidth underutilization

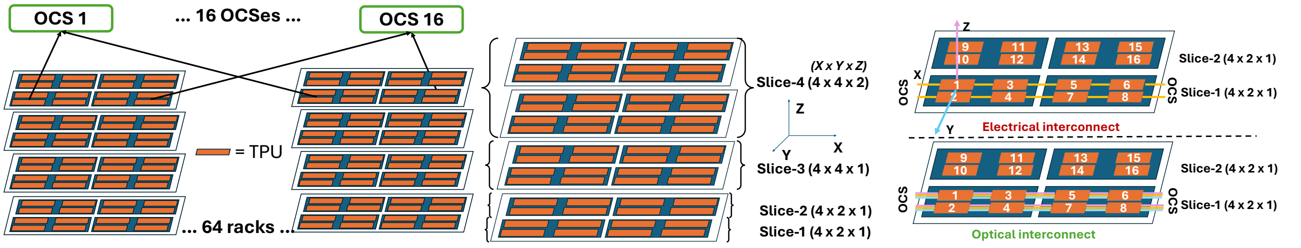
A slice consists of a subset of TPU chips allocated to a single cloud tenant. Typically, slices can only be allocated in regular shapes, forming tori of specific dimensions [3]. Tenants deploy their training and inference jobs on the allocated TPU slice, during which collective communication primitives are executed over the slice torus using the multi-dimensional bucket algorithm [39]. We note that TPU slices allocated to customers or tenants do not always span multiple racks. Most inference workloads need smaller slices as model sizes rarely exceed the memory of all TPUs in a rack. The standard multi-dimensional bucket algorithm sequentially executes data transfers in rings across all the dimensions of the torus. So, in a 3D torus, 3 rings are executed in the order of dimensions XYZ. As a result, connectivity in two of the three dimensions is always underutilized since only one ring is active at a given time. To address this under-utilization, researchers have proposed algorithms that subdivide the data buffer to execute several multidimensional bucket algorithms in different sequences of dimensions, *e.g.,* YZX and ZXY, *simultaneously* such that all the dimensions are utilized throughout the collective [41]. However, this does not offer better performance, as we will discuss later.

Our key observation: bandwidth underutilization. We define congestion in a direct-connect topology as the scenario where multiple transfers occur simultaneously on the same link, similar to the definition in theoretical computer science [26, 38]. A slice optimally utilizes the bandwidth only when it communicates on all three dimensions. Note that due to the design of a torus, this can only happen when a slice spans multiple racks. For instance, in Figure 5b, rings along the Z dimension of all the slices of tenants share the links between servers in the Z dimension which leads to multiple transfers on the same link simultaneously causing congestion. If we avoid the Z dimensional ring in all the slices to prevent congestion, then the bandwidth is underutilized by 33% because the slices have access to only two of the three dimensions. Figure 5c demonstrates how even smaller slices can suffer up-to 66% lower bandwidth. Slice-1 and Slice-2 share both the Y and Z dimensions with other slices and can only execute the X dimensional ring without causing congestion, which leads to suboptimal performance.

Elec. α cost	Optics α cost	Elec. β cost	Optics β cost
$7 \times \alpha$	$7 \times \alpha + r$	$N\left(\frac{8-1}{8}\right) \cdot \beta$	$N\left(\frac{8-1}{8}\right) \cdot \left(\frac{\beta}{3}\right)$

Table 1: ReduceScatter costs of Slice-1, where N is the buffer size. Electrical interconnects induce 3× the β cost due to their inability to fully utilize bandwidth in all dimensions.

Impact of the underutilized bandwidth on $\alpha - \beta$ costs. We use the $\alpha - \beta$ cost model [42] to reason about the cost of collective communication. α is the software overhead of sending data buffers. β is the transmission delay, which is inversely proportional to the bandwidth of a single link of the TPU. We model the reconfiguration of optical components between rounds of the collective with reconfiguration latency r. Since β is several magnitudes of order higher than α, and also the large buffer sizes of most modern ML models, we focus on reducing β cost. The AllReduce bucket algorithm on a D dimensional torus has D ReduceScatter operations followed by D AllGather operations and can reach optimal β cost of $\frac{2N}{D}$ with simultaneous rings in all D dimensions. (The constituent ReduceScatter operation thus meets its β-cost lower bound of $\sim \frac{N}{D}$.) However, Table 1 shows that the cost of one ReduceScatter on Slice-1 is proportional to 3 times the optimal ($\frac{1}{bandwidth} \cdot \frac{1}{D}$), because the slice is utilizing the bandwidth of only one dimension of the torus.



