On the Unprecedented Scalability of the FISSION (Flexible Interconnection of Scalable Systems Integrated using Optical Networks) Datacenter

Ashwin Gumaste, Aniruddha Kushwaha, Tamal Das, Bala Murali Krishna Bheri and Jianping Wang

Abstract—Internet traffic is doubling almost every other year which implies that datacenter (DC) scalability will play a critical role in enabling future communications. In this paper, we propose FISSION (Flexible Interconnection of Scalable Systems Integrated using Optical Networks) – a scalable, fault-tolerant DC architecture based on a switchless optical-bus backplane and carrier-class switches, and its supporting protocol.

The FISSION DC enables unprecedented scalability using affordable optics and standardized electrical switches. It is architecturally bifurcated into sectors that internally have a non-blocking carrier-class switching interconnection structure. Sectors are connected in the switchless backplane using optical-buses. Each sector can receive traffic on all wavelengths (achieved through optical-bus property without any switch reconfiguration) and across all fibers, but a sector transmits on only a group of wavelengths and only in one of the fiber rings in the backplane. The switches function based on an SDN methodology that facilitate mapping of complex protocols and addresses to DC-specific addressing that is scalable and easier to use.

We present an analysis to optimize the FISSION architecture. A simulation model is proposed that (1) compares the FISSION approach to other contemporary designs; (2) provides scalability analysis and protocol performance measurement; and, (3) provides optical layer modeling to validate working of the FISSION framework at high line-rates. Our architecture, which provides 100% bisection bandwidth, is validated by simulation results exhibiting negligible packet loss and low end-to-end latency.

Index Terms—Datacenters, Scalability, SDN, Carrier Ethernet

I. INTRODUCTION

Datacenters are critical Internet resources that manifest as crucial exchange points for information processing, resulting in massive repositories of information. With the surge of data leading to a compounded annual growth rate (CAGR) that doubles almost every other year, the role of the datacenter (DC) is becoming ever-so-important. DC architecture has gained interest in the research community, especially with the meeting of communication, computation and collaborative tools deployed by both application service providers (ASPs) and telecommunication service providers.

It is becoming increasingly important to store, process and act upon large repositories of information within and across DCs in a cloud scenario. A key challenge to DC design is the aspect about architecture scalability. Scaling a DC involves two inter-twined challenges: (1) designing a non-blocking fabric that enables the interconnection of a large number of servers to each other, and, (2) facilitating a fault-tolerant scalable protocol within the DC for fast-switching, address resolution, load balancing and VM migration. The aspect of DC scalability is most important from the perspective of future DC proliferation and will be indicative of the success of large ASPs, especially in cloud environments.

There have been various approaches in literature towards scalable DC design. Most of these have focused on creating a novel interconnection pattern [3, 5, 6] but they have limitations in terms of scalability and performance. This limitation manifests due to the engineering complexity of providing a full non-blocking switch fabric that is difficult to design with large number of pods/ports/server hosts. It is also not economical due to the inability of electronic technology to provide switching beyond 40Gbps line rate (in the electronic domain), leading to a maximum switch fabric of a few Tb/s such as witnessed in [1].

The limitation manifested by electronics could be overcome by the use of optics that can provide switching at much higher line-rates. However, all the approaches towards an optical switch-centric DC are early designs that are difficult to realize in practice or are plagued by performance issues due to nascent or sometimes slow-speed optical switching technology. Many contemporary approaches such as [2-7] aim at DCs with optical switches as a key switching element, but such designs are not immediately practical to deploy (see Section II) due to: (a) slow optical switching technologies, and, (b) an upper-bound on the number of wavelengths available for switching.

Fig. 1. The sector architecture in the FISSION framework. This shows three backplane fiber rings (essentially functioning as optical busses) connecting six sectors. Each sector either comprises a bunch of servers, or a gateway to the Internet (via the data-center interconnection point or DCIPs). Each electro-optical switches (EOS) interface between each sector and the backplane.

Hence, though it is clear that optics holds the answer to DC scalability, it is not well understood as to how we can deploy pragmatic optical networking technology, while
implementing a scalable, fast-switching-capable, fault-tolerant DC architecture.

