λ^{group}: Using Optics to Take Group Data Delivery in the Datacenter to the Next Degree

Howard Wang*, Yiting Xia†, Keren Bergman*, T. S. Eugene Ng†,
Kunwadee Sripanidkulchai§
*Columbia University, †Rice University, §NECTEC Thailand

ABSTRACT

The increasing number of datacenter applications with heavy one-to-many communications has raised the need for an efficient group data delivery solution. This paper presents an unconventional clean-slate architecture called λ^{group} that uses optical networking technologies to enable ultra-fast, energy-efficient, low cost, and highly reliable group data delivery in the datacenter. By replicating data purely optically, λ^{group} enables reliable one-to-many communication independent of modulated data rate, while simultaneously being energy-efficient. λ^{group} makes use of software defined network switches to facilitate direct interactions between applications and the network, eliminating the need for conventional multicast routing and group management protocols without sacrificing link-stress or latency performance. From an end-to-end perspective, λ^{group} provides predictable latency and near-zero packet loss between application end points, allowing even simple end-to-end flow control and packet loss recovery mechanisms to work well. We implement a λ^{group} prototype to measure its physical layer characteristics and end-to-end performance benefits to applications. Extensive simulations using synthetic traffic show λ^{group} provides an order of magnitude performance improvement relative to a number of existing alternative group data delivery approaches.

1. INTRODUCTION

There are many application scenarios in which a block of data needs to be delivered reliably over a network to multiple receivers. This operation is termed Reliable Group Data Delivery (RGDD) in the literature. Enabling efficient high-volume RGDD in the modern datacenter environment is an especially important problem because high-volume RGDD is required by many fundamental system operations and datacenter management tasks such as distributed file system data replication [10], database replication [26], parallel database relational join operation [15], iterative MapReduce data analytics [21], virtual machine cluster provisioning [14], and system software updates [2].

RGDD requires data to be duplicated for each receiver. In conventional solutions, data duplication happens either in the network layer on routers and switches, or in the application layer on end hosts. There is in fact a third alternative that has not been explored by existing solutions: data duplication can happen in the physical layer on photonic devices when data is transmitted optically.

Optical data duplication: How and why. The use of optical data transceivers is already common place in datacenter networks where link bandwidth exceeds 10 Gb/s. Once data is transmitted optically, an Optical Data Duplication Device (ODDD), e.g. an optical power splitter, that is inserted in-line can physically duplicate the data signal multiple times (analogous to a SLR camera prism splitting a light beam), effectively creating copies of the data on the fly. One application of this technique is in passive optical networks (PONs) (e.g. fiber-to-the-home installations), where the data signal from one fiber is split to reach multiple homes in an area. Performing data duplication in the optical domain has three key desirable properties:

- Data rate transparency. Data is duplicated at line rate on the fly regardless of the modulation speed. It does not matter whether the data is transmitted at 10 Gb/s, 40 Gb/s, 100 Gb/s or beyond; optical data duplication is thus *future proof*.
- Decouples reliability from data rate. Regardless of data rate, high data reliability is achieved as long as the duplicated data signal has enough remaining power for a transceiver to decode. This can be ensured by limiting the number of duplications without signal amplification.
- Low power consumption. ODDDs can be passive, thus drawing no power. Based on today's transceiver technologies, in theory, a data signal sent at the default power level can be optically duplicated hundreds of times without significantly increasing the bit-error rate. Optical amplification that consumes relatively little power (i.e., on the order of tens of Watts) can further raise this limit.

In contrast, as data rate increases, reliably duplicating data in network routers, switches or end hosts will

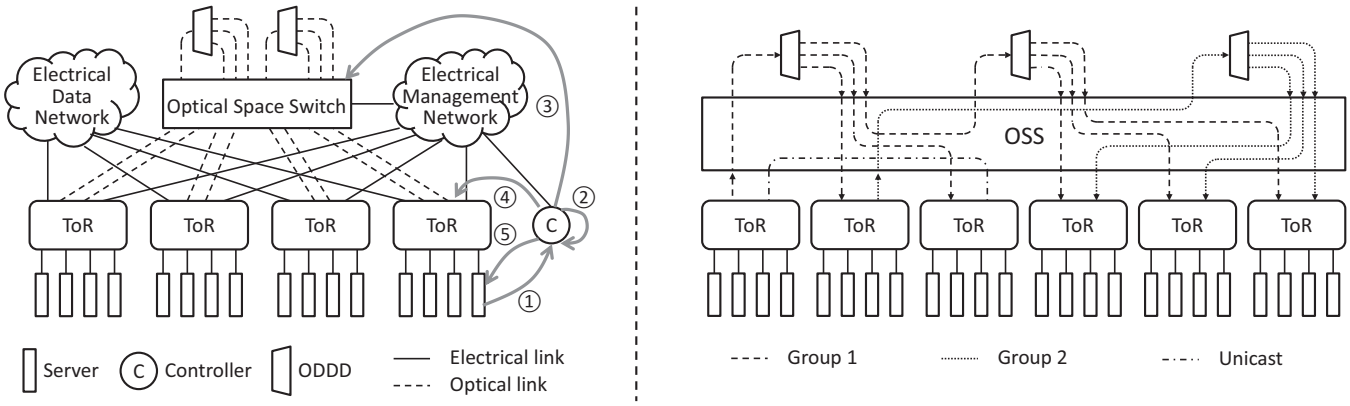

Figure 1: λ^{group} overview: architecture (left), optical network interconnections (right)

require ever faster electronics and more energy.

Challenges. Extrapolating from these properties, it is not hard to see that optical data duplication has the unique potential to enable fast, reliable, and energy efficient group data delivery in the datacenter. However, duplicating data in the physical layer is a highly unconventional approach that does not immediately fit in with the existing assumptions in the datacenter network stack. Specifically, when an optical transmitter is simultaneously connected to multiple receivers through an ODDD, data link negotiation protocols that assume point-to-point connections will not work. Traditional multicast routing and group membership protocols will not work either since the ODDD inserted into fiber-optic links are invisible to the network layer protocols. Application layer overlay protocols also have no way of leveraging such physical layer capabilities. It turns out the data link issue can be addressed fairly easily by designing the ODDD to allow only one receiver port's signal to propagate back to the sender port. However, the other issues remain fundamental. Furthermore, an ODDD can only statically duplicate data from one input port to a fixed number of output ports, and yet clearly a solution must enable the dynamic use of ODDDs to interconnect different senders and receivers at different times to be useful and cost effective.

