Why choose when you can have both: Programmable data planes meet programmable optics

Chris Misa*
University of Oregon

Matthew Nance-Hall*†
Scala Computing, Inc.

Reza Rejaie
University of Oregon

Walter Willinger
NIKSUN, Inc.

Ramakrishnan Durairajan
University of Oregon and Link Oregon

Abstract

Recent advances in programmable optics have shown great promise in providing runtime control over a network's topological behavior to achieve spatial adaptability (*e.g.,* dynamically provision new wavelengths). At the same time, the emergence of programmable data plane technologies has revolutionized how a network's forwarding behavior can be controlled at runtime to accomplish temporal flexibility (*e.g.,* "on-the-fly" traffic aggregation). Unfortunately, a lingering chasm between optical systems and digital packet systems researchers prevents modern-day network applications to simultaneously benefit from both of these exciting developments. To overcome this divide, we propose in this paper ShapeShifter, a novel and principled approach towards integrating programmability in both packet and optical layers and jointly realizing spatial adaptability and temporal flexibility in practice. To provide the necessary technological foundation for this integration, ShapeShifter relies on recent progress in *runtime programmability* in both communities.

CCS Concepts

• **Networks → Network design principles**; *Cross-layer protocols*; *Physical links*; *Intermediate nodes*.

ACM Reference Format:
Chris Misa, Matthew Nance-Hall, Reza Rejaie, Walter Willinger, and Ramakrishnan Durairajan. 2025. Why choose when you can have both: Programmable data planes meet programmable optics. In *2nd Workshop on Networks for AI Computing (SIGCOMM '25), September 8–11, 2025, Coimbra, Portugal.* ACM, New York, NY, USA, 8 pages. https://doi.org/10.1145/3748273.3749200

1 Introduction

The low-latency, high-bandwidth demands of emergent AI applications lead to a new and important class of management and security tasks in cloud provider wide-area networks (WANs) that require fine-grained runtime control of both how packets are forwarded through a network and the spatial configuration of the network's underlying optical topology. For example, streaming VBR video to and from edge devices with minimal round-trip latency (*e.g.,* to enable real-time video interaction with AI-based avatars) requires careful control over forwarding (*e.g.,* to minimize latency) and topology (*e.g.,* to adjust available bandwidth based on shifting user patterns). On the one hand, rapid advances in programmable data plane technologies (*e.g.,* switches), coupled with software-defined networking (SDN) and network function virtualization (NFV), have enabled ever more runtime control over a network's *forwarding behavior* to provide the desired temporal flexibility. On the other hand, recent progress in the area of programmable optics has provided new opportunities to exert greater runtime control over a network's *topological behavior* to achieve higher degrees of spatial adaptability.

Unfortunately, these exciting developments notwithstanding, achieving the necessary temporal flexibility and spatial adaptability at the same time in today's AI network infrastructures is currently largely infeasible [44]. For one, there has been a long-standing rift between the digital packet systems researchers and their optical systems counterparts [25]. This so called *packet-optical chasm* [34] is no accident as the Internet's success is, to a large degree, enabled by independent and logically separated abstract layers. Here, the optical systems researchers focus almost exclusively on the optical network layer and argue that because of theoretical efforts and optimization techniques for a programmable optical layer [21, 39, 58, 63], spatial adaptability is feasible in practice. At the same time, the digital packet systems community is mainly concerned with the higher layers of the network stack, assumes a largely static optical layer that allows for little to no programmability due to pragmatic issues such as reconfiguration delays caused by amplifiers [33], and leverages innovations such as programmable switches, SDN, and NFV to enable the intelligence and agility to support the desired temporal flexibility.

Besides this existing division, there are also technical reasons that have prevented the simultaneous realization of temporal flexibility and spatial adaptability to date. For example, existing efforts to jointly achieve both properties focus on a narrow class of tasks (*e.g.,* traffic engineering [40, 69, 87]). Even for this narrow class, the main concerns are the optimization problems that stem from the joint consideration of the two layers (*e.g.,* traffic forwarding and wavelength assignment is NP-Complete [85]). Finally, a lack of large-scale testbeds with programmable optics and switches is an additional practical deterrent. As a result, state-of-the-art solutions are mutually exclusive, catering to temporal flexibility and spatial adaptability individually but not simultaneously. An outward sign of this mutual exclusivity is a general lack of datasets that provide both IP packet-level information (*i.e.,* traffic matrices) and optical layer information (*i.e.,* physical topologies) for real-world production

*Both authors contributed equally to this research.
†Work completed while at the University of Oregon.

SIGCOMM '25, Coimbra, Portugal

ACM ISBN 979-8-4007-2082-6/25/09
https://doi.org/10.1145/3748273.3749200

networks. Until recently (*e.g.,* [13, 29]), such datasets were typically proprietary in nature.

To overcome these existing roadblocks, we propose a unified approach that we call ShapeShifter to seamlessly integrate the programmability in both the data plane and optical plane, thereby bridging the existing packet-optical chasm. The key novelty of ShapeShifter (compared to prior efforts, *e.g.,* [44]) is the goal of coordinating *runtime control* across the packet-optical chasm to achieve temporal flexibility and spatial adaptability at the same time. ShapeShifter leverages recent momentum in *runtime programmability* in both the data and optical planes (§ 4) to enable flexible network fabric for future AI applications to experience the best of both worlds. Through illustrative scenarios (§ 3.2), we qualitatively demonstrate the rigidity inherent in packet-only and optics-only solutions, highlighting the limitations they impose on the class of tasks we aim to support. Additionally, our empirical experiments (in § 3.3) provide quantitative evidence showcasing the tangible benefits achieved through the jointly optimized packet and optical layers. We detail the key technology developments in § 4 and the challenges and building blocks of ShapeShifter in § 5. We outline how the community can help in realizing ShapeShifter in § 6.

2 Background and Preliminaries

An ever-evolving cyberattack landscape and growing performance demands (see § 3.1) are driving a paradigm shift in how networks for AI computing—*e.g.,* cloud provider WANs—are configured and managed. In response to this paradigm shift, we have witnessed two key developments.

(i) **Programmable data planes** (PDPs), composed of programmable switches [5, 10, 19] coupled with innovations in SDN and NFV, have recently emerged as a powerful endpoint technology to address several networking problems [32, 46, 54, 55, 75, 81]. Their key advantage is the ability to inspect every packet at line rate and perform a flexible (though still limited) set of computations per packet—henceforth known as *temporal flexibility*. The results of these computations can be aggregated and exported to a control plane for traffic monitoring [32, 55] or used to make intelligent per-packet forwarding decisions (*e.g.,* select output port, drop packet, etc.), influencing network's traffic *forwarding behavior*.