Our proposal, FISSION, is perhaps the first approach to using pragmatic optics, while maintaining packet-level granularity and without the need for fast optical switching. The FISSION architecture facilitates pods of servers (called **sectors**) that are interconnected via one-to-many optical bus based backplane(s) (see Fig. 1), in a broadcast-and-select interconnection paradigm. The unidirectional point-to-multipoint (partial) bus architecture ensures a switchless backplane and serves as the interconnect architecture between the sectors. The bus architecture is particularly chosen as it scales without limitations on the number of wavelengths, by **simply adding more** concentric fiber buses (aligned as concentric fiber rings in the backplane).

From the perspective of protocol support, migrating Internet traffic onto a DC-oriented protocol is both a challenge and an opportunity. Approaches towards adopting *software defined networking* (SDN) could guide towards a new DC control plane [8]. A DC protocol must: (a) scale to 10,000s of hosts, (b) cater to different types of services, (c) provide a fault-tolerant design, and, (d) facilitate traffic engineering.

Traditional distributed layer-3 protocols do not cater to most of these requirements. In fact, these requirements are more characteristic of a carrier-class transport protocol than a layer-3 best-effort protocol. We propose the use of Carrier Ethernet in either of the MPLS-TP or PBB-TE forms to suffice for the above protocol requirements. Carrier Ethernet has also recently been considered as an SDN underlay by vendors [20].

This paper is organized as follows: Section II surveys the related literature, while Section III introduces the FISSION concept and architecture. Section IV describes specifics related to its architecture and design, as Section V describes our proposed protocol for communication within the FISSION DC. Section VI presents a model to optimize the provisioned traffic within a FISSION DC, while Section VII presents simulation results for: (a) architecture performance (b) optical layer communication and (c) comparison with other DC schemes. Finally, we present some concluding remarks in Section VIII.

II. RELATED WORK AND OUR CONTRIBUTIONS

Several initiatives for a scalable and fault-tolerant DC architecture have been proposed. These are classified into two types: DC architectures and DC protocols. We highlight our contributions in this paper vis-à-vis existing literature.

A. Datacenter Architectures

Hybrid (electrical/optical) and all-optical solutions have been proposed for DC architectures.

Helios [3], is a hybrid architecture that implements the DC backplane using electrical packet switches and MEMS-based circuit switches. It uses a Topology Manager (TM) to continuously read flow-counters from an aggregation switch and compute a traffic matrix, which is then used to calculate the new topology as well as configure optical circuit switches. As the number of servers increase, the traffic matrix size increases, making the architecture difficult to scale.

OSA [2], is an optical DC solution that also benefits from the reconfigurable properties of optical devices to facilitate dynamic set-up of optical circuits. The reconfiguration delay of the MEMS-based switches and WSSs or *Wavelength Selective Switches*, (in the order of several milliseconds) is a bottleneck for both OSA and Helios. In fact, OSA recognizes this delay that further affects latency-sensitive mice flows.

Mordia [4] in some ways is similar to our concept and yet is significantly different. Mordia uses Nistica’s ultrafast WSS and the whole presumption is the very fast switching of the WSS. This particular WSS is built using digital light processing (DLP) technology. In our implementation, we neither require fast switching nor DLP technology that is known to have reliability issues vis-à-vis the more stable liquid crystal on silicon (LCoS) technology. Further Mordia requires a TDMA scheme that is traditionally difficult to implement and signaling-wise complicated. Our protocol in contrast is a flow-centric approach that does not require TDMA type MAC for implementation using more robust carrier-class techniques.

Quartz [25], presents an optical DC architecture, forming a ring by TOR switches and creates a full mesh network using WDM muxes/demuxes. Although this solution provides low end-to-end delay and reduces wiring complexity, its full mesh requirement limits it due to switch size. Quartz admits its design to be used just as an element of a large DC network, rather than as a replacement to the entire DC network.

Architectures c-Through [5], and WaveCube [6], are other approaches towards a hybrid design. WaveCube assumes multipath routing on a torus and dynamic link bandwidth scheduling, both of which are avoided in our scheme. Both require perfect matching-based graph techniques and create tight bounds on wavelength-flow assignment relationships. The bisection bandwidth for WaveCube is at the most 70-80%, whereas it is 100% for the FISSION approach. Tight bounds for perfect matching are not required in our scheme, and further there is no requirement of constant WSS reconfiguration.