λ^{group}: A clean-slate approach. To address these challenges and realize the benefits of using optics, we propose a clean-slate system architecture called λ^{group}. Figure 1 shows a schematic of λ^{group}. While the design details are given in Section 2, here we highlight a few key design decisions. First, λ^{group} allows applications to directly request optical resources for RGDD through an API exposed by a λ^{group} controller. Applications convey group membership directly to the controller, eliminating the need for a distributed group membership management protocol. Applications also convey data volume, if known, to the controller, allowing for intelligent resource allocation. Second, for backward compatibility, familiar multicast IP addresses are still used for group identification in requests and data packets. Third, λ^{group} leverages software-defined networking (SDN) switches, such as OpenFlow, as top-of-rack (ToR) switches. Group forwarding states for an RGDD task exist only on ToR switches and are directly configured by the controller, eliminating the need for a multicast routing protocol. Fourth, to keep the system simple, λ^{group} is deployed to serve traffic within a datacenter region. With today's available technologies, a region can cover on the order of hundreds of racks supporting thousands of servers. A RGDD task that spans multiple regions should be decomposed into multiple intra-region RGDD sessions at the application layer.[1] Fifth, the communication capability provided by λ^{group}, although highly reliable, remains best-effort. Reliability, congestion control, and flow control issues are handled by end-to-end mechanisms as usual, keeping the network simple. Sixth, to realize dynamic use of ODDDs, λ^{group} leverages an optical space switch (OSS) as a reconfigurable connectivity fabric for interconnecting ODDD and ToR switches as needed.

Advantages. The advantages of λ^{group} over the alternatives are multi-fold. In addition to the aforementioned rate transparency, reliability and energy advantages of using ODDDs, λ^{group} achieves optimal network link stress and optimal end-to-end delay (i.e. the same advantages of network layer multicast), but also greatly eases end-to-end reliability, flow control, and congestion control (i.e. the same advantages of application overlay). Because the data path provided by λ^{group} is highly reliable and has highly predictable bandwidth and delay, as we will experimentally show in Section 4, even simple protocols are able to reach high stable throughput. λ^{group} is also cost effective compared to electronic switching-based alternatives, especially since the optical components are rate transparent and future proof.

[1]How to optically enable RGDD across regions is outside the scope of this paper.

One concern is whether λ^{group} can scale sufficiently. To address this, we experimentally show that ODDDs can potentially scale to cover hundreds of racks even without using optical amplification. Last but not least, the performance advantage of λ^{group} is very large. In Section 5, extensive simulations using synthetic traffic show λ^{group} provides an order of magnitude performance improvement relative to a number of existing alternative RGDD approaches.

The rest of this paper is organized as follows: Section 2 presents design details; Section 3 analyzes λ^{group}'s advantages; Section 4 presents our experimental testbed, as well as results from end-to-end experiments and system measurements; Section 5 presents simulations that compare λ^{group} against alternatives; Section 6 presents related work; we conclude in Section 7.

2. SYSTEM DESIGN

2.1 Optical Data Duplication

Optical data duplication, otherwise known as an optical multicast, can be realized as a straightforward physical layer operation in photonics. Using a variety of techniques that exploit fundamental properties of the optical medium, a number of devices can be leveraged to deliver both broadband and wavelength-dependent optical multicast.

Power splitter is the most basic device for wavelength and bit-rate transparent physical layer data duplication [18]. Depending on its specific design parameters, a power splitter duplicates an incoming data stream by dividing the incident optical power at predetermined ratios to its output ports. Typically implemented using fused fibers, optical power splitters (otherwise known as directional couplers) are a basic building-block of today's telecom networks, ranging from application in core routers to FTTx (Fiber to the x) installations. As a result of commoditization, optical power splitters are of extremely low cost and high reliability.

Optical power splitters can also be realized using interferometric devices, e.g. multi-mode interferometers (MMIs) and Mach-Zehnder interferometers (MZIs) [18]. In contrast to fused-fiber implementations limited to a fixed power split, these devices can be designed to achieve tunability, allowing precise real-time control of the splitting ratio.

Physical phenomena arising under certain operating conditions in a variety of optical media, such as cross-gain modulation (XGM) and four-wave mixing (FWM), can be exploited to achieve all-optical wavelength multicasting [8, 4], where a data stream modulated on a given optical wavelength is selectively duplicated to a subset of other wavelengths in a data-rate agnostic manner.

While the excess optical power losses of today's passive power splitters are minimal, they do introduce a fundamental insertion loss, e.g. -3 dB for a balanced power split. Since the power of the optical signal must be greater than the sensitivity of the receiver to ensure error-free data recovery, optical interconnects are engineered with sufficient optical link budgets to accommodate these losses. To mitigate the power budget limitation associated with larger-degree power splits, optical amplification is typically leveraged to increase the effective link budget of a given system. While amplifiers are active devices that require additional power, they nevertheless maintain data format transparency to effectively decouple energy consumption from bit rate.

Furthermore, the rapid progress and maturation of integrated photonics in a variety of material systems has enabled the possibility of realizing these devices—both passive and active—with even higher densities, lower costs, and greater operating efficiencies. For example, up to 64-way passive splits on a single planar lightwave circuit (PLC) with a footprint of 2 cm^2 are commercially available. Photonic integrated circuits featuring more than 1000 monolithically integrated functional components on a 0.4 cm^2 chip, with nearly 200 semiconductor optical amplifiers (SOAs) and nearly 300 splitters/combiners, have also been demonstrated [27].

2.2 Network Architecture

We propose a unique architectural solution to enable the run-time configuration and connection of ODDDs to nodes across a datacenter region as they are needed, providing the dynamism necessary to enable efficient use of ODDDs. A high level depiction of our proposed architecture is shown in Figure 1. Each datacenter region is served by an optical network subsystem and a separate electrical management network. An electrical data network that interconnects multiple regions still exists for unicast communications. Each dotted-line represents a physical fiber connection to a high-radix optical space switch (OSS), e.g. a 3D-MEMS-based optical switch. To leverage the capacity advantages offered by Wavelength-Division Multiplexing (WDM), we can envision a design where, at each ToR switch, fixed-wavelength transceivers generate sets of non-overlapping channels, which are then spatially multiplexed and connected to a single port of the OSS.

Depending on the expected RGDD needs of a particular system, any number of variable-size ODDDs can be connected to a subset of the ports of the OSS. As depicted on the right-hand side of Figure 1, we can then dynamically connect each ODDD to specific ToR switches by appropriately configuring the OSS at run-time to construct a transparent end-to-end WDM "light-tree" between nodes participating in a particular RGDD. While each ODDD is statically designed to support a certain number of ports, they can be

cascaded through the OSS to effectively achieve larger-scale optical multicast when necessary. The resulting light-trees represent direct high-bandwidth, low-loss, bufferless and data-rate transparent paths providing homogeneous, hop-less, and constant-latency connections between the multicast senders and receivers. In addition, since the OSS is fully operational as a stand-alone point-to-point circuit switch, the fabric can be configured to completely bypass the ODDDs, forming direct point-to-point circuits when needed.

A centralized controller manages the ToR switches, OSS, ODDDs, and their organization into light-trees. The controller has full topological information of the datacenter region. It processes and accepts explicit requests for RGDD from applications. The centralized controller also interacts with Software-Defined Networking (SDN) capable ToR switches to dynamically and reconfigurably demultiplex flows to appropriate route towards either the multicast-enabled optical fabric or the general-purpose electronic packet-switched aggregation layer at the sender and receiver hosts. All control plane communications happen over the electrical management network to avoid performance problems that might exist in the electrical data network.