(ii) **Optical topology programming** (OTP) enables runtime adaptations to the physical topology of a network [24, 59, 60]. OTP involves a set of programmable operations applied to optical resources (*e.g.,* fiber-optic links, transponders, ROADMs, etc.). These operations include (1) adding new wavelengths between two nodes, increasing bandwidth between them; (2) deactivating wavelengths, reducing the bandwidth between the nodes; and (3) reallocating as well as moving bandwidth from one link to another. With these operations, OTP has the potential to revolutionize WAN network operations by providing unparalleled reconfigurability to achieve *spatial adaptability*: dynamically adapt topology to changing traffic demands by opportunistically programming (*i.e.,* adding, removing, or reallocating) wavelengths. Spatial adaptability influences a network's *topological behavior*.

3 Application pull

Networks for AI computing increasingly demand high degrees of flexibility from the underlying infrastructure. For example, see the usecases listed in § 1 of [22]. We posit that a jointly optimized optical-packet architecture can give networks more power to serve these demand better (§ 4) and that the only thing in the way of accessing this power is a prevailing entrenched view that assumes temporal flexibility and spatial adaptability are two mutually exclusive properties. To illustrate, we consider the following two instances in detail, though our work is applicable to the wider range of applications considered in prior efforts [22, 79, 80, 84].

3.1 Representative Applications

Next-generation Traffic Management. Large-scale adoption of generative AI models [78] and their growing integration into a myriad of widely-used services (*e.g.,* web search, video streaming, extended reality) significantly accelerates the traffic demand on cloud WANs supporting AI compute. For example, streaming real-time, AI-generated, high-fidelity 3D content requires orders of magnitude higher bandwidth—far beyond what the current networks can handle [20]. This trend is well documented in the SDN [11, 14, 38, 77], PDP [62, 83] and OTP [48, 51, 70, 76] communities, and of importance to cloud/content providers, enterprises, and ISPs alike.

Effectively dealing with tomorrow's traffic entails carefully identifying different traffic classes based on observed traffic features (*e.g.,* video flows vs. interactive web application flows) and treating each traffic class according to its distinctive semantic demands (*e.g.,* consistent throughput for video, low latency for web applications). A wide range of work leverages temporal flexibility of existing packet-processing systems to maximize performance on a given static network topology (*e.g.,* in SDN [17, 45] and using programmable switch hardware [15, 31, 42, 43, 52, 61, 66]). However, talk about the fast approaching death of "Moore's law for networking" [27, 50, 65] (which puts a finite upper-bound on the amount of data that can be transmitted over a single span of optical fiber [28]), coupled with observed trends of ever-increasing traffic demands implies that ultimately traffic management solutions will require spacial adaptability to dynamically and intelligently increase link bandwidths in response to large yet sudden changes in the encountered traffic (*e.g.,* flash crowds). While there have been responses to this issue (see § 4) in the form of proprietary testbeds and lab-based experiments on OTP, they have yet to be deployed in actual production networks in tandem with PDP technologies. By jointly considering OTP and PDP, network operators would be able to accurately identify various traffic classes within the network using PDP, while ensuring performance guarantees for each identified traffic class using OTP (*e.g.,* by segregating/optimizing their respective transmission paths).

Programmable network defenses. Today, a sophisticated attacker can use LLMs to help design and execute increasingly sophisticated multi-stage network intrusions [67] against cloud provider WANs. For example, an adversary could target vulnerable links and endpoints with dynamic and massive volumetric DDoS attacks [56] or leverage infected VMs to maliciously compromise distributed learning algorithms [16, 47].

Irrespective of their precise nature, these attacks typically unfold in discrete phases (*e.g.,* “kill chain" [37, 64]). First, there is reconnaissance with active network measurement tests to infer topology information and/or vulnerable algorithms and implementations. This is followed by small-scale trial attacks to validate and optimize the discovered weaknesses. Finally, the attack is tuned to the network or application of interest and executed at full force leading to disruption of service and/or interference with learning outcomes [41, 72].

Combining OTP and PDP promises to address core limitations of current approaches to defending against such sophisticated multi-stage intrusions targeting networks for AI compute. Current works explore defenses against multi-stage attacks using spatial adaptability [35, 60] or temporal flexibility [46, 81, 86] in isolation. However, combining OTP and PDP enables defenses that *(i)* accurately distinguish attack packets from benign packets using PDP, and *(ii)* use OTP to ensure that any thus-identified attack traffic is physically isolated from benign traffic by routing them through separate wavelengths. For example, if an adversary launches a dynamic volumetric DDoS attack against an AI compute datacenter, bandwidth demand from the attack will affect different network links to varying degrees depending on the attack target links, and changing attacks vectors will stress the attack detection at endpoints. In response, PDP could quickly and accuracy identify attack traffic in the network while OTP dynamically allocates capacity to prevent attack traffic from disrupting critical links used by benign traffic.

3.2 An Illustrative Example

The above-mentioned motivating application classes exhibit two general properties: *(i)* they consider multiple traffic classes and seek to enforce radically different forwarding policies for each class, and *(ii)* they are limited by particular topological structures as traffic in each class shifts.

To concretely illustrate these properties, we consider a simple dumbbell network shown in Figure 1 and the following traffic scenario. Suppose that originally the network has been configured (*e.g.,* to meet some initial traffic demand) as shown by the dashed lines in Figure 1a where each line represents an optical link with 1 Gbps capacity. Now suppose the network faces the traffic demands also shown in Figure 1a with distinct high-priority (light green) and low-priority (dark blue) traffic. In particular, from A to C we need to forward 1 Gbps of high-priority traffic and 2 Gbps of low-priority traffic while from B to D we need to forward 2 Gbps of high-priority traffic and 2 Gbps of low-priority traffic. For example, the high-priority traffic could represent benign traffic while the low-priority traffic represents packets flagged by PDP as potentially being part of a DDoS attack. Alternatively, high-priority traffic could be user web traffic while low-priority traffic could associated with backup and synchronization tasks. The goal is to forward all high-priority traffic with zero loss, dropping or limiting low priority traffic as needed.