(a) TPUv4 cluster of 4096 chips. (b) Slices smaller than a (4x4x4) rack. (c) Redirecting bandwidth with optics.

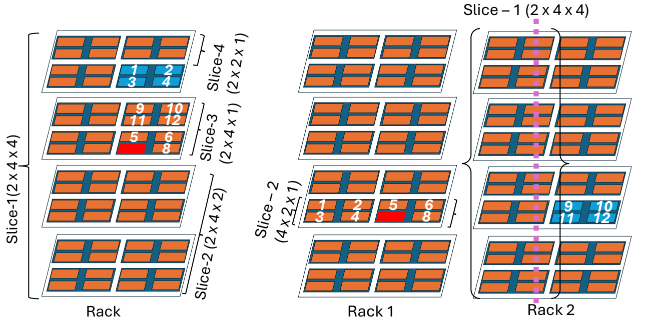
Figure 5: 5a: A TPU cube has TPUs electrically interconnected in a 4x4x4 3D torus. TPUs on every face of the cube are connected to OCSes which can be reconfigured to build larger 3D tori with multiple cubes. 5b shows a rack with multiple slices. Slices smaller than a rack are unable to fully utilize the bandwidth on a chip in the electrically interconnected racks. Optically interconnected racks can leverage reconfigurability to maximally utilize the bandwidth. 5c shows that electrical interconnects underutilize bandwidth in slices smaller than a rack and reconfigurable optical interconnects like Lightpath maximize the bandwidth utilization for the same slices.

Opportunity: redirect GPU bandwidth on demand. Server-scale optical interconnects can solve this problem by fully utilizing the bandwidth of all the torus dimensions. The output of I/O ports of the TPU chip along different dimensions can be redirected to one dimension by dynamically programming the MZI switches (§3). The β cost of a single torus bucket algorithm with redirected bandwidth is the same as a executing several torus bucket algorithms simultaneously. Specifically, given a buffer size of N, D dimensions, and total bandwidth B, the bandwidth per dimension is $\frac{B}{D}$. Dividing N to run simultaneous bucket algorithms across all D dimensions yields a cost of $\frac{N}{D} \cdot \frac{D}{B} = \frac{N}{B}$ since each dimension has $\frac{B}{D}$ bandwidth. This is the same as running the algorithm once, using all the bandwidth in each step (only feasible with Lightpath or other photonic interconnects), also with a β cost of $\frac{N}{B}$. The additional redirected wavelengths along a given dimension are all accommodated on different waveguides which avoids congestion within waveguides.

In Figure 5c, we program the MZI switches on Slice-1 to redirect all of their bandwidth along the ring in the X dimension and execute one instance of the algorithm. The algorithm's β cost using reconfigurable optical interconnects is 3× lower compared to using static electrical interconnects because we redirect the unused bandwidth to the X dimension. The main tradeoff of reconfiguration is a constant r time units before the start of a ring which is 3.7μs in Lightpath. Similarly, in Slice-3, we redirect the bandwidth of the Z dimension to rings along X and Y dimensions. The β cost for Slice-3 in Table 2 is 1.5× higher for electrical interconnects.

4.2 Shrink the blast radius of failures

In this section, we argue that reconfiguration reduces the blast radius of failures. Replacing a failed chip with a chip in the same or a different rack causes congestion in the electrical torus. The current policy avoids this by handling faults at rack granularity [60] leading to a large blast radius.



(a) Single rack (b) Multi-rack

Figure 6: In 6a, replacing the failed chip (red) with one of the free chips (blue) is impossible without congestion. In 6b, replacing the failed chip (red) with a free chip in rack 2 causes congestion on the Y dimension (purple line).