Scalability analysis. In the λ^{group} architecture, since the primary purpose of the OSS is to serve as a reconfigurable system-wide connectivity substrate connecting ToR switches to ODDDs, the main requirement for this substrate is port-count scalability. While there are optical switch designs with capabilities such as wavelength-selectivity or nanosecond-scale switching speeds, the scalability requirement is best met by high-radix wavelength-transparent space switches.

Because ODDDs are attached to a subset of the ports of the OSS, R, the total number of optical connections to racks, and $r_{\max}$, the maximum achievable group size, represent trade-offs that are ultimately constrained by S, the port count of the OSS. To support a group data delivery from a sender rack to $r_{\max}-1$ receiver racks using ODDDs with N output ports, a complete N-ary tree of cascaded ODDDs with a depth[2] of $\lceil \log_N(r_{\max}-1)\rceil - 1$ must be constructed. Since the number of nodes in an N-ary tree increases as a geometric series, the total number of N-port ODDDs needed to support $r_{\max}$ is $k = \left\lceil \frac{r_{\max}-2}{N-1} \right\rceil$. However, the total number of rack and ODDD connections to the OSS cannot exceed S. In other words, the following constraint must be met

$$R + k(N+1) \leq S$$

For example, assuming a 1000-port OSS [11] and $N = 16$ ODDDs, by setting $r_{\max} = R$ and solving the constraint above, one single RGDD to the entire region can be achieved when $r_{\max} = R \approx 470$, thus supporting a region of more than 18,000 servers. Assuming $r_{\max} = \frac{1}{2}R$, we can support a region of more than 24,000 servers ($r_{\max} = 303$).

[2]The root of the tree is at a depth of 0.

2.3 Control Plane

2.3.1 Application Interface

λ^{group} lets applications inform the network controller the RGDD traffic demand explicitly, since the data source has accurate and readily available information of the RGDD traffic, namely the data size and the receivers. Applications interact with the controller via the following types of messages:

Request: The application requests for optical resources using the RGDD traffic demand, the IP addresses of the receivers, and a multicast IP address to uniquely identify the request. The RGDD traffic demand is defined as the source data volume multiplied by the number of receivers, which is the cumulative traffic volume if transmitted using unicast flows.[3] The network controller collects the requests and computes the best resource allocation based on the traffic demand. If a request cannot be serviced immediately, it is put in a queue for consideration in the next decision round.

Keep-alive: The application periodically sends a small keep-alive message carrying the multicast IP address of an active request. If the controller stops receiving keep-alive messages for a request, the request is considered withdrawn implicitly. The period and the threshold for missing keep-alive messages are chosen by the network operator.

Withdraw: The application sends a withdraw message to notify the controller of the end of a RGDD session. In addition, since a request may be serviced after being queued for a long time, the application may decide to stop waiting and use an alternative method such as application overlay for RGDD. In that case, the application also sends a withdraw message so the network controller may remove the request from the queue.

Accept: If the optical network has available resources to service a request, the network controller sends an accept message to the application. Then the application can start RGDD using optical multicast.

Reject: The optical network only accepts request for elephant flows to maximize utilization. There are multiple ways to distinguish elephant flows from mice flows. For example, the network operator can pre-define a threshold, or the network controller can adaptively set the threshold based on statistics. The network controller sends a reject message to the application if the requested traffic volume is not heavy enough. Requests asking for a group size larger than the optical network can provide are rejected as well. In these cases, the

[3]This definition gives high priority to RGDD with a large number of receivers, because they potentially create a heavy burden on the network if transmitted as unicast flows.

application may use an alternative method instead.

There can be richer application-network interactions. For instance, the network controller can provide more informative notifications, e.g. the predicted queueing time of the request; and the applications can negotiate with the network using QoS requirements, e.g. giving higher priority to more urgent communications. These issues are however outside of the scope of this paper.

2.3.2 Control Algorithm

Given the traffic demand, the network controller runs the control algorithm to compute the optical network topology, with the goal of maximizing the amount of traffic offloaded to the optical network. This can be formulated as a maximum weighted b-matching problem in hypergraph with additional constraints [20][17]. The RGDD requests form a hypergraph $H = (V, E)$. V is the set of vertices, where each vertex represents a rack; and E is the set of hyperedges connecting any number of vertices, where each hyperedge embodies all the racks involved in a RGDD. The weight of a hyperedge is the RGDD traffic demand. We seek a maximum weight subcollection of hyperedges such that each vertex is met by at most b hyperedges, where b is the number of optical ports per rack. Since ODDDs can be cascaded, we need to consider these additional constraints:

1. Any RGDD to be serviced cannot occupy more optical ports than λ^{group} can provide. Suppose N is the output port count of an ODDD and k is the total number of available ODDDs. For a particular RGDD i that involves r_i racks, we have $r_i \leq k(N+1) - 2(k-1)$.

2. The total number of consumed ODDDs cannot exceed the given number. As explained in Section 2.2, a RGDD of r_i racks consumes $m_i = \lceil \frac{r_i - 2}{N-1} \rceil$ ODDDs. So, $\sum_i m_i = \sum_i \lceil \frac{r_i - 2}{N-1} \rceil \leq k$.

Hypergraph matching is NP-hard [17]. Although there is no existing model for maximum weighted hypergraph b-matching with additional constraints, we envision it has similar complexity. We develop a greedy algorithm to solve the problem. The RGDD requests are sorted by decreasing traffic demand, and those violating constraint 1 are rejected directly. Then the algorithm iteratively selects the RGDD with the greatest demand as long as every involved rack has appeared in less than b previously selected requests and the cumulative number of consumed ODDDs does not violate constraint 2. The algorithm continues until no more requests can be added with all the constraints satisfied. As long as a request is accepted, it is dedicated to the optical network until it finishes. So the control algorithm takes the RGDDs in the middle of transmission as already selected and only allocates the residual resources in each decision process.

Due to space limit, detailed evaluation on optimality and execution time of the algorithm based on recently reported datacenter traffic statistics will appear in an upcoming technical report. At the high level, this algorithm can service a total amount of optical traffic as much as 99% of the optimal solution; and it can finish computation within 6 ms for a 200-rack setting.

2.3.3 Reconfiguration

The network controller informs the OSS of the new topology and instructs the change of interconnections. This operation can be done through the command-line interface, such as TL1, on a typical OSS. Then the network controller sets forwarding rules on the SDN-enabled ToR switches to direct RGDDs through the optical paths. The ToR switches involved in the group communication are each added with an forwarding rule to match its multicast IP address. Finally, the network controller notifies the applications and they can start sending traffic.

2.3.4 Compatibility to Unicast

Although λ^{group} is designed to accelerate RGDD, it can also improve unicast transmission because it is simple to bypass the ODDDs and connect two ToR switches directly through the OSS. λ^{group} adopts c-Through's approach of measuring the cross-rack unicast traffic demand and offload some unicast traffic to the optical network [23]. Since unicast can be regarded as a special case of RGDD with only one receiver, the control algorithm works properly under the mixture of unicast and RGDD traffic. The main difference is that no explicit requests and withdraws are associated with unicast traffic demand. Instead, at each round of traffic demand estimation, all previously estimated demands are considered withdrawn, and all current estimated demands are considered active. Network operators can also set policies to handle unicast and RGDD differently, such as leaving a proportion of optical resources particularly for unicast.