Limits of PDP-only Approaches. Programmable switches are fundamentally limited in their ability to control a network's topological behavior; that is, *PDP has the intelligence required to alter a network's traffic forwarding behavior only*. Though PDP can select among a switch's set of candidate output ports, each port must already have a link provisioned and active. Also, PDP can encapsulate and decapsulate (by modifying headers), but the tunnels (*e.g.,* VLANs, VXLANs) must already be provisioned in all other relevant network devices. In short, although several proposals leverage programmable switches [46, 80, 81, 86], they are fundamentally constrained by the limited set of actions that could be applied on a per-packet basis *e.g.,* in switch hardware.

Figure 1b considers the case where PDP is used to classify high- and low-priority traffic and make different forwarding decisions for each class. In particular, nodes A and B can drop all low-priority traffic to reserve as much bandwidth as possible for high-priority traffic on the bottleneck link (E to F). However, even in this case, a total of 3 Gbps must be forwarded between E and F leading to loss of high-priority traffic since the available capacity is only 2 Gbps.

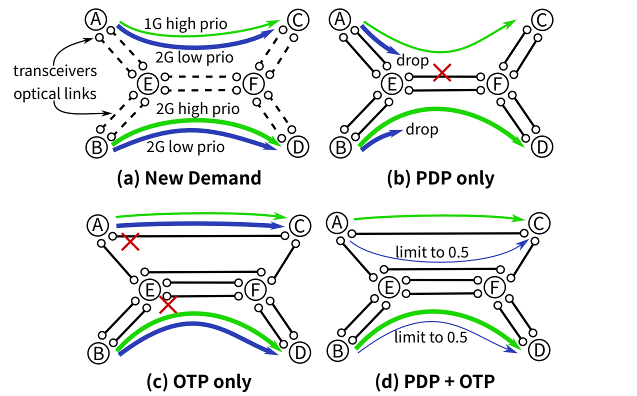


Figure 1: Example of meeting new demand for two different priorities of traffic (shown by green and blue arrows). Both PDP and OTP are required to ensure no high-priority traffic is dropped.

Limits of OTP-only Approaches. Programmable optics are fundamentally limited in their ability to control a network's traffic forwarding behavior; that is, *OTP has the ability to alter a network's topology behavior only*. Though OTP can select among a set of candidate links and capacities to activate, it cannot of its own perform fine-grained forwarding decisions to manage how specific traffic utilizes the updated capacities [59].

Figure 1c considers the case where OTP is used to change the optical topology between nodes, but nodes are not able to separate high and low priority traffic. In particular, the same transceivers can be used to form a new link directly between A and C as well as an additional link between E and F. However, even then high-priority traffic is dropped since the demand on the new link from A to C is 3 Gbps whereas the available capacity is only 1 Gbps. (A similar situation occurs for the traffic between B and D.)

Potential of PDP+OTP. The joint consideration of the two approaches opens up a new dimension to boost the agility of applications: programmatically control both forwarding and topological behavior of the network. To illustrate, Figure 1d considers the potential of a unified PDP + OTP approach where PDPs at each node are able to classify traffic priorities *and* OTP is able to modify the optical-layer topology between nodes. In this case all high-priority traffic is forwarded without any loss by classifying traffic at node A and sending only high-priority traffic over the new link to C. Moreover by applying rate limiting to low-priority traffic at both A and

B, the link between E and F has sufficient capcity to forward all high-priority traffic on the path from B to D.

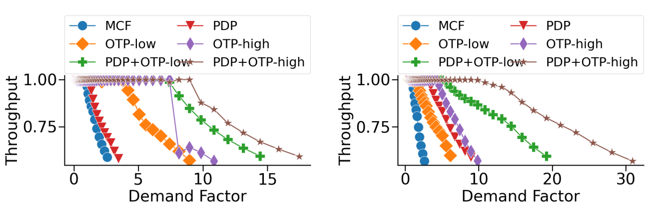
3.3 Empirical Examples

To substantiate the benefits empirically, we run packet-level simulations[1] based on two concrete examples from the example multi-commodity flow (MCF)-based TE and DDoS mitigation applications introduced in § 3.1.

Dataset. We use the DARPA SEARCHLIGHT data set [13], which consists of two topologies, namely *heavy dumbbell* with ten clients and three servers separated by three intermediate hops, and *tired topology* with five servers connected to twelve clients via a tree-like structure. (We observe qualitatively similar results for both topologies and hence only show results for the heavy dumbbell topology due to space limitations.) The data set also includes realistic emulated traffic for several traffic classes atop the two topologies, including interactive sessions (*e.g.,* cloud-based collaborative document editing and video/voice teleconferencing) and non-interactive video streaming sessions.

Methodology. To model the performance of PDP and OTP for MCF-based TE, we selected one class of traffic and considered it *high-priority* and the other classes *low-priority*. To model the performance of PDP and OTP for DDoS, we aggregated all traffic classes into one group and introduce a new class of malicious DDoS attack traffic. In this scenario, the benign traffic is high-priority and the attack traffic is not.

Approaches compared. We compared six different approaches to both applications as follows: *(i)* a baseline multi-commodity flow (**MCF**) algorithm; *(ii)* a **PDP**-only approach that clears all high-priority traffic before sending other traffic; *(iii-iv)* **OTP-low** and **OTP-high** approaches that use OTP to create shortcut links around bottlenecks while assuming two and four available unused transponders respectively at core nodes for activating new bandwidths; and *(v-vi)* **PDP + OTP-high** and **PDP + OTP-low** approaches that combine packet-level forwarding policies with the OTP flexibility enabled by the OTP-low and OTP-high approaches respectively.



(a) Traffic engineering **(b) DDoS mitigation**

Figure 2: Comparison of throughput achieved by PDP, OTP, and PDP+OTP on two topologies (from [13]) as the total traffic volume (demand factor) increases.

Takeaways. Figure 2 shows the throughput performance for both applications as we increase the scale of traffic demand (*i.e.,* the *demand factor*) relative to the dataset's original traffic linearly for all traffic classes in the heavy dumbbell topology. On the one hand, for both applications, the baseline MCF experiences throughput degradation at low demand factors (*e.g.,* 2×) indicating it is unable to scale to higher TE performance requirements or larger volumetric DDoS attack volumes without trading off reduced throughput. On the other hand, our proposed PDP + OTP approaches retain near-perfect throughput up to higher demand factors indicating they enable TE to scale to higher traffic volumes and can effectively mitigate larger volumetric DDoS attacks. In particular, the combined benefit of PDP+OTP in satisfying increased demand with no throughput degradation was up to 7.4× for the TE application and up to 7.62× for the DDoS mitigation application. Overall, these benefits of OTP+PDP hold for all topologies and applications in general, and clearly demonstrate how a unified PDP+OTP approach can scale TE's throughput as well as minimize the impact of DDoS attacks (on benign traffic) in particular.