Congestion within a rack. Figure 6a shows a single rack with a failed TPU in Slice-3 (marked in red) and free TPUs in the same rack (shown in blue). To utilize a free TPU, we must connect the failed chip's neighbors in the X and Y dimension rings to one of the free TPUs. We need to connect TPUs 5 and 9 to a free chip to complete the ring for 5, 9 and 11 along the Y dimension. While reaching any free chip from TPU 5 without causing congestion is straightforward, doing the same from

Elec. α cost	Optics α cost	Elec. β cost	Optics β cost
$3 \times \alpha$	$3 \times \alpha + r$	$N \cdot \left(\frac{4-1}{4}\right) \cdot \left(\frac{\beta}{2}\right)$	$N \cdot \left(\frac{4-1}{4}\right) \cdot \left(\frac{\beta}{3}\right)$
$3 \times \alpha$	$3 \times \alpha + r$	$\frac{N}{4} \cdot \left(\frac{4-1}{4}\right) \cdot \left(\frac{\beta}{2}\right)$	$\frac{N}{4} \cdot \left(\frac{4-1}{4}\right) \cdot \left(\frac{\beta}{3}\right)$

Table 2: ReduceScatter $\alpha - \beta$ costs of Slice-3 with $D = 2$, forming 4 rings of size 4 each in X and Y dimensions. The bucket algorithm executes in two stages. It first executes rings in the X dimension with a buffer size of N with costs shown in the first row. After these rings complete, the algorithm executes rings in the Y dimension with a buffer size of $\frac{N}{4}$ whose costs are shown in the second row.

TPU 9 without congestion is impossible. Any chosen free chip can only be reached from TPU 9 through TPUs 5, 6, or 8 and then taking the link between servers in the Z dimension. This constraint arises because any alternative path would induce congestion on Slice-4. If the path reaches 5 or 6, there is congestion on the ring through TPUs 5, 11, and 9. If the path reaches TPU 8, congestion impacts both rings, through TPUs 5, 11, and 9 and through 8, 6, 10, and 12.

Congestion across racks. In Figure 6b, Slice-2 in rack 1, with 8 TPUs, has a failed TPU. While rack 1 has no available chips, rack 2 has 4 free chips. So, TPU 4 should connect to a free chip in rack 2. However, connecting TPU 4 through the X or Y dimensions would cause congestion in Slice-2 or other slices of rack 1, as all orange TPUs are allocated to slices. Thus, TPU 4 must use the Z dimension to reach rack 2 via the OCS. The network topology limits TPU 4's communication to TPUs along the purple line. Since Slice-1's 3D torus bucket ring algorithms already use this line's Y dimension in rack 2, any new traffic will cause congestion.

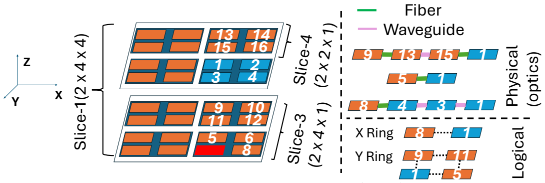


Figure 7: After the failure of TPU 7 (red) in Slice-3, the Y-dimension ring is broken since there is no TPU between 9 and 5. The X-dimension ring is broken since there is no TPU connected to 8. Optical reconfiguration can connect a free TPU (1) to TPUs 9, 5 and 8 to fix the broken rings.

Opportunity: Establish non-overlapping optical circuits to replace failed chips. The root cause of congestion is multiple paths sharing a single electrical link. Increasing the number of electrical links between TPUs reduces interconnect congestion but introduces on-chip congestion due to the absence of electrical switching on chips. Traffic not destined for a TPU must be forwarded, consuming its bandwidth. The network will be congestion-free if each path has a dedicated end-to-end circuit. An optical interconnect like LIGHTPATH provides thousands of waveguides between chips and 10s of fibers across servers. We propose to leverage these physical links to establish non-overlapping circuits. We can program MZI switches in the rack on each TPU (§3) to establish a dedicated circuit between TPUs. In Figure 7, we show how optical circuits can repair the broken rings along the X and Y dimensions in Slice-3 from Figure 6a by replacing the failed TPU with TPU 1. The neighbors of the failed TPU in the broken rings are now optically connected to TPU 1 with end-to-end optical circuits. We place these optical circuits on separate waveguides and fibers to avoid congestion and achieve optimal performance.

5 Challenges

While multi-accelerator photonic interconnects are promising, we must solve many technical challenges to realize them.

Exploding paths. Each tile can contain tens of thousands of waveguides, providing a circuit entering a tile with thousands of possible paths. Optimizing these paths for all circuits is a scalability challenge. While simple collective operations, such as those using ring ALLREDUCE where each accelerator communicates with only two others, are relatively straightforward, handling all-to-all traffic is much more complex.