3. ANALYSIS OF ADVANTAGES

Scalable error-free multicast. Most photonic interconnects are designed to operate at extremely low bit-error rates, i.e. $< 10^{-12}$, with sufficient optical power budgets for error-free propagation across hundreds of kilometers. Given the generally shorter reach requirements of datacenters, our design leverages the margins designed into current optical transceivers to support large-scale optical multicast. As will be demonstrated in Section 4, large fan-outs can be realized using 1-Gb/s single-mode transceivers to enable optical multicast groups of over 750 racks or 30,000 servers without amplification.

Simplified group management. λ^{group} provides efficient RGDD without the complexity of traditional routing or group membership protocols. By leveraging a clean-slate approach in which a centralized controller receives explicit RGDD requests from applications and allocates optical resources accordingly, λ^{group} avoids the need for distributed join/leave protocols. As a result, group membership management is greatly simplified, reducing group state knowledge to just the controller and ToR switches.

Efficient data delivery trees. Realizing efficient reliable IP multicast is challenging; alternatively, application layer overlays compromise efficiency for reduced complexity. However, overlay solutions have high network link stress. Even the recently proposed datacenter-optimized BitTorrent has a default link stress of 12 [7]. In contrast, λ^{group} provides RGDD using optical multicast at a layer even below IP, so it has the optimal link stress of 1. Furthermore, each multicast tree over the optical network has homogeneous depth of only 3 hops from the sender to each receiver, ensuring low latency. Lastly, as the optical network is data-rate transparent, data can be transmitted as fast as the transceiver's sending capacity. Since λ^{group}'s optical paths are congestion free, RGDD can be achieved efficiently at full capacity of the end servers.

Simplified flow control, congestion control and reliability. Flow control, congestion control and reliability are simplified, because the optical paths provide a lossless data delivery environment. Loss events are rare, and when they do occur, of-the-shelf reliable multicast protocols are able to recover quickly as shown in Section 4. This is in contrast to a lossy environment where many reliable multicast protocols suffer from feedback (NACK/ACK) implosion and continuous retransmission of the same packet causes congestion collapse [16].

Low power consumption. λ^{group} has fundamental advantages in both energy and capacity comparing to electronic multicast. By duplicating data "in the links", it avoids the high cost and complexity associated with the need for intermediate packet-based multicast-capable core switches. The inherent packet and data-rate transparency of photonics also obviates the need for costly conversions between the electronic and optical domains. This design decouples the power consumption of the photonic fabric from data-rate, thus providing built-in support for speeds beyond 40 Gb/s without any modification to the fabric.

Today's 10–40-Gb/s optical transceivers can draw anywhere from 1–3.5 W per port depending on technology (i.e., single- or multi-mode). So a system of a commercially-available 50-W 320-port 3D-MEMS-based OSS in combination with passive ODDD and one optical channel per port would consume as little as 370 W at 10 Gb/s. Even a worst-case 40-Gb/s single-mode optical system for the same solution would consume no more than 1.2 kW.

In comparison, even a system connected with a single state-of-the-art high-radix Arista 7500E-class switch featuring a best-in-class per-port power consumption of as little 4 W per 10-Gb/s port for a fully-populated chassis would still consume more than 1.3 kW. At 40 Gb/s, the power consumption of such a system increases to approximately 16 W per port, drawing over 5.1 kW for an equivalent 320-rack system.

Low cost of deployment. λ^{group} can also yield immediate cost savings compared to the conventional electronic alternative in terms of both immediate capital outlay and upgrade costs. For example, a recent quote for a Calient 320-port MEMS OSS priced the switch \$90k, or \$281 per port, with splitter-based ODDDs costing less than \$100 each. In comparison, a 7500E-class switch costs \$550 per 10 Gb/s port and \$2,200 per 40 Gb/s port. Furthermore, as the network is upgraded to support faster data rates, the transparency of the optical fabric allows it to stay in place, while the electronic packet-switched solutions must be upgraded accordingly.

4. TESTBED

4.1 Experimental Setup

In order to accurately assess the performance of λ^{group} under the constraints imposed by the application, network protocols, and underlying physical hardware, we construct an end-to-end hardware testbed implementing a small-scale instantiation of our proposed design.

Our end-nodes consist of four hosts running Ubuntu Linux connected to the 1-Gb/s Ethernet ports of a Pronto 3290 switch running an implementation of Open vSwitch (OVS). Through the bridging functionality of OVS, we logically partition the switch into four distinct segments, modeling the functionality of four separate ToR switches. Uplink ports on each ToR switch are connected to both a commodity 100 Mb/s Ethernet switch and a Polatis piezoelectric beam-steering optical space switch. These uplink ports interface with the optics through a media converter attached to a GbE SFP transceiver module. At a subset of the optical switch's ports, we attached a 1×3 balanced optical splitter. While there are a variety of technologies that can be leveraged to achieve optical multicasting, given the criticality of cost and energy consumption in today's datacenters, we start by considering the basic optical power splitter as our ODDD. Finally, the OSS is configured to map the input and two of the outputs of the ODDD to each of our three ToR switches.

The inherent unidirectionality of the ODDD repre-

sents an incompatibility with the physical layer Ethernet standard as implemented in our system. In order to ensure link establishment, the Ethernet PHY requires a signal to be incident on its receiver before allowing message transmission on its transmitter. We successfully address this issue by including terminated optical circulators in line with $N-1$ of the ODDDs' output ports, leaving one backward propagating path through the ODDD in order to satisfy the aforementioned requirement at the sender. This is not an issue for transceivers at the output of the multicaster.

4.2 Experimental Results

4.2.1 Reliable Multicast Performance

At each end host, we run sender and receiver instances of an application built from JGroups [1]—a toolkit enabling reliable multicast—to evaluate the performance of our system under application traffic. JGroups utilizes IP multicast and implements its own set of protocols to detect and retransmit dropped packets. This reliability is implemented using fairly primitive credit-based flow control and basic NAK/ACK unicast-based mechanisms.[4]

The flow control mechanism operates as follows. Each JGroups sender has a maximum number of credits and decrements them whenever a message is sent. The sender blocks when the credits fall to 0, and only resumes sending messages when it receives replenishment credits from all receivers. The receivers maintain a table of credits for all senders and decrement the given sender's credits when a message is received. When a sender's credits drops below a threshold, the receiver will send a replenishment message to the sender.

As a result, a combination of both multicast group data and unicast retransmissions is produced. By matching on pre-established rules at the ToR switch, inbound unicast and multicast traffic is appropriately demultiplexed to the 100 Mb/s electronic packet switch and full-rate optical multicast fabric, respectively.

In order to characterize the performance of RGDD over λ^{group}'s optical core, we design the following experiment. First, we establish a reliable multicast group via JGroups with one node configured as a sender and three nodes configured to join the group as receivers. With the 1×3 ODDD connected *a priori* to the participating nodes through the appropriate configuration of the OSS, we direct all JGroups multicast traffic originating from the sender to the input of the ODDD by inserting the corresponding rules into our OpenFlow switch. Any back-propagating traffic originating from the receivers is isolated from the ODDD and sent exclusively through the 100 Mb/s packet switch.