[1]We thank authors of [60] for sharing their simulation framework.

4 Technology push

Our optimism to achieve joint programmability of optical and data planes to simultaneously achieve the envisioned temporal flexibiltiy and spatial adaptability is motivated by recent trends in *runtime programmability*.

Trends in Programmable Optics. First is the innovation in silicon photonics-based NICs and switches [53]. Unlike the traditional electrical technologies, the silicon photonics technology is poised to use less energy and can transmit data at faster rates. At the same time, the fear of silicon reaching its potential limit due to the end of Moore's law is also rising. Consequently, several hardware experts believe that one of the possible ways forward is to build domain-specific runtime architectures. In light of this, we posit that customized runtime optical hardware will be on the rise. This is evident from recent efforts such as Lightelligence [6], Optalysys [9], and Ayar Labs [1] that seek to design optical systems for targeted applications (*e.g.,* AI/ML). Moreover, industry trends also indicate arise of optics for network management [23].

Second, the paradigm of resource disaggregation is also on the rise due to benefits such as efficient scaling and rapid deployment [74]. As a result, the key onus to ensure the success of such efforts is on the networking community, in general, and physical layer/optical community, in particular. In response, several operators in the optical community started the Open and Disaggregated Transport Network (ODTN) effort [8], paving the way for Disaggregated Optical Networks.

Finally, exposing the optics to the higher layers has also been shown to aid operators in preventing link failures [69], combating DDoS [60], and designing better traffic engineering techniques [29, 59, 68, 87]. In addition, leveraging free-space optics in datacenters has been shown to reduce latency for intra-datacenter transfers [30]. Efforts have also looked into the traffic locality and structure of bandwidth-hungry machine learning and Big Data applications. In their attempt to making workloads and communication predictable, these efforts have resulted in recent innovations in runtime programmable optics [78, 82].

Trends in Programmable Dataplanes. Driven by the increasing performance requirements of today's applications, a heightened security impact of networked applications, as well as rapid increases in scale and complexity of network infrastructures and traffic, a rich body of prior work seeks to increase temporal flexibility of packet forwarding. The first efforts known now as software-defined networking (SDN) [12, 26, 49, 57] enabled flexible reactions to the arrival of new flows (*e.g.,* installing custom forwarding rules). Recently, advances in packet-processing technologies like programmable switches [5, 10, 19] along with general purpose interfaces for packet-level programming of heterogeneous data plane components (*e.g.,* switches, virtual switches, NICs) [2, 7, 18, 71] have enabled fine-grained temporal flexibility in the data plane. For example, DRILL [31] makes per-packet load balancing decisions.

Recently, fungible data paths [79, 80] and other *runtime programmable* efforts [22] seek to further expand temporal flexibility of packet processing by enabling data plane programs to be changed incrementally and arbitrarily at runtime. This leads to a notion of "full-stack" programmability [73, 77] where the once fixed-function layers of control and data plane both become temporally flexible under a unified interface within the still fixed confines of the lower (*e.g.,* optical) layer.

Our position is that by extending the programmability in a vertical manner to encompass the optical layer, the agility of network infrastructures to support a wide-variety of emerging applications could be magnified further, especially when spatial adaptability is taken into account.

5 ShapeShifter: A Unified Programmable Network Core

Building on these trends in programmable optics and switches, we propose ShapeShifter—a unified approach to integrating programmability in both packet and optical layers and simultaneously realizing spatial adaptability and temporal flexibility. We start with an outline of the fundamental components of ShapeShifter, as depicted in Figure 3. Next, we focus on the practical hurdles associated with implementing these components and suggest possible solutions, leaving the implementation and evaluation of ShapeShifter for future work.

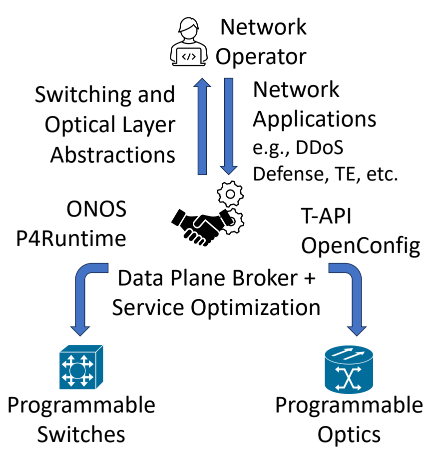


Figure 3: Building blocks of ShapeShifter.

5.1 Packet-optical data plane coordination

The first challenge is the absence of coordination mechanisms at the data plane level to seamlessly connect the workflow between programmable switches and optical equipment (*e.g.,* optical line systems or OLS, and transponders) to support the needed dynamism for ShapeShifter. Concretely, building ShapeShifter requires a set of common data plane mechanisms and interfaces to coordinate actions of optical and programmable switches. While community efforts such as ODTN [8] have made progress to tackle these issues (*e.g.,* exposing operations support systems or OSS that provide performance and security management of OLS via TAPIs), to our knowledge, we currently lack unified interfaces to connect optical gear and programmable switches, as well as mechanisms to jointly program switch ASICs and wavelengths [44].

Emerging solutions, such as NVIDIA Spectrum-X and NVIDIA Quantum-X silicon photonics networking switches, demonstrate the potential of integrating optical and packet-level switching on a unified platform. However, even these advanced devices require a coordination layer to fully exploit their programmable silicon photonics capabilities. One possible direction is to design a broker that will enable reconfigurability at the data plane by coordinating with the relevant optical components, including integrated systems like Spectrum-X and Quantum-X, and programmable switches in response to application requirements. For example, the broker can coordinate the forwarding behavior (for a given topology) by leveraging the ODTN P4Runtime interface, while simultaneously orchestrating the dynamic configuration of optical paths through Spectrum-X/Quantum-X interfaces and established protocols such as TAPIs (over RESTCONF) and OpenConfig (over NETCONF).

5.2 Packet-optical control plane abstractions

The second challenge is the lack of a central coordination and control capability to transparently transform existing as well as potentially new cross-layer applications written by operators into concrete data plane configurations that are mindful of the performance of the network as well as those constraints imposed by programmable optical gear and switches.

To address this challenge, we plan to investigate an integrative, cross-layer abstraction to *(a)* translate application requirements into data plane configurations, *(b)* identify optimal allocation of switch (*e.g.,* ASICs) and optical (*e.g.,* wavelengths) resources, and *(c)* instantiate those allocations on the relevant switches and optics using the data plane broker.