Proteus [26] is an all-optical architecture based on MEMS and WSS, establishing direct optical connections between Top-of-Rack (ToR) switches by dynamic reconfiguration. Proteus uses WDM to provide bandwidth flexibility, and is limited to container-sized DCNs constrained by slow reconfiguration time of MEMS and WSS.

The Torus [27] DCN architecture is based on *hybrid optoelectronic packet router* (HOPR), where the HOPR adapts an N-dimensional torus topology. The HOPR consists of an optical packet switch, a label processor, an optical circuit switch, fiber delay links (FDL) and a controller for the OPS. The size of the OPS depends the number of FDLs. It has been reported that such centrally-controlled OPS require at least $m log m$ clock-cycles for reconfiguration [29], where $m$ is the port count of the switch. Thus, for large port-count DCs the expected latency is high.

Lightness [28] presents a DCN architecture based on OPS and optical circuit switching (OCS) technologies, where servers are connected to both OPS and OCS through a ToR. The OCS used in Lightness claims to scale to thousands of input/output ports and can be used to support long-duration data flows, whereas OPS switches are used to support short duration flows. This architecture uses a distributed control for the switch that makes reconfiguration time independent of port-count, but this architecture faces challenges in scalability of the DC as it requires OPS switches of large port count (~1024 x 1024) and relies on advancement in photonic integrated technologies. In FISSION, we do not require OPS
which is a nascent technology. There are issues of reliability of OPS as well as header recognition and switching time. The FISSION architecture is oblivious of these drawbacks.

B. Datacenter Protocols

SEATTLE [9], describes a protocol backward-compatible with Ethernet that facilitates ‘plug-and-play’ operation, allowing for dynamic repositioning of servers and placing no restriction on the underlying network topology. In SEATTLE, data forwarding is based on flat 48-bit MAC addresses. Hence, to provide all-to-all communication, switches are required to maintain entries for every host in the DC. SEATTLE also uses broadcast-based link state protocol for discovering topology that takes considerable time for convergence post a failure. Another issue is that forwarding loops may also exist in the network.

PortLand [10], uses network topology discovery with Location Discovery Protocol. PortLand uses a three-layer fat-tree topology similar to a Clos network. As servers are added to the network, either the port density of the switches need to be increased, or one or more layers of switches need to be added. Hence, in-situ upgradation of the DC is complex.

VL2 [11], provides a fully connected backbone by connecting the aggregation and core switches in a Clos topology. VL2 uses distributed link state protocol for topology discovery, which leads to high convergence time post failures.

DCell [12], was proposed to efficiently connect large number of end-hosts. It provides in-situ upgradation of the DC without affecting the existing network topology. The upper bound on the number of servers that can be added to the DC is limited by the number of network interface cards supported by the servers. The design of DCell is also limited by high wiring complexity and the requirement of end-hosts to switch packets at the cost of reduced bisection bandwidth.

Hedera [13], proposed a dynamic flow scheduling algorithm for multi-rooted fat-tree DC networks.

While OSPF convergence is an issue in terms of latency due to its best-effort nature and hence most providers use carrier-class protocols, we also note that there have been efforts to use BGP as a routing protocol in [30-32] within the DC. Specifically, BGP-based routing through the use of Extended BGP (EBGP) facilitates the same level of control, scalability and performance as a carrier-class protocol due to its explicit nature of policy management and advertising, though to implement such a protocol one does require the routers to be able to adapt to EBGP, which is non-trivial in terms of required architectural changes.

Our approach completely eliminates the need for dynamic flow scheduling, since inter-sector communication is achieved using an optical fiber ring. Logically, there is only one unified backplane for addition of a sector in a FISSION DC, and hence every sector needs to send the inter-sector traffic to one and only one fiber ring. Further, FISSION handles ARP efficiently with the help of a centralized server (called GEMS; see Section V). The concept of handling ARP with the help of a centralized server was also leveraged in SEATTLE and PortLand. However, if the resolving proxy fails, then hosts will be unreachable in SEATTLE, whereas in FISSION if the management plane fails to serve ARP request, the ARP packet is broadcasted to all localized sector-specific entity (LEMS; see Section V) in the DC.