Decentralized algorithms. Another critical challenge is developing algorithms for traffic patterns that are outside known collective operations, such as those required for Mixture of Experts [43] (MoE) inference. MoE inference relies on a runtime gating function, necessitating dynamic programming of circuits. A naive solution would rely on a centralized controller tracking the state of every waveguide to avoid overlaps. However, this approach does not scale well when dealing with hundreds of accelerators, highlighting the need for decentralized algorithms to manage dynamic traffic.

Minimizing fiber requirement for fault tolerance. Fault-tolerant circuit pathfinding must intelligently manage the addition of fibers, aiming to minimize fiber usage while effectively managing faults. This requires sophisticated algorithms capable of dynamically reconfiguring the network in real-time, ensuring continued operation despite faults without excessively increasing infrastructure requirements.

6 Related Work

Recent work leverages optical components for failure resilience [59], capacity augmentation [47, 48] and long-running bulk data transfers [22]. There has also been growing commercial interest in making optics an active part of routing in datacenter interconnects. For instance, Google has recently replaced the spine layer packet switches in their datacenter interconnects with optical circuit switches [23, 35, 46]. Researchers have used optics for reconfiguring the datacenter interconnect [18, 30, 49, 53]. Researchers have used silicon photonic interconnects for improving the performance of machine learning workloads [24, 37, 50, 55, 56]. This work focuses on slow and infrequent reconfiguration of the interconnect, called topology engineering.

Acknowledgements. We thank anonymous reviewers for their feedback. This research is supported by a gift to the Cisco-Cornell Bowers CIS Strategic Partnership and NSF Award #2444537. AVK is supported by the Cornell Bowers CIS-LinkedIn Grant, and AD is supported by the NSF Graduate Research Fellowship.

Ethics. This work does not raise any ethical concerns.

References

[1] 2010. IEEE Standard for Reduced-Pin and Enhanced-Functionality Test Access Port and Boundary-Scan Architecture. *IEEE Std 1149.7-2009* (2010), 1–985. https://doi.org/10.1109/IEEESTD.2010.5412866

[2] 2021. Scaling Distributed Machine Learning with In-Network Aggregation. In *18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21)*. USENIX Association, 785–808. https://www.usenix.org/conference/nsdi21/presentation/sapio

[3] 2023. TPU v4 Documentation. https://cloud.google.com/tpu/docs/v4. Accessed on 2024-05-29.

[4] Saksham Agarwal, Arvind Krishnamurthy, and Rachit Agarwal. 2023. Host Congestion Control. In *Proceedings of the ACM SIGCOMM 2023 Conference* (New York, NY, USA) *(ACM SIGCOMM '23)*. Association for Computing Machinery, New York, NY, USA, 275–287. https://doi.org/10.1145/3603269.3604878

[5] Mohammad Al-Fares, Alexander Loukissas, and Amin Vahdat. 2008. A scalable, commodity data center network architecture. In *Proceedings of the ACM SIGCOMM 2008 Conference on Data Communication* (Seattle, WA, USA) *(SIGCOMM '08)*. Association for Computing Machinery, New York, NY, USA, 63–74. https://doi.org/10.1145/1402958.1402967

[6] Paul Barford, Cara Caida, David Choffnes, Ramakrishnan Durairajan, and Walter Willinger. 2015. InterTubes: A Study of the US Long-haul Fiber-optic Infrastructure. In *Proceedings of the ACM SIGCOMM Conference*. ACM, London, United Kingdom, 565–578. https://doi.org/10.1145/2785956.2787481

[7] Cerebras 2021. The future of AI is Wafer-Scale. https://www.cerebras.net/product-chip/.

[8] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860* (2019).

[9] Wei Deng, Junwei Pan, Tian Zhou, Deguang Kong, Aaron Flores, and Guang Lin. 2021. DeepLight: Deep Lightweight Feature Interactions for Accelerating CTR Predictions in Ad Serving. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining* (Virtual Event, Israel) *(WSDM '21)*. Association for Computing Machinery, New York, NY, USA, 922–930. https://doi.org/10.1145/3437963.3441727

[10] Jack Dongarra et al. 2013. MPI: A message-passing interface standard version 3.0. *High Performance Computing Center Stuttgart (HLRS)* 2, 5 (2013), 32.