[4]Here we show that even a simple protocol such as JGroups yields near optimal performance using λ^{group}. As such, while there exists more sophisticated reliability protocols (e.g. OpenPGM [19]), we expect similarly optimal results.

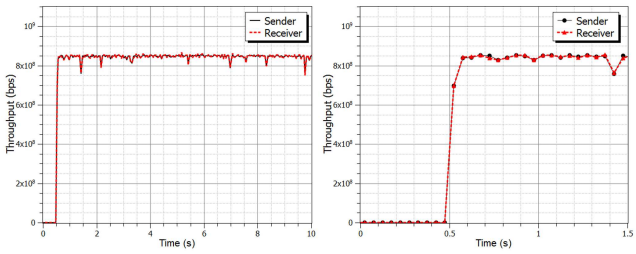

Figure 2: Throughput performance of JGroups at the sender and a representative receiver; zoom-in of the time from 0–1.5 seconds

Figure 2 plots the throughput of JGroups as observed through *tcpstat* at the sender and one representative receiver over a 10-second window sampling every 50 ms. As soon as the application begins transmitting, the transmission rate quickly saturates and stabilizes in under two sample periods (i.e., < 100 ms). We observe little to no difference in their instantaneous throughput, indicating minimal packet loss across the fabric. We also measured JGroups's performancewhen run over a 1-Gb/s electronic packet switch and observe no appreciable difference in throughput performance, indicating that the sender is sending as fast as its interface allows in both cases.

JGroups performs optimally over λ^{group} in spite of it's simplicity. To illustrate the inefficacy of JGroups under less ideal network conditions, we briefly evaluated its performance over the oversubscribed packet-switched core. While the core is capable of 100 Mb/s, we measured average throughput of only 10 Mb/s. Closer observation of the instantaneous throughput revealed what appeared to be regularly-spaced bursts and subsequent back-offs. This can be attributed to the fact that, while the interfaces at the nodes see a 1-Gb/s link to the ToR switch, there exists a 10:1 bottleneck that quickly becomes congested as the nodes attempt to inject traffic into the network at the rate of their interface. JGroups' flow control mechanism detects the resulting packet loss and attempts to retransmit the lost packets, but continues to do so at the rate negotiated by its interface with the ToR switch.

In summary, while JGroups' simplistic flow control mechanism may be ineffective when operating in an oversubscribed environment, λ^{group} can ensure consistent network performance by enabling zero packet loss between participants in a group data delivery.

4.2.2 Physical Layer Scalability

The optical power budget provided by a given transceiver places an upper bound to the scalability of λ^{group} to larger group sizes when utilizing purely

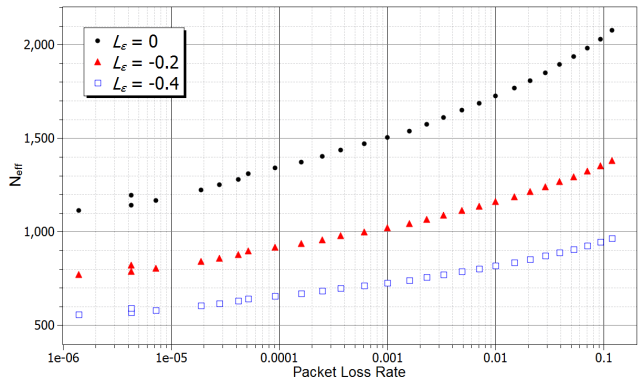

Figure 3: Packet loss rate vs. effective ODDD size (N_{eff})

passive optical splitters without amplification.

To evaluate the limitation on group size imposed by this power budget on a purely passive implementation of λ^{group}, we first formulate an analytical expression relating the insertion loss of a practical N-way power splitter to N. Using our testbed, we then experimentally quantify the effect of insertion loss on the end-to-end performance of our system to determine the maximum theoretical values for N assuming no amplification.

For an N-port balanced splitter, $1/N$ of the incident optical power is ideally delivered to each output port. However, excess losses and non-uniform splitting ratios resulting from imperfections in the manufacturing process can lead to additional insertion loss. As such, for a practical balanced N-way splitter, the total insertion loss L_N (in dB) can be estimated by the following:

$$L_N = 10\log_{10}\left(\frac{1}{N}\right) + \log_n(N)L_\epsilon(n)$$

The first term represents the insertion loss in dB introduced by an ideal balanced power split. The second term captures the total additional loss arising from non-idealities. This relation holds assuming the N-way splitter is implemented using a balanced tree of single-fusion $1 \times n$ couplers, with each introducing a non-ideal loss of $L_\epsilon(n)$ (in dB). Splitters with $N > 2$ are typically implemented using trees of 1×2 couplers, yield a design where the number of coupling elements to which a signal is subjected is maximized. Given the worst-case insertion loss reported by the manufacturer of telecom-grade power splitters used in our testbed for $2 \leq N \leq 16$, we assume a non-ideal loss $L_\epsilon(2) \approx -0.4$ dB.

Next, we experimentally model the insertion loss introduced by a N-way splitter-based ODDD by inserting a variable optical attenuator (VOA) in line with a path through our testbed. As we increase the attenuation on the optical signal using the VOA, we measure the packet loss rate (PLR) using *iperf* at a receiver to determine the network level performance in response to the physical layer impairment introduced by an increasing effective ODDD size. In Figure 3 we plot the measured PLR vs. N_{eff}—derived from the expression for L_N above—for various values of L_ϵ. Considering a single unamplified N-way ODDD and the power budget afforded by the components used in our implemented system, N_{eff} thus represents the maximum number of racks reachable while maintaining a given PLR. Assuming 40 servers per rack, we see that we can maintain a PLR < 0.0001 while potentially supporting group sizes as large as 25,000 assuming $L_\epsilon = -0.4dB$ and more than 36,000 assuming a reasonable $L_\epsilon = -0.2$ dB.

5. SIMULATION

In this section, we evaluate λ^{group} in a larger setting using flow-level simulations for a production-scale datacenter. We first compare its performance with a variety of state-of-the-art datacenter architectures for RGDD, then investigate the effects of varying the availability of optical resources and group sizes.

5.1 Simulation Setting

5.1.1 *Simulation Methodology*

There are 120 racks in the simulated network, each with 40 servers. We use the flow completion time as our performance metric, computed based on the flow's max-min fair share bandwidth. A multicast flow's max-min fair share is determined by the most congested link on the multicast tree. RGDD is completed when all the data for that flow is delivered to all receivers. Detailed transport layer protocol behavior is not considered in this flow-level simulation, so our simulation results provide an ideal-case upper-bound on transport layer performance for each of the architectures compared. We believe this bound is relatively tight for λ^{group} because packet loss is expected be rare. In contrast, we believe this upper-bound is fairly lose for the other architectures compared because packet loss is expected to be more common.

5.1.2 *Networks Compared*

1. **Oversubscribe**: As a baseline, we simulate a 4:1 oversubscribed network with no IP multicast capability using a single core switch. The server links have 1 Gb/s of bandwidth and the ToR switch to core switch links have 10 Gb/s. RGDD is handled by naive unicast.