Epilogue: This paper is a comprehensive extension of our earlier works in [14, 16, 17]. In [14], we introduced the FISSION architecture, detailing issues such as scalability, and working, protection was discussed in [16], and engineering aspects in [17]. The architecture in this paper is a significant improvement over the one in [14] – the excess connections have been removed (namely between edge and backbone switches); moreover, this paper details how the backplane is developed and includes a new interconnection architecture. The paper proposes two methods single hop and multihop as techniques for using a switchless backplane. We also show a comprehensive analysis of the FISSION DC architecture, using optimization models, and perform extensive simulations over its architecture, protocol, optical performance, and comparison with related DC architectures.

III. FISSION SYSTEM DESIGN

In this section, we describe the FISSION DC architecture. The FISSION DC broadly consists of two subsystems, namely, the sectors and the backplane. Table 1 summarizes the key parameters in the FISSION DC.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>F</td>
<td>Number of fiber rings.</td>
</tr>
<tr>
<td>σ</td>
<td>Number of wavelengths per ring.</td>
</tr>
<tr>
<td>N</td>
<td>Number of ports in an ES</td>
</tr>
<tr>
<td>α</td>
<td>Number of sub-carriers per super channel.</td>
</tr>
<tr>
<td>γ</td>
<td>Number of super channels per ring.</td>
</tr>
<tr>
<td>H</td>
<td>Number of super channels per sector.</td>
</tr>
<tr>
<td>M</td>
<td>Multiplexing gain at EOS or per wavelength speedup in the backplane.</td>
</tr>
<tr>
<td>W</td>
<td>= ατ, Number of subcarriers per sector.</td>
</tr>
<tr>
<td>n</td>
<td>Number of sectors in each fiber backplane.</td>
</tr>
<tr>
<td>S</td>
<td>Number of servers in the FISSION DC.</td>
</tr>
</tbody>
</table>

Fig. 2. FISSION architecture and layout. There are three fiber rings in the backplane and each ring is divided into 8 sectors. The colored arrows show the add connections to the backplane, while the dotted curves show the drop connections. The first sector is (1, 45), the second one is (1, 90) and so on, with the 8th sector in the first ring labeled (1, 360). The second ring further subdivides 8 more sectors labeled (2, 45) through to (2, 360), while the third ring subdivides 8 more sectors labeled (3, 45) through to (3, 360).

Table 1: Summary of FISSION DC parameters


A. Sector Architecture

The FISSION DC is arranged as sectors that resemble pods of optical MUX/DEMUX along with electrical switches, with each sector physically organized in a single or a group of racks, and sectors back-wired by a series of optical buses. Each sector can receive data from all fiber rings, but can send data into only one ring.

The way the sectors are arranged is as shown in . Sectors are defined by two variables \((f, \theta)\), where \(f\) is the fiber ring into which a sector adds data, and \(\theta\) is the angular position of the sector along that ring, traced through some preseleced reference point. Assume \(n\) sectors along the circumference of a single fiber ring, and at each of the \(n\) locations there are \(F\) sectors (one corresponding to each fiber ring). Each sector receives data from all the \(F\) backplanes but can transmit to only one backplane ring. A backplane is an open fiber ring. The number of sectors per backplane is assumed to be uniform across all the rings.

Each sector consists of edge switches (ES), aggregation switches (AS) and electro-optic switches (EOS). The EOSs are connected to each other via an all-optical backplane. An ES is analogous to a top-of-the-rack (ToR) switch and connects to servers. Multiple ESs are connected to each other in a folded Clos fashion via the ASs. The ASs are connected to the EOS. The EOS connects to the optical backplane.

The optical backplane is a switchless entity comprising of concentrically laid-out optical buses (similar to light-trails [15]) that facilitate one-to-many communication. Each ES is connected to \(N\) servers and internally has an \(N \times N\) switching fabric. ESs are connected to each other via a second row of \(N\) ASs that are of \(\frac{W.M}{N} \times \frac{W.M}{N}\) port count, where \(W\) is the number of wavelengths allotted to a sector for addition in the backplane, and \(M\) is a muxponding gain or a speed up factor for the backplane. We define muxponding gain as follows: Muxponding gain \(M\) is defined as the number of servers mapped on to a single subcarrier (in case of use of coherent optics in the backplane) or to a single wavelength (in case of use of non-coherent optics in the backplane) and indicates the speedup between the server line-rate and the backplane. For sake of simplicity, we assume each subcarrier to be of 10Gbps line-rate and a server with 10Gbps output. To achieve a Clos non-blocking schema, \(N\) ports of the ES are connected to \(N\) different ASs.