[11] Nathan Farrington, George Porter, Sivasankar Radhakrishnan, Hamid Hajabdolali Bazzaz, Vikram Subramanya, Yeshaiahu Fainman, George Papen, and Amin Vahdat. 2010. Helios: A Hybrid Electrical/Optical Switch Architecture for Modular Data Centers. In *Proceedings of the ACM SIGCOMM 2010 Conference* (New Delhi, India) *(SIGCOMM '10)*. Association for Computing Machinery, New York, NY, USA, 339–350. https://doi.org/10.1145/1851182.1851223

[12] William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch Transformers: Scaling to Trillion Parameter Models with Simple and Efficient Sparsity. *CoRR* abs/2101.03961 (2021). arXiv:2101.03961 https://arxiv.org/abs/2101.03961

[13] Fierce Electronics. (Accessed on 2023-05-26). ChatGPT runs 10K Nvidia training GPUs with potential for thousands more. https://www.fierceelectronics.com/sensors/chatgpt-runs-10k-nvidia-training-gpus-potential-thousands-more.

[14] Nadeen Gebara, Manya Ghobadi, and Paolo Costa. 2021. In-network Aggregation for Shared Machine Learning Clusters. In *Proceedings of Machine Learning and Systems*, A. Smola, A. Dimakis, and I. Stoica (Eds.), Vol. 3. 829–844. https://proceedings.mlsys.org/paper/2021/file/eae27d77ca20db309e056e3d2dcd7d69-Paper.pdf

[15] 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* (Florianopolis, Brazil) *(SIGCOMM '16)*. Association for Computing Machinery, New York, NY, USA, 216–229. https://doi.org/10.1145/2934872.2934911

[16] Global Foundries. (Accessed on 2023-06-16). Global Foundries. https://gf.com/.

[17] Albert Greenberg, James R. Hamilton, Navendu Jain, Srikanth Kandula, Changhoon Kim, Parantap Lahiri, David A. Maltz, Parveen Patel, and Sudipta Sengupta. 2009. VL2: A Scalable and Flexible Data Center Network. In *Proceedings of the ACM SIGCOMM 2009 Conference on Data Communication* (Barcelona, Spain) *(SIGCOMM '09)*. Association for Computing Machinery, New York, NY, USA, 51–62. https://doi.org/10.1145/1592568.1592576

[18] 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* (Chicago, Illinois, USA) *(SIGCOMM '14)*. Association for Computing Machinery, New York, NY, USA, 319–330. https://doi.org/10.1145/2619239.2626328

[19] HotChips 34. (Accessed on 2023-05-26). Passage—A Wafer-Scale, Programmable Photonic Communication Substrate. https://hc34.hotchips.org/assets/program/conference/day1.

[20] Intel Gaudi AI accelerator 2021. Intel Gaudi AI accelerator. https://habana.ai/products/gaudi/.

[21] Peter B. Griffin James D. Plummer. 2023. *Integrated Circuit Fabrication Science and Technology*. Cambridge University Press, Cambridge, United Kingdom.

[22] Xin Jin, Yiran Li, Da Wei, Siming Li, Jie Gao, Lei Xu, Guangzhi Li, Wei Xu, and Jennifer Rexford. 2016. Optimizing Bulk Transfers with Software-Defined Optical WAN. In *Proceedings of the 2016 ACM SIGCOMM Conference* (Florianopolis, Brazil) *(SIGCOMM '16)*. Association for Computing Machinery, New York, NY, USA, 87–100. https://doi.org/10.1145/2934872.2934904

[23] Norman P. Jouppi, George Kurian, Sheng Li, Peter Ma, Rahul Nagarajan, Lifeng Nai, Nishant Patil, Suvinay Subramanian, Andy Swing, Brian Towles, Cliff Young, Xiang Zhou, Zongwei Zhou, and David Patterson. 2023. TPU v4: An Optically Reconfigurable Supercomputer for Machine Learning with Hardware Support for Embeddings. arXiv:2304.01433 [cs.AR]

[24] Mehrdad Khani, Manya Ghobadi, Mohammad Alizadeh, Ziyi Zhu, Madeleine Glick, Keren Bergman, Amin Vahdat, Benjamin Klenk, and Eiman Ebrahimi. [n. d.]. SiP-ML: High-Bandwidth Optical Network Interconnects for Machine Learning Training. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference*.