2. **Multicast oversubscribe**: An intuitive way of improving RGDD is to use IP multicast, so we simulate a 4:1 oversubscribed network with IP multicast enabled in all the switches.

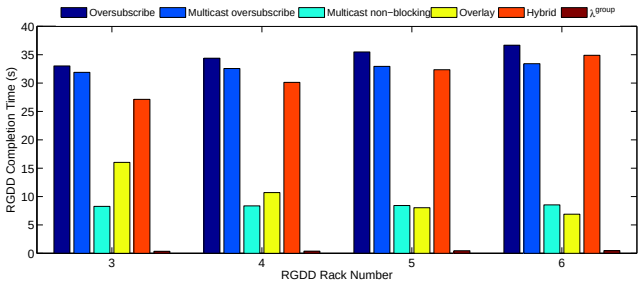

Figure 4: Average RGDD completion time of various architectures under heavy traffic

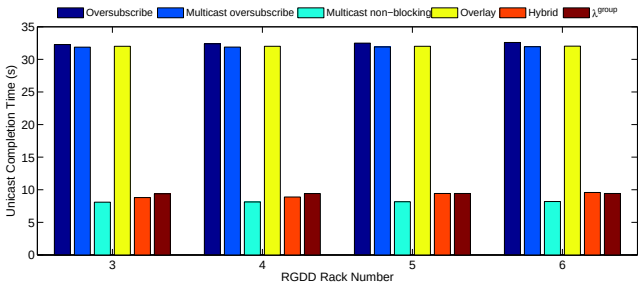

Figure 5: Average unicast completion time of various architectures under heavy traffic

3. **Multicast non-blocking**: As a further improvement, we simulate a non-blocking network with IP multicast enabled. It is the same as the multicast oversubscribed architecture except that the links to the core switch have 40 Gb/s bandwidth.

4. λ^{group}: We simulate λ^{group} deployed on the oversubscribed electrical network. The optical network has a 480-port OSS, which connects to 40 6-port ODDDs and to each ToR switch on 2 ports. All optical links are 40 Gb/s to guarantee non-blocking inter-rack communications. We assume the circuit reconfiguration delay is 10 ms. The control algorithm computation time is measured at run time. The reconfiguration interval is set to 100 ms.

5. **Hybrid**: We simulate the key properties of c-Through [23] and Helios [9]. The setting is similar to λ^{group}, but each ToR switch has 4 optical links connected to the 480-port OSS. All heavy cross-rack communications, unicast or RGDD, are accelerated optically. RGDD is handled by naive unicast.

6. **Overlay**: We simulate a multi-rooted tree overlay network [6] with modifications inspired by the topology-awareness of Cornet [7] to minimize cross-rack communications. We form a swarm among each of the leading servers across racks, and then subsequently distribute the content among servers in the same rack in another swarm. The overlay is built on top of the oversubscribed network.

5.1.3 *Communication Patterns*

We adopt the synthetic unicast traffic patterns in Helios [9] to stress the network. Unicast and RGDD traffic patterns are generated in rounds each lasting 10 seconds and mixed as follows:

Unicast traffic: We create both **light** and **heavy** unicast traffic. The racks are indexed from 0 to 119 and the servers in each rack are indexed from 0 to 39. The traffic shifts round by round, with new traffic patterns created at the beginning of each round. All flows are 100 Mb in size. These patterns serve as background traffic, so RGDD traffic can be added onto either the light or the heavy unicast traffic.

1. **Light unicast traffic**: In round t, any server j in rack i talks to server j in racks $(i+t\pm1)$ mod 120, $(i+t\pm2)$ mod 120, and $(i+t\pm3)$ mod 120.

2. **Heavy unicast traffic**: In round t, any server j in rack i talks to server j in racks $(i+t\pm1)$ mod 120, $(i+t\pm2)$ mod 120, $(i+t\pm3)$ mod 120, ..., $(i+t\pm40)$ mod 120.

RGDD traffic: For RGDD of a particular size n, we randomly choose n racks in each round and let server 0 in one rack send to servers 1-39 in the same rack and to all servers 0-39 in the other $n-1$ racks. We vary the number of groups and the group sizes to evaluate different scales of RGDD. The data size is 100 Mb in each group.

5.2 Simulation Results

5.2.1 *Performance Comparison*

In each round, we simulate 40 simultaneous RGDDs involving 3 to 6 racks. The simulation is run 10 times for each traffic pattern, each run lasting 60 seconds. Figure 4 and 5 show the average RGDD and unicast completion time under the heavy traffic scenario. As the light traffic scenario has similar trends, we only describe the results in the text. We make the following observations:

First, λ^{group} can accelerate RGDD by an order of magnitude compared to alternative approaches under heavy traffic. In Figure 4, λ^{group} takes less than 0.4 s to finish RGDD of any group size, resulting in 14× to 93× improvement compared to the other architectures. The benefit of λ^{group} is less significant under light traffic because the electrical network core is less congested. Regardless, λ^{group} provides improvements of at least 2×.

λ^{group} uses optical multicast, so it benefits from optimal link stress and ultra-fast optical transmission.

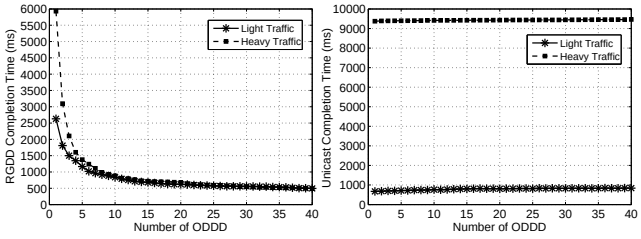

Figure 6: λ^{group} RGDD completion time (left) and unicast completion time (right) under various numbers of ODDDs

Interestingly, λ^{group} even outperforms the multicast non-blocking architecture. Since λ^{group} is built on the slow electrical packet-switched network, the unicast flows traversing the congested network core receive a small share and cannot fully utilize the link bandwidth at the edge. This residual bandwidth at the edge is used by the multicast flow when accelerated optically. In the non-blocking network, the transmission speed of all the flows are determined by their max-min fair share rate. Because the network core no longer rate-limits the unicast flows, group deliveries get a smaller fair share bandwidth at the edge.

Second, λ^{group} can improve unicast transmission almost as much as the hybrid architecture. After the RGDD traffic is serviced, there are still remaining optical ports for unicast flows. The relatively short reconfiguration interval (100 ms) also enables the optical ports to be utilized by unicast traffic shortly after the RGDD is finished. In Figure 5, λ^{group} performs slightly worse than the hybrid network when the group size is small. This is because optical ports are connected to ODDDs and occupied by the multicast traffic, but they could otherwise be used by unicast flows. Nevertheless, the hybrid network is quickly surpassed as the group size grows, since it needs to create an increasing number of unicast flows to service group communications, thus degrading the transmission speed.

Third, IP multicast is not effective when the network is under heavy congestion. We observe that the multicast oversubscribed architecture only improves upon the oversubscribed network slightly in Figure 4. This is because the network core is still very congested under heavy unicast traffic, even though the group deliveries can be realized by multicast flows. In contrast, the multicast non-blocking structure shows a dramatic improvement, since the network core has full bisection bandwidth. However, it is still about 20× slower than λ^{group} as discussed above.