A server sends data to an EOS (via the ES-AS); the EOS further maps this data onto one of the available super channels. A superchannel has multiple subcarriers. In the case that there is no muxponding gain, the line-rate of a server is equal to the subcarrier rate. In the case that we do not use superchannels, i.e. without the use of coherent optics, then we simply have wavelengths in the core subscribing to NRZ modulation format.

There are \(n\) \(F\) sectors in the DC, with \(n\) sectors supported by every fiber backplane, and \(F\) fiber ring based backplanes to support the entire DC interconnection. Each sector can have up to \(W.M\) servers, and if we assume \(C\) to be the line-rate of each wavelength, then the total capacity of a sector is \(W.C\). The EOS sends data on a group of wavelengths, and data is tapped at all other sectors downstream through the use of optical couplers. There is a drop coupler and an add coupler at the interconnection of each sector with the backplane.

![Fig. 3. FISSION Multi-sector implementation. This figure details the sector architecture with its Add WSSs, Drop WSS, EOS and aggregate and edge switches. The open-ended connections from the edge switches are to the servers, which are omitted for clarity. The optical signal from each backplane is dropped onto each sector, which is then forwarded to the respective server by the EOS.](image)

We now describe the add and drop portions of the EOS that facilitate the scalable FISSION framework. 

**EOS Drop side:** As shown in Fig. 3, each of the \(F\) fiber rings subtends a drop coupler, whose asymmetric splitting ratio ensures that maximum signal power is retained for pass-through channels.

Each drop coupler is first connected to a \(1 \times F\) splitter that allows the dropped signal to be sent to the \(F\) sectors at that location along the ring(s). Booster optical amplifiers are used to boost the signal to an acceptable level. Each port of the \(1 \times F\) coupler is connected to a \(1 \times \psi\) coupler. \(\psi\) is an engineering variable and adheres to \(1 \leq \psi \leq F\). The choice of \(\psi\) indicates the number of banks with which we design the EOS. The higher the number of banks the more contention we can support. For example, with 2 banks we can support to drop two identical wavelengths (from two different fiber rings) on to the same sector EOS. Each of the \(\psi\) ports of the \(1\psi\) coupler may have a 1x1 switch that can limit the spread of signal to select banks. The \(1\psi\) coupler is then connected to \(\psi\) WSSs of the \(M \times N\) variety where \(M = F\) and \(N = \sigma\). The total number of drop ports = \(\psi \times \sigma\). We have shown in [14, 16, 17] that with \(H + nF - 1\) drop ports a non-blocking combination for static assignment can be achieved.

**Optical Reach:** With coherent optics supporting transponders for very short distances (such as in a DC), a large number of channels are possible, though about 400 wavelengths (with no coherent optics at 25 GHz spacing across the C and L band) or 100 superchannels (with coherent optics) each with 4·16 subcarriers, suffice for even very large DC implementations.

**Drop Coupler Configuration:** To drop channels from a ring to a particular sector, a \(1 \times 2\) splitter with power splitting ratio of \(X:Y\) is used, where the port with \(X\%) power is used to continue the channels in the same ring, whereas the port with \(Y\%) power is used for drop purposes. The \(X\%) port is further connected to a \(1 \times F\) splitter and each port of this splitter is dropped to a sector.

**Received Power Calculation:** Let \(P_{in}\) be the input power of a channel (after EOS at the ingress sector), \(P_{WSS}^{ADD}\) and \(P_{WSS}^{DROP}\) are the insertion losses at Add WSS (at the ingress sector) and Drop WSS (at the egress sector), respectively, and \(P_{coupler}\) be the combined coupler losses, then received channel power at the \(j^{th}\) sector (from the ingress sector) is:
The EOS consists of muxponders (opto-electro-opto devices) that electronically aggregate traffic by mapping these onto a superchannel. The muxponded optical signals are then multiplexed using the WSS and sent into the fiber-ring using the Add Coupler (AC).

B. Backplane Architecture and Scalability