[25] ChonLam Lao, Yanfang Le, Kshiteej Mahajan, Yixi Chen, Wenfei Wu, Aditya Akella, and Michael Swift. 2021. ATP: In-network Aggregation for Multi-tenant Learning. In *18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21)*. USENIX Association, 741–761. https://www.usenix.org/conference/nsdi21/presentation/lao

[26] Frank Thomson Leighton, Bruce M. Maggs, and Satish Rao. 1994. Packet Routing and Job-Shop Scheduling in (Congestion + Dilation) Steps. *Combinatorica* 14, 2 (1994), 167–186. https://doi.org/10.1007/BF01215349

[27] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. 2020. GShard: Scaling Giant Models with Conditional Computation and Automatic Sharding. *CoRR* abs/2006.16668 (2020). arXiv:2006.16668 https://arxiv.org/abs/2006.16668

[28] Hong Liu, Ryohei Urata, Kevin Yasumura, Xiang Zhou, Roy Bannon, Jill Berger, Pedram Dashti, Norm Jouppi, Cedric Lam, Sheng Li, Erji Mao, Daniel Nelson, George Papen, Mukarram Tariq, and Amin Vahdat. 2023. Lightwave Fabrics: At-Scale Optical Circuit Switching for Datacenter and Machine Learning Systems. In *Proceedings of the ACM SIGCOMM 2023 Conference (ACM SIGCOMM '23)*. Association for Computing Machinery, New York, NY, USA, 499–515. https://doi.org/10.1145/3603269.3604836

[29] William M. Mellette, Rajdeep Das, Yibo Guo, Rob McGuinness, Alex C. Snoeren, and George Porter. 2020. Expanding across time to deliver bandwidth efficiency and low latency. In *17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20)*. USENIX Association, Santa Clara, CA, 1–18. https://www.usenix.org/conference/nsdi20/presentation/mellette

[30] William M. Mellette, Rob McGuinness, Arjun Roy, Alex Forencich, George Papen, Alex C. Snoeren, and George Porter. 2017. RotorNet: A Scalable, Low-Complexity, Optical Datacenter Network. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication* (Los Angeles, CA, USA) *(SIGCOMM '17)*. Association for Computing Machinery, New York, NY, USA, 267–280. https://doi.org/10.1145/3098822.3098838

[31] MT-NLG 2021. Using DeepSpeed and Megatron to Train Megatron-Turing NLG 530B, the World's Largest and Most Powerful Generative Language Model. https://www.microsoft.com/en-us/research/blog/using-deepspeed-and-megatron-to-train-megatron-turing-nlg-530b-the-worlds-largest-and-most-powerful-generative-language-model/. Accessed October 2021.

[32] Rolf Neugebauer, Gianni Antichi, José Fernando Zazo, Yury Audzevich, Sergio López-Buedo, and Andrew W. Moore. 2018. Understanding PCIe Performance for End Host Networking. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication* (Budapest, Hungary) *(SIGCOMM '18)*. Association for Computing Machinery, New York, NY, USA, 327–341. https://doi.org/10.1145/3230543.3230560

[33] Nvidia DGX Systems 2021. Nvidia DGX Systems. https://www.nvidia.com/en-us/data-center/dgx-systems/.

[34] Nvidia NVLink 2021. Nvidia NVLink and NVSwitch. https://www.nvidia.com/en-us/data-center/nvlink/.

[35] Leon Poutievski, Omid Mashayekhi, Joon Ong, Arjun Singh, Mukarram Tariq, Rui Wang, Jianan Zhang, Virginia Beauregard, Patrick Conner, Steve Gribble, Rishi Kapoor, Stephen Kratzer, Nanfang Li, Hong Liu, Karthik Nagaraj, Jason Ornstein, Samir Sawhney, Ryohei Urata, Lorenzo Vicisano, Kevin Yasumura, Shidong Zhang, Junlan Zhou, and Amin Vahdat. 2022. Jupiter evolving: transforming google's datacenter network via optical circuit switches and software-defined networking. In *Proceedings of the ACM SIGCOMM 2022 Conference* (Amsterdam, Netherlands) *(SIGCOMM '22)*. Association for Computing Machinery, New York, NY, USA, 66–85. https://doi.org/10.1145/3544216.3544265

[36] Sudarsanan Rajasekaran, Manya Ghobadi, Gautam Kumar, and Aditya Akella. 2022. Congestion control in machine learning clusters. In *Proceedings of the 21st ACM Workshop on Hot Topics in Networks* (Austin, Texas) *(HotNets '22)*. Association for Computing Machinery, New York, NY, USA, 235–242. https://doi.org/10.1145/3563766.3564115