Fourth, the hybrid architecture hardly helps with RGDD. The hybrid network accelerates unicast transmission dramatically, but its RGDD completion time is only 10% better than the oversubscribed network in Figure 4. Since the RGDD completion time is determined by the slowest flow, even failing to accelerate one flow optically drags down overall group transmission.

Fifth, overlays can benefit RGDD to a certain degree, but still much less than λ^{group}. RGDD is faster using overlay because the sender creates many unicast sessions to send a piece of data to each of the receivers, thus taking a greater cumulative fair share bandwidth from the competing flows. However, overlays still use unicast flows, so the acceleration is very limited compared to λ^{group}, which has optimal link stress. We also observe that overlays improve RGDD more significantly as the group size increases, since it can grab bandwidth more aggressively by sending smaller data pieces to a greater number of recipients using more sessions. However, this trend cannot hold forever because 1) the transmission rate is bounded by the link capacity at the end server; and 2) TCP performs poorly if the source data is divided into tiny flows.

5.2.2 *Effect of Varying the Number of ODDDs*

To quantify the effect of optical multicast, we use the same experiment setting as the previous subsection and measure the performance of λ^{group} under various numbers of ODDDs in Figure 6. Due to space limitations, we show results for group size of 6, but the trend is similar for the other group sizes. The average RGDD completion time decreases as the number of ODDDs grows—very rapidly at first, but diminishing after a certain point. This indicates that λ^{group} should be allocated with sufficient—or at least a reasonable number of—ODDDs, as adding ODDDs can improve the performance continuously with significant improvement from the first few additions.

Unicast traffic and multicast traffic compete for the optical ports. When more ODDDs are available, the optical network can undertake more multicast traffic, which takes away some ports originally used for unicast traffic. We observe that adding ODDDs, or facilitating optical multicast, only causes the unicast flow completion time to increase slightly. However, the multicast performance can improve significantly at the cost of a slight decay in unicast performance, indicating that adding ODDDs to the optical network brings more benefit than detriment.

5.2.3 *Effect of Varying the Group Size*

We evaluate the performance of λ^{group} in handling large groups. The group size ranges from 10 to 120 (the maximum group size for the network) with an interval of 5. We use 4, 6, 8, 10 as the number of simultaneous groups and only perform the simulation on the light traffic scenario, where the network is able to clear up the traffic within each round. To accommodate these large multicast groups, we use 15 16-port ODDDs, which take

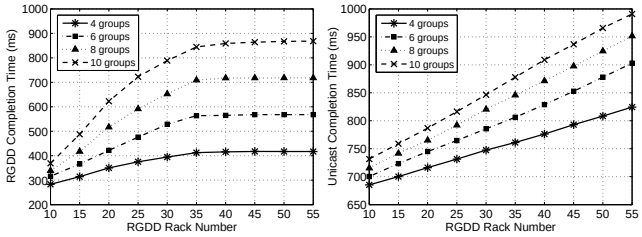

Figure 7: λ^{group} RGDD completion time (left) and unicast completion time (right) for various numbers of groups and group sizes

up the same number of ports on the 480-port OSS as the previous experiments.

In the left subplot of Figure 7, the average RGDD completion time increases with the group size until the group size reaches 35. The RGDD completion time is the sum of the service time and the wait time. The service time is always the same since the transmission bandwidth is fixed in λ^{group}. Group data deliveries of larger sizes occupy more optical ports, so the wait time increases. However, it is bounded after the group size reaches a certain level and the optical ports are depleted.

Comparing the curves in the left figure, we observe the average RGDD completion time grows linearly with the number of groups when the group size is the same, because more groups render longer wait times. In the right figure, the average unicast completion time increases linearly with respect to both the group size and the number of groups, because the unicast traffic leverages the amount of optical resource left by the group traffic. The shape of the curves in both the left and the right figures show good scalability of λ^{group} with increasing group sizes and number of groups.

6. RELATED WORK

The most closely related work are our previous papers [24, 25] that partially laid out the basis for using optics for enabling multicast communications. In [24], we show the feasibility of using a single ODDD from the perspective of the physical layer by duplicating a synthetically-generated 80-Gb/s WDM data stream composed of 8 × 10-Gb/s pseudo-random bit streams while maintaining a bit-error rate of $< 10^{-12}$. In [25], we make a positional argument for a framework where ToR switches are dynamically connected to optical devices through an optical switch to support several traffic patterns, multicast being one of them. However, no concrete system design is proposed, nor is there any analysis on system scalability. The resource allocation algorithm considered is primitive and does not even allow the cascade of ODDDs. The experiment presented only shows the feasibility of sending traffic end-to-end; no analysis is given on the effectiveness of end-to-end flow control and recovery, or on the physical layer scalability of cascading ODDDs. Finally, the limited simulation results only evaluate the primitive resource allocation algorithm, and only use an oversubscribed network and a non-blocking network as comparison points. All in all, this paper is the first time all of these issues are addressed in-depth.

A number of previous work present solutions that improve IP multicast in datacenters. Dr. Multicast selectively maps multicast to unicast transmissions to mitigate the disruption caused by a large number of multicast groups [22]. Li *et al.* design a novel multi-class Bloom Filter to reduce the Bloom Filter traffic leakage and thus efficiently compress the multicast forwarding table [12]. ESM builds efficient multicast trees customized to datacenters and achieves scalable multicast routing by combining in-packet Bloom Filter and in-switch routing entries [13]. RDCM assists reliable group data delivery by repairing lost packets in a peer-to-peer way [14]. Compared to this set of work, λ^{group} does not involve conventional multicast routing protocols. It builds simple and homogeneous multicast trees to perform reliable data replication optically, thereby simplifying flow control, congestion control, and loss recovery. Leveraging SDN also allows the network controller to have global intelligence, thus freeing end servers and network switches from complicated state management.

RGDD can also be realized by non-IP multicast approaches. Twitter uses BitTorrent to distribute software updates to its servers [2]. Cornet develops a BitTorrent-like protocol optimized for datacenter group transmission [7]. These application layer overlay solutions use unicast for group transmission, so the network link stress can be very high. Evidence in [3] also shows application layer overlay can exhibit instability and low throughput in real datacenters. λ^{group} achieves optimal link stress. The zero-loss nature and high capacity of the optical network also guarantee stable transmission. Datacast [5] proposes packet caching at switches and edge disjoint Steiner trees forwarding to realize improved RGDD. This solution is not readily implementable because packet caching adds significant complexity to switches and the idea is still being researched; moreover, only specialized network structures such as BCube and CamCube can benefit from multiple Steiner tree forwarding. In contrast, λ^{group} is designed based on off-the-shelf SDN switches and photonic devices, making it highly practical.