One possible direction to address *(a)* is to develop a new network policy language, (*e.g.,* NetCore [57]), language extension (*e.g.,* how SNAP [12] extends NetCore), or API (*e.g.,* SOL [36]) that integrates OTP primitives alongside traditional forwarding-behavior primitives. A key challenge in this direction is how to design policy primitives that capture spatial adaptability enabled by OTP. For example, forwarding policies specified in NetCore could be annotated with additional performance-level information such as the bandwidth available for each particular policy or physical isolation constraints between policies.

At the core of *(b)* is the classical optimization problem (§ 1): how to best allocate wavelengths and switch resources on network paths in such a way as to satisfy the required performance and operational properties while ensuring high efficiency for the allocated paths? In practice, this kind of optimization problem cannot be solved within a short period of time using a state-of-the-art optimizer (*e.g.,* Gurobi). We envision two directions to address *(b)*: reducing the search space or relaxing the optimization objective. In ShapeShifter, we plan to *(i)* explore rounding-based approximations and clustering techniques to reduce the search space, and *(ii)* develop heuristics for faster convergence to near-optimal solutions.

6 Discussion

The community can come together to tackle some of the practical challenges listed above through the following actions.

To the Academic Community. The assumption of a *stable physical layer* model by the networking community is at odds with programmable optics assumed in ShapeShifter. Changing this assumption and quantifying its benefits in an operational setting requires measurement-driven innovations including *(i)* creation of end-to-end measurement capabilities (*e.g., optical layer traceroute*) that can offer visibility into several optical devices in a network path, *(ii)* unified interfaces to expose the measurements from the optical layer to the higher layers of the network stack, and *(iii)* ShapeShifter to dynamically reprogram data plane devices (*e.g.,* reprogram Tofino stage(s) in the face of Q-drop at the optical layer) to benefit applications. This would also facilitate the creation of community-wide datasets that simulataneusly provide information about both the packet and optical layers.

Building these innovations, however, is fraught with challenges. First, building an optical layer traceroute requires participation from operators from several constituents (*e.g.,* cloud providers, transit providers, etc.). Second, building ShapeShifter calls for expertise/collaboration among optics and networking researchers. Third, we posit that the fate of the envisioned measurement tools will be similar to layer-3 traceroute due to privacy and security reasons (*e.g.,* blocking/dropping measurements, malicious intent to map the wavelength allocation in a network, etc.). Assuming participation from network operators, one way to address this challenge is to build an enclave (similar to secure containers in Intel SGX) in optical devices where SNMP or TL1 could be used to query the devices and provide responses *without* violating privacy and security concerns of operators.

To the Funding Agencies. Large-scale testbeds for prototyping network applications atop programmable switches (*e.g.,* FABRIC [4]) as well as with capabilities to conduct optical experiments (*e.g.,* COSMOS [3]) abound. However, similar to the isolated nature of the two communities, the current testbeds suffer from the packet-optical chasm when it comes to their ability to offer control of packet and optical layers to end users. This problem is aggravated by the lack of OTP-ready testbeds in the community (§ 1). Consequently, we need concerted efforts from the funding agencies to support the creation of new (or extension of existing) community-wide testbeds for a programmable network core in which both switches *and* optics could be programmed simultaneously.

Acknowledgments

We thank the anonymous reviewers for their constructive feedback. This work is supported by the National Science Foundation through CNS-2145813, OAC-2126281, SaTC-2132651, CICI-2319944, and CNS-2212590, and the Internet Society Foundation (ISOC) grants. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of NSF or ISOC.

References

[1] [n. d.]. Ayar Labs. https://ayarlabs.com/.
[2] [n. d.]. Behavioral Model Repository. https://github.com/p4lang/behavioral-model. Accessed: 2023-03-05.
[3] [n. d.]. COSMOS Testbed. https://cosmos-lab.org/.
[4] [n. d.]. FABRIC Testbed. https://fabric-testbed.net/.
[5] [n. d.]. Intel Tofino Series. https://www.intel.com/content/www/us/en/products/details/network-io/programmable-ethernet-switch/tofino-series.html. Accessed: 2023-02-20.
[6] [n. d.]. Lighttelligence. https://www.lightelligence.ai/.
[7] [n. d.]. NPL-Open, High-level language for developing feature-rich solutions for programmable networking platforms. https://nplang.org/. Accessed: 2023-03-08.
[8] [n. d.]. Open Disaggregated Transport Network. https://www.opennetworking.org/odtn/. Accessed: 2023-06-26.
[9] [n. d.]. Optalysys. https://optalysys.com/.
[10] [n. d.]. Trident4 / BCM56880 Series. https://www.broadcom.com/products/ethernet-connectivity/switching/strataxgs/bcm56880-series. Accessed: 2023-05-04.
[11] Firas Abuzaid, Srikanth Kandula, Behnaz Arzani, Ishai Menache, Matei Zaharia, and Peter Bailis. 2021. Contracting Wide-area Network Topologies to Solve Flow Problems Quickly.. In *NSDI*. 175–200.
[12] Mina Tahmasbi Arashloo, Yaron Koral, Michael Greenberg, Jennifer Rexford, and David Walker. 2016. SNAP: Stateful network-wide abstractions for packet processing. In *Proceedings of the 2016 ACM SIGCOMM Conference*. 29–43.
[13] Calvin Ardi, Connor Aubry, Brian Kocoloski, Dave DeAngelis, Alefiya Hussain, Matt Troglia, and Stephen Schwab. 2022. The DARPA SEARCHLIGHT Dataset of Application Network Traffic. In *Proceedings of the 15th Workshop on Cyber Security Experimentation and Test*. 59–64.
[14] Hitesh Ballani, Paolo Costa, Raphael Behrendt, Daniel Cletheroe, István Haller, Krzysztof Jozwik, Fotini Karinou, Sophie Lange, Kai Shi, Benn Thomsen, and Hugh Williams. 2020. Sirius: A Flat Datacenter Network with Nanosecond Optical Switching. In *SIGCOMM '20: Proceedings of the 2020 Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication, Virtual Event, USA, August 10-14, 2020*, Henning Schulzrinne and Vishal Misra (Eds.). ACM, 782–797. doi:10.1145/3387514.3406221
[15] Tom Barbette, Chen Tang, Haoran Yao, Dejan Kostic, Gerald Q Maguire Jr, Panagiotis Papadimitratos, and Marco Chiesa. 2020. A High-Speed Load-Balancer Design with Guaranteed Per-Connection-Consistency.. In *NSDI*. 667–683.
[16] Gilad Baruch, Moran Baruch, and Yoav Goldberg. 2019. A little is enough: Circumventing defenses for distributed learning. *Advances in Neural Information Processing Systems* 32 (2019).
[17] Divyashri Bhat, Jason Anderson, Paul Ruth, Michael Zink, and Kate Keahey. 2019. Application-based QoE support with P4 and OpenFlow. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*. IEEE, 817–823.
[18] Pat Bosshart, Dan Daly, Glen Gibb, Martin Izzard, Nick McKeown, Jennifer Rexford, Cole Schlesinger, Dan Talayco, Amin Vahdat, George Varghese, et al. 2014. P4: Programming protocol-independent packet processors. *ACM SIGCOMM Computer Communication Review* 44, 3 (2014), 87–95.
[19] Pat Bosshart, Glen Gibb, Hun-Seok Kim, George Varghese, Nick McKeown, Martin Izzard, Fernando Mujica, and Mark Horowitz. 2013. Forwarding metamorphosis: Fast programmable match-action processing in hardware for SDN. *ACM SIGCOMM Computer Communication Review* 43, 4 (2013), 99–110.
[20] Guodong Chen, Filip Hácha, Libor Váša, and Mallesham Dasari. 2025. TVMC: Time-Varying Mesh Compression Using Volume-Tracked Reference Meshes. (2025).
[21] Angela L. Chiu, Gagan Choudhury, George Clapp, Robert Doverspike, Mark Feuer, Joel W. Gannett, Janet Jackel, Gi Tae Kim, John G. Klincewicz, Taek Jin Kwon, Guangzhi Li, Peter Magill, Jane M. Simmons, Ronald A. Skoog, John Strand, Ann Von Lehmen, Brian J. Wilson, Sheryl L. Woodward, and Dahai Xu. 2012. Architectures and Protocols for Capacity Efficient, Highly Dynamic and Highly Resilient Core Networks (Invited). *IEEE/OSA Journal of Optical Communications and Networking* 4, 1 (January 2012), 1–14.
[22] Yong Feng, Zhikang Chen, Haoyu Song, Wenquan Xu, Jiahao Li, Zijian Zhang, Tong Yun, Ying Wan, and Bin Liu. 2022. Enabling In-situ Programmability in Network Data Plane: From Architecture to Language. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*. 635–649.
[23] Mark Filer, Jamie Gaudette, Monia Ghobadi, Ratul Mahajan, Tom Issenhuth, Buddy Klinkers, and Jeff Cox. 2016. Elastic optical networking in the Microsoft cloud. *Journal of Optical Communications and Networking* 8, 7 (2016), A45–A54.
[24] Klaus-Tycho Foerster, Long Luo, and Manya Ghobadi. 2020. Optflow: A Flow-based Abstraction for Programmable Topologies. In *Proceedings of the Symposium on SDN Research*. 96–102.