[37] Anthony Rizzo and Keren Bergman. 2022. Realizing Pb/s IO with Silicon Photonic Chiplets, In Optica Advanced Photonics Congress 2022. *Optica Advanced Photonics Congress 2022*, NeTu1D.1. https://doi.org/10.1364/NETWORKS.2022.NeTu1D.1

[38] Thomas Rothvoss. 2012. A simpler proof for O(congestion + dilation) packet routing. arXiv:1206.3718 [cs.DS]

[39] Paul Sack and William Gropp. 2015. Collective Algorithms for Multiported Torus Networks. *ACM Trans. Parallel Comput.* 1, 2, Article 12 (feb 2015), 33 pages. https://doi.org/10.1145/2686882

[40] Paul Sack and William Gropp. 2015. Collective Algorithms for Multiported Torus Networks. *ACM Trans. Parallel Comput.* 1, 2, Article 12 (feb 2015), 33 pages. https://doi.org/10.1145/2686882

[41] Daniele De Sensi, Tommaso Bonato, David Saam, and Torsten Hoefler. 2024. Swing: Short-cutting Rings for Higher Bandwidth Allreduce. In *21st USENIX Symposium on Networked Systems Design and Implementation (NSDI 24)*. USENIX Association, Santa Clara, CA, 1445–1462. https://www.usenix.org/conference/nsdi24/presentation/de-sensi

[42] Aashaka Shah, Vijay Chidambaram, Meghan Cowan, Saeed Maleki, Madan Musuvathi, Todd Mytkowicz, Jacob Nelson, Olli Saarikivi, and Rachee Singh. 2023. TACCL: Guiding Collective Algorithm Synthesis using Communication Sketches. In *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*. USENIX Association, Boston, MA, 593–612. https://www.usenix.org/conference/nsdi23/presentation/shah

[43] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. 2017. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In *International Conference on Learning Representations*. https://openreview.net/forum?id=B1ckMDqlg

[44] Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2019. Megatron-LM: Training Multi-Billion Parameter Language Models Using Model Parallelism. *CoRR* abs/1909.08053 (2019). arXiv:1909.08053 http://arxiv.org/abs/1909.08053

[45] Jane M. Simmons. 2008. *Optical Network Design and Planning* (2nd ed.). Springer, New York.

[46] Arjun Singh, Joon Ong, Amit Agarwal, Glen Anderson, Ashby Armistead, Roy Bannon, Seb Boving, Gaurav Desai, Bob Felderman, Paulie Germano, Anand Kanagala, Jeff Provost, Jason Simmons, Eiichi Tanda, Jim Wanderer, Urs Hölzle, Stephen Stuart, and Amin Vahdat. 2015. Jupiter Rising: A Decade of Clos Topologies and Centralized Control in Google's Datacenter Network. *SIGCOMM Comput. Commun. Rev.* 45, 4 (aug 2015), 183–197. https://doi.org/10.1145/2829988.2787508

[47] Rachee Singh, Nikolaj Bjorner, Sharon Shoham, Yawei Yin, John Arnold, and Jamie Gaudette. 2021. Cost-Effective Capacity Provisioning in Wide Area Networks with Shoofly. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference* (Virtual Event, USA) *(SIGCOMM '21)*. Association for Computing Machinery, New York, NY, USA, 534–546. https://doi.org/10.1145/3452296.3472895

[48] Rachee Singh, Manya Ghobadi, Klaus-Tycho Foerster, Mark Filer, and Phillipa Gill. 2018. RADWAN: Rate Adaptive Wide Area Network. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication* (Budapest, Hungary) *(SIGCOMM '18)*. Association for Computing Machinery, New York, NY, USA, 547–560. https://doi.org/10.1145/3230543.3230570

[49] Min Yee Teh, Zhenguo Wu, Madeleine Glick, Sebastien Rumley, Manya Ghobadi, and Keren Bergman. 2022. Performance trade-offs in reconfigurable networks for HPC. *Journal of Optical Communications and Networking* 14, 6 (2022), 454–468. https://doi.org/10.1364/JOCN.451760

[50] Min Yee Teh, Shizhen Zhao, Peirui Cao, and Keren Bergman. 2023. Enabling Quasi-Static Reconfigurable Networks With Robust Topology Engineering. *IEEE/ACM Transactions on Networking* 31, 3 (2023), 1056–1070. https://doi.org/10.1109/TNET.2022.3210534

[51] The Next Platform. 2024. How NVIDIA Blackwell Systems Attack 1-Trillion-Parameter AI Models. *The Next Platform* (2024). https://www.nextplatform.com/2024/03/19/how-nvidia-blackwell-systems-attack-1-trillion-parameter-ai-models/ Accessed: 2024-06-22.