7. CONCLUSION

This paper presents the design of λ^{group}, a clean-slate approach that performs data duplication in the physical layer to support RGDD. Compared to the conventional solutions that duplicate data in the network or applica-

tion layer, λ^{group} achieves efficient data transmission over the ultra-fast, loss-free, energy-efficient and low cost optical paths, with simplified flow control, congestion control, and group membership management. The architecture scales to RGDD of over 450 racks using a 1000-port OSS, and a simple control algorithm can produce near-optimal optical resource allocation within 6 ms for a 200-rack setting. A prototype implementation demonstrates that λ^{group} has minimal packet loss on the optical paths and it can achieve a high ODDD fan-out of over 750 racks or 30,000 receivers without amplification. Large-scale simulations show that λ^{group} provides 14× to 93× performance improvement against various state-of-the-art RGDD approaches under synthetic traffic.

8. REFERENCES

[1] JGroups - A Toolkit for Reliable Multicast Communication, http://www.jgroups.org/.

[2] Murder: Fast Datacenter Code Deploys Using BitTorrent, http://engineering.twitter.com/2010/07/murder-fast-datacenter-code-deploys.html.

[3] D. Basin, K. Birman, I. Keidar, and Y. Vigfusson. Sources of Instability in Data Center Multicast. In *LADIS '10*, pages 32–37, Zurich, Switzerland, Nov. 2010.

[4] A. Biberman, B. G. Lee, A. C. Turner-Foster, M. A. Foster, M. Lipson, A. L. Gaeta, and K. Bergman. Wavelength Multicasting in Silicon Photonic Nanowires. *Opt. Express*, 18(17):18047–18055, Aug. 2010.

[5] J. Cao, C. Guo, G. Lu, Y. Xiong, Y. Zheng, Y. Zhang, Y. Zhu, and C. Chen. Datacast: a Scalable and Efficient Reliable Group Data Delivery Service for Data Centers. In *CoNEXT '12*, pages 37–48, Nice, France, Dec. 2012.

[6] M. Castro, P. Druschel, A.-M. Kermarrec, A. Nandi, A. Rowstron, and A. Singh. SplitStream: High-bandwidth Multicast in Cooperative Environments. In *SOSP '03*, pages 298–313, Bolton Landing, NY, USA, Oct. 2003.

[7] M. Chowdhury, M. Zaharia, J. Ma, M. I. Jordan, and I. Stoica. Managing Data Transfers in Computer Clusters with Orchestra. In *SIGCOMM '11*, pages 98–109, Toronto, Canada, Aug. 2011.

[8] G. Contestabile, N. Calabretta, R. Proietti, and E. Ciaramella. Double-stage Cross-gain Modulation in SOAs: an Effective Technique for WDM Multicasting. *Photonics Technology Letters, IEEE*, 18(1):181–183, 2006.

[9] N. Farrington, G. Porter, S. Radhakrishnan, H. H. Bazzaz, V. Subramanya, Y. Fainman, G. Papen, and A. Vahdat. Helios: A Hybrid Electrical/Optical Switch Architecture for Modular Data Centers. In *SIGCOMM '10*, page 339, New Delhi, India, Aug. 2010.

[10] S. Ghemawat, H. Gobioff, and S.-T. Leung. The Google File System. In *SOSP '03*, pages 29–43, Bolton Landing, NY, USA, Oct. 2003.

[11] Kim, J., et. al. 1100 x 1100 port mems-based optical crossconnect with 4-db maximum loss. *Photonics Technology Letters, IEEE*, 15(11):1537–1539, 2003.

[12] D. Li, H. Cui, Y. Hu, Y. Xia, and X. Wang. Scalable Data Center Multicast Using Multi-class Bloom Filter. In *ICNP '11*, pages 266–275, Vancouver, Canada, Oct. 2011.

[13] D. Li, Y. Li, J. Wu, S. Su, and J. Yu. ESM: Efficient and Scalable Data Center Multicast Routing. *IEEE/ACM Transactions on Networking*, 20(3):944–955, June 2012.

[14] D. Li, M. Xu, M.-C. Zhao, C. Guo, Y. Zhang, and M.-Y. Wu. RDCM: Reliable Data Center Multicast. In *INFOCOM '11*, pages 56–60, Shanghai, China, Apr. 2011.

[15] W. Mach and E. Schikuta. Parallel Database Join Operations in Heterogeneous Grids. In *PDCAT '07*, pages 236–243, Washington D.C., USA, 2007.

[16] K. Obraczka. Multicast Transport Protocols: a Survey and Taxonomy. *Communications Magazine, IEEE*, 36(1):94–102, 1998.

[17] O. Parekh. Iterative Packing for Demand and Hypergraph Matching. In *IPCO'11*, pages 349–361, Berlin, Heidelberg, 2011. Springer-Verlag.

[18] R. Ramaswami, K. Sivarajan, and G. H. Sasaki. *Optical Networks: A Practical Perspective*. Morgan Kaufmann, 3rd edition, 2009.

[19] T. Speakman, J. Crowcroft, J. Gemmell, D. Farinacci, S. Lin, D. Leshchiner, M. Luby, T. Montgomery, L. Rizzo, A. Tweedly, N. Bhaskar, R. Edmonstone, R. Sumanasekera, and L. Vicisano. PGM Reliable Transport Protocol Specification. RFC 3208, Dec. 2001.

[20] A. Tamir and J. S. Mitchell. A Maximum b-Matching Problem Arising From Median Location Models With Applications To The Roommates Problem. *Mathematical Programming*, 80:171–194, 1995.

[21] K. Thomas, C. Grier, J. Ma, V. Paxson, and D. Song. Design and Evaluation of a Real-Time URL Spam Filtering Service. In *IEEE Symposium on Security and Privacy*, Oakland, CA, USA, May 2011.

[22] Y. Vigfusson, H. Abu-Libdeh, M. Balakrishnan, K. Birman, R. Burgess, G. Chockler, H. Li, and Y. Tock. Dr. multicast: Rx for Data Center Communication Scalability. In *EuroSys '10*, pages 349–362, Paris, France, Apr. 2010.

[23] G. Wang, D. G. Andersen, M. Kaminsky, K. Papagiannaki, T. S. E. Ng, M. Kozuch, and M. Ryan. c-Through: Part-time Optics in Data Centers. In *SIGCOMM '10*, page 327, New Delhi, India, Aug. 2010.

[24] H. Wang, C. Chen, K. Sripanidkulchai, S. Sahu, and K. Bergman. Dynamically Reconfigurable Photonic Resources for Optically Connected Data Center Networks. In *OFC/NFOEC '12*, 2012.

[25] H. Wang, Y. Xia, K. Bergman, T. S. E. Ng, S. Sahu, and K. Sripanidkulchai. Rethinking the Physical Layer of Data Center Networks of the Next Decade: Using Optics to Enable Efficient *-Cast Connectivity. *SIGCOMM Computer Communication Review*, 43(3):To appear, July 2013.

[26] M. Wiesmann, F. Pedone, A. Schiper, B. Kemme, and G. Alonso. Database Replication Techniques: a Three Parameter Classification. In *SRDS '00*, pages 206–215, 2000.

[27] A. Wonfor, H. Wang, R. Penty, and I. White. Large Port Count High-Speed Optical Switch Fabric for Use Within Datacenters. *Optical Communications and Networking, IEEE/OSA Journal of*, 3(8):A32–A39, 2011.