[25] Alex Forencich and George C Papen. 2019. O-Net: an" open" optical networking framework. In *Proceedings of the ACM SIGCOMM 2019 Workshop on Optical Systems Design*. 1–1.

[26] Nate Foster, Rob Harrison, Michael J Freedman, Christopher Monsanto, Jennifer Rexford, Alec Story, and David Walker. 2011. Frenetic: A network programming language. *ACM Sigplan Notices* 46, 9 (2011), 279–291.

[27] Martin Geddes. 2016. No Moore's Law for Networks. https://themobilecentury.com/moores-law-for-networks/. Accessed: 2023-06-26.

[28] Thomas Gerard, Kari Clark, Adam Funnell, Kai Shi, Benn Thomsen, Philip Watts, Krzysztof Jozwik, Istvan Haller, Hugh Williams, Paolo Costa, et al. 2021. Fast and uniform optically-switched data centre networks enabled by amplitude caching. In *2021 Optical Fiber Communications Conference and Exhibition (OFC)*. IEEE, 1–3.

[29] Monia Ghobadi and Ratul Mahajan. 2016. Optical layer failures in a large backbone. In *Proceedings of the 2016 Internet Measurement Conference*. 461–467.

[30] 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*. 216–229.

[31] Soudeh Ghorbani, Zibin Yang, P Brighten Godfrey, Yashar Ganjali, and Amin Firoozshahian. 2017. Drill: Micro load balancing for low-latency data center networks. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*. 225–238.

[32] Arpit Gupta, Rob Harrison, Marco Canini, Nick Feamster, Jennifer Rexford, and Walter Willinger. 2018. Sonata: Query-driven streaming network telemetry. In *Proceedings of the conference of the ACM Special Interest Group on Data Communication (SIGCOMM)*. 357–371.

[33] Matthew Nance Hall, Paul Barford, Klaus-Tycho Foerster, Manya Ghobadi, William Jensen, and Ramakrishnan Durairajan. 2021. Are WANs ready for optical topology programming?. In *Proceedings of the ACM SIGCOMM 2021 Workshop on Optical Systems*. 28–33.

[34] Matthew Nance Hall and Ramakrishnan Durairajan. 2020. Bridging the Optical-Packet Network Chasm via Secure Enclaves. In *Proceedings of the ACM SIGCOMM 2020 Workshop on Optical Systems Design. ACM*.

[35] Matthew Nance Hall, Guyue Liu, Ramakrishnan Durairajan, and Vyas Sekar. 2020. Fighting fire with light: Tackling extreme terabit ddos using programmable optics. In *Proceedings of the Workshop on Secure Programmable Network Infrastructure*. 42–48.

[36] Victor Heorhiadi, Michael K Reiter, and Vyas Sekar. 2016. Simplifying Software-Defined Network Optimization Using SOL. In *13th USENIX Symposium on Networked Systems Design and Implementation (NSDI 16)*. 223–237.

[37] Eric M Hutchins, Michael J Cloppert, Rohan M Amin, et al. 2011. Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. *Leading Issues in Information Warfare & Security Research* 1, 1 (2011), 80.

[38] Sushant Jain, Alok Kumar, Subhasree Mandal, Joon Ong, Leon Poutievski, Arjun Singh, Subbaiah Venkata, Jim Wanderer, Junlan Zhou, Min Zhu, et al. 2013. B4: Experience with a globally-deployed software defined WAN. *ACM SIGCOMM Computer Communication Review* 43, 4 (2013), 3–14.

[39] Su Jia, Xin Jin, Golnaz Ghasemiesfeh, Jiaxin Ding, and Jie Gao. 2017. Competitive Analysis for Online Scheduling in Software-Defined Optical WAN. In *INFOCOM*. 1–9.