[52] Guohui Wang, David G. Andersen, Michael Kaminsky, Konstantina Papagiannaki, T.S. Eugene Ng, Michael Kozuch, and Michael Ryan. 2010. C-Through: Part-Time Optics in Data Centers. In *Proceedings of*

the ACM SIGCOMM 2010 Conference (New Delhi, India) *(SIGCOMM '10)*. Association for Computing Machinery, New York, NY, USA, 327–338. https://doi.org/10.1145/1851182.1851222

[53] Weiyang Wang, Moein Khazraee, Zhizhen Zhong, Manya Ghobadi, Zhihao Jia, Dheevatsa Mudigere, Ying Zhang, and Anthony Kewitsch. 2023. TopoOpt: Co-optimizing Network Topology and Parallelization Strategy for Distributed Training Jobs. In *20th USENIX Symposium on Networked Systems Design and Implementation (NSDI 23)*. USENIX Association, Boston, MA, 739–767. https://www.usenix.org/conference/nsdi23/presentation/wang-weiyang

[54] A. Willner. 2019. *Optical Fiber Telecommunications*. Number v. 11. Elsevier Science. https://books.google.com/books?id=A5W2DwAAQBAJ

[55] Zhenguo Wu, Liang Yuan Dai, Ziyi Zhu, Asher Novick, Madeleine Glick, and Keren Bergman. 2023. SiP Architecture For Accelerating Collective Communication in Distributed Deep Learning. In *2023 Optical Fiber Communications Conference and Exhibition (OFC)*. 1–3. https://doi.org/10.1364/OFC.2023.W1G.1

[56] Yawei Yin, Mingyang Zhang, Zuqing Zhu, and S. J. B. Yoo. 2013. Fragmentation-Aware Routing, Modulation and Spectrum Assignment Algorithms in Elastic Optical Networks, In Optical Fiber Communication Conference/National Fiber Optic Engineers Conference 2013. *Optical Fiber Communication Conference/National Fiber Optic Engineers Conference 2013*, OW3A.5. https://doi.org/10.1364/OFC.2013.OW3A.5

[57] K. P. Zetie, S. F. Adams, and R. M. Tocknell. 2000. How Does a Mach-Zehnder Interferometer Work? https://www.cs.princeton.edu/courses/archive/fall06/cos576/papers/zetie_et_al_mach_zehnder00.pdf. Accessed on 2024-06-01.

[58] Liangyu Zhao, Saeed Maleki, Ziyue Yang, Hossein Pourreza, Aashaka Shah, Changho Hwang, and Arvind Krishnamurthy. 2024. ForestColl: Efficient Collective Communications on Heterogeneous Network Fabrics. arXiv:2402.06787 [cs.NI]

[59] Zhizhen Zhong, Manya Ghobadi, Alaa Khaddaj, Jonathan Leach, Yiting Xia, and Ying Zhang. 2021. ARROW: Restoration-Aware Traffic Engineering. In *Proceedings of the 2021 ACM SIGCOMM 2021 Conference* (Virtual Event, USA) *(SIGCOMM '21)*. Association for Computing Machinery, New York, NY, USA, 560–579. https://doi.org/10.1145/3452296.3472921

[60] Yazhou Zu, Alireza Ghaffarkhah, Hoang-Vu Dang, Brian Towles, Steven Hand, Safeen Huda, Adekunle Bello, Alexander Kolbasov, Arash Rezaei, Dayou Du, Steve Lacy, Hang Wang, Aaron Wisner, Chris Lewis, and Henri Bahini. 2024. Resiliency at Scale: Managing Google's TPUv4 Machine Learning Supercomputer. In *21st USENIX Symposium on Networked Systems Design and Implementation (NSDI 24)*. USENIX Association, Santa Clara, CA, 761–774. https://www.usenix.org/conference/nsdi24/presentation/zu