[40] 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 *ACM SIGCOMM*. 87–100.

[41] Min Suk Kang, Soo Bum Lee, and Virgil D Gligor. 2013. The crossfire attack. In *2013 IEEE symposium on security and privacy*. IEEE, 127–141.

[42] Naga Katta, Mukesh Hira, Changhoon Kim, Anirudh Sivaraman, and Jennifer Rexford. 2016. Hula: Scalable load balancing using programmable data planes. In *Proceedings of the Symposium on SDN Research*. 1–12.

[43] Elie F Kfoury, Jorge Crichigno, Elias Bou-Harb, David Khoury, and Gautam Srivastava. 2019. Enabling TCP pacing using programmable data plane switches. In *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*. IEEE, 273–277.

[44] Ramana Rao Kompella, Albert Greenberg, Jennifer Rexford, Alex C Snoeren, and Jennifer Yates. 2005. Cross-layer Visibility as a Service. In *Proc. of fourth workshop on Hot Topics in Networks (HotNet-IV)*.

[45] Steven SW Lee and Kwan-Yee Chan. 2019. A traffic meter based on a multicolor marker for bandwidth guarantee and priority differentiation in SDN virtual networks. *IEEE Transactions on Network and Service Management* 16, 3 (2019), 1046–1058.

[46] Zaoxing Liu, Hun Namkung, Georgios Nikolaidis, Jeongkeun Lee, Changhoon Kim, Xin Jin, Vladimir Braverman, Minlan Yu, and Vyas Sekar. 2021. Jaqen: A High-Performance Switch-Native Approach for Detecting and Mitigating Volumetric DDoS Attacks with Programmable Switches. In *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, Michael Bailey and Rachel Greenstadt (Eds.). USENIX Association, Berkeley, California, USA, 3829–3846. https://www.usenix.org/conference/usenixsecurity21/presentation/liu-zaoxing

[47] Chuan Ma, Jun Li, Kang Wei, Bo Liu, Ming Ding, Long Yuan, Zhu Han, and H Vincent Poor. 2023. Trusted ai in multiagent systems: An overview of privacy and security for distributed learning. *Proc. IEEE* 111, 9 (2023), 1097–1132.

[48] Carmen Mas Machuca, Sai Kireet Patri, and Saquib Amjad. 2022. Long-term Capacity Planning in Flexible Optical Transport Networks. In *2022 Optical Fiber Communications Conference and Exhibition (OFC)*. IEEE, 1–3.

[49] Nick McKeown, Tom Anderson, Hari Balakrishnan, Guru Parulkar, Larry Peterson, Jennifer Rexford, Scott Shenker, and Jonathan Turner. 2008. OpenFlow: enabling innovation in campus networks. *ACM SIGCOMM computer communication review* 38, 2 (2008), 69–74.

[50] Serge Melle. 2021. Closing the gap to the Shannon Limit in real-world networks. https://www.nokia.com/blog/closing-the-gap-to-the-shannon-limit-in-real-world-networks/. Accessed: 2023-06-26.

[51] Serge Melle, Thierry Zami, Oriol Bertran-Pardo, and Bruno Lavigne. 2021. Comparing IP-optical architectures & WDM transport technologies in metro, regional and long-haul networks. In *2021 Optical Fiber Communications Conference and Exhibition (OFC)*. IEEE, 1–3.

[52] Rui Miao, Hongyi Zeng, Changhoon Kim, Jeongkeun Lee, and Minlan Yu. 2017. Silkroad: Making stateful layer-4 load balancing fast and cheap using switching asics. In *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*. 15–28.

[53] G Michelogiannakis, Y Shen, X Meng MY Teh, B Aivazi, T Groves, J Shalf, M Glick, M Ghobadi, L Dennison, and K Bergman. 2019. Bandwidth steering for hpc using silicon nanophotonics. ACM. In *IEEE Supercomputing Conference (SC)*, Vol. 10.

[54] Chris Misa, Ramakrishnan Durairajan, Arpit Gupta, Reza Rejaie, and Walter Willinger. 2024. Leveraging Prefix Structure to Detect Volumetric DDoS Attack Signatures with Programmable Switches. In *2024 IEEE Symposium on Security and Privacy (SP)*. IEEE Computer Society, 205–205.

[55] Chris Misa, Walt O'Connor, Ramakrishnan Durairajan, Reza Rejaie, and Walter Willinger. 2022. Dynamic Scheduling of Approximate Telemetry Queries. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*. 701–717.

[56] Mitre. 2022. Network Denial of Service: Direct Network Flood. https://attack.mitre.org/versions/v13/techniques/T1498/001/. Accessed 2023-06-26.

[57] Christopher Monsanto, Nate Foster, Rob Harrison, and David Walker. 2012. A compiler and run-time system for network programming languages. *Acm sigplan notices* 47, 1 (2012), 217–230.

[58] Uiara Moura, Miquel Garrich, Heitor Carvalho, Matheus Svolenski, Alexandre Andrade, Amílcar C Cesar, Juliano Oliveira, and Evandro Conforti. 2016. Cognitive methodology for optical amplifier gain adjustment in dynamic DWDM networks. *Journal of Lightwave Technology* 34, 8 (2016), 1971–1979.

[59] Matthew Nance-Hall, Paul Barford, Klaus-Tycho Foerster, and Ramakrishnan Durairajan. 2023. Improving Scalability in Traffic Engineering via Optical Topology Programming. *IEEE Transactions on Network and Service Management* (2023).

[60] Matthew Nance-Hall, Zaoxing Liu, Vyas Sekar, and Ramakrishnan Durairajan. 2024. Analyzing the Benefits of Optical Topology Programming for Mitigating Link-Flood DDoS Attacks. *IEEE Transactions on Dependable and Secure Computing* (2024).

[61] Vladimir Olteanu, Alexandru Agache, Andrei Voinescu, and Costin Raiciu. 2018. Stateless datacenter load-balancing with beamer. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*. 125–139.

[62] Francesco Paolucci, Filippo Cugini, Piero Castoldi, and Tomasz Osiński. 2021. Enhancing 5G SDN/NFV edge with P4 data plane programmability. *IEEE Network* 35, 3 (2021), 154–160.

[63] P Papanikolaou, Kostas Christodoulopoulos, and E Varvarigos. 2016. Joint multi-layer planning of survivable elastic optical networks. In *Optical Fiber Communication Conference*. Optical Society of America, M2K–3.

[64] Paul Pols and Jan van den Berg. 2017. The unified kill chain. *CSA Thesis, Hague* (2017), 1–104.

[65] Arslan Sajid Raja, Sophie Lange, Maxim Karpov, Kai Shi, Xin Fu, Raphael Behrendt, Daniel Cletheroe, Anton Lukashchuk, Istvan Haller, Fotini Karinou, et al. 2021. Ultrafast optical circuit switching for data centers using integrated soliton microcombs. *Nature communications* 12, 1 (2021), 5867.

[66] Naveen Kr Sharma, Ming Liu, Kishore Atreya, and Arvind Krishnamurthy. 2018. Approximating fair queueing on reconfigurable switches. In *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*. 1–16.

[67] Brian Singer, Keane Lucas, Lakshmi Adiga, Meghna Jain, Lujo Bauer, and Vyas Sekar. 2025. On the Feasibility of Using LLMs to Execute Multistage Network Attacks. *arXiv preprint arXiv:2501.16466* (2025).

[68] 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*. 534–546.

[69] Rachee Singh, Manya Ghobadi, Klaus-Tycho Foerster, Mark Filer, and Phillipa Gill. 2018. RADWAN: Rate Adaptive Wide Area Network. In *ACM SIGCOMM* (Budapest, Hungary) *(SIGCOMM '18)*. ACM, New York, NY, USA, 547–560.

[70] Giuseppe Siracusano, Salvator Galea, Davide Sanvito, Mohammad Malekzadeh, Gianni Antichi, Paolo Costa, Hamed Haddadi, and Roberto Bifulco. 2022. Re-architecting traffic analysis with neural network interface cards. In *19th USENIX Symposium on Networked Systems Design and Implementation (NSDI 22)*. 513–533.

[71] Anirudh Sivaraman, Alvin Cheung, Mihai Budiu, Changhoon Kim, Mohammad Alizadeh, Hari Balakrishnan, George Varghese, Nick McKeown, and Steve Licking. 2016. Packet transactions: High-level programming for line-rate switches. In *Proceedings of the 2016 ACM SIGCOMM Conference*. 15–28.

[72] Ahren Studer and Adrian Perrig. 2009. The coremelt attack. In *European Symposium on Research in Computer Security*. Springer, 37–52.

[73] Debnil Sur, Ben Pfaff, Leonid Ryzhyk, and Mihai Budiu. 2022. Full-stack SDN. In *Proceedings of the 21st ACM Workshop on Hot Topics in Networks*. 130–137.

[74] Jason Taylor. 2015. Facebook's data center infrastructure: Open compute, disaggregated rack, and beyond. In *Optical Fiber Communication Conference*. Optical Society of America, W1D–5.

[75] Muhammad Tirmazi, Ran Ben Basat, Jiaqi Gao, and Minlan Yu. 2020. Cheetah: Accelerating database queries with switch pruning. In *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. 2407–2422.

[76] Amir Varasteh, Sai Kireet Patri, Achim Autenrieth, and Carmen Mas-Machuca. 2021. Towards dynamic network reconfigurations for flexible optical network planning. In *2021 Optical Fiber Communications Conference and Exhibition (OFC)*. IEEE, 1–3.

[77] Guohui Wang, TS Eugene Ng, and Anees Shaikh. 2012. Programming your network at run-time for big data applications. In *Proceedings of the first workshop on Hot topics in software defined networks*. 103–108.

[78] Weiyang Wang, Moein Khazraee, Zhizhen Zhong, Zhijao Jia, Dheevatsa Mudigere, Ying Zhang, Anthony Kewitsch, and Manya Ghobadi. 2022. Topoopt: Optimizing the network topology for distributed dnn training. *arXiv e-prints* (2022), arXiv–2202.

[79] Jiarong Xing, Kuo-Feng Hsu, Matty Kadosh, Alan Lo, Yonatan Piasetzky, Arvind Krishnamurthy, and Ang Chen. 2022. Runtime programmable switches. *Proc. NSDI (to appear)* (2022).

[80] Jiarong Xing, Yiming Qiu, Kuo-Feng Hsu, Hongyi Liu, Matty Kadosh, Alan Lo, Aditya Akella, Thomas Anderson, Arvind Krishnamurthy, T. S. Eugene Ng, and Ang Chen. 2021. A Vision for Runtime Programmable Networks. In *HotNets '21: The 20th ACM Workshop on Hot Topics in Networks, Virtual Event, United Kingdom, November 10-12, 2021*. ACM, 91–98. doi:10.1145/3484266.3487377

[81] Jiarong Xing, Wenqing Wu, and Ang Chen. 2021. Ripple: A Programmable, Decentralized Link-Flooding Defense Against Adaptive Adversaries. In *30th USENIX Security Symposium, USENIX Security 2021, August 11-13, 2021*, Michael Bailey and Rachel Greenstadt (Eds.). USENIX Association, 3865–3881. https://www.usenix.org/conference/usenixsecurity21/presentation/xing

[82] Mingran Yang, Alex Baban, Valery Kugel, Jeff Libby, Scott Mackie, Swamy Sadashivaiah Renu Kananda, Chang-Hong Wu, and Manya Ghobadi. 2022. Using trio: juniper networks' programmable chipset-for emerging in-network applications. In *Proceedings of the ACM SIGCOMM 2022 Conference*. 633–648.

[83] Abbas Yazdinejad, Reza M Parizi, Ali Bohlooli, Ali Dehghantanha, and Kim-Kwang Raymond Choo. 2020. A high-performance framework for a network programmable packet processor using P4 and FPGA. *Journal of Network and Computer Applications* 156 (2020), 102564.

[84] Liangcheng Yu, John Sonchack, and Vincent Liu. 2020. Mantis: Reactive programmable switches. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*. 296–309.

[85] Hui Zang, Jason P Jue, Biswanath Mukherjee, et al. 2000. A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks. *Optical networks magazine* 1, 1 (2000), 47–60.

[86] Menghao Zhang, Guanyu Li, Shicheng Wang, Chang Liu, Ang Chen, Hongxin Hu, Guofei Gu, Qianqian Li, Mingwei Xu, and Jianping Wu. 2020. Poseidon: Mitigating volumetric ddos attacks with programmable switches. In *the 27th Network and Distributed System Security Symposium (NDSS 2020)*.

[87] Zhizhen Zhong, Manya Ghobadi, Alaa Khaddaj, Jonathan Leach, Yiting Xia, and Ying Zhang. 2021. ARROW: restoration-aware traffic engineering. In *ACM SIGCOMM*. 560–579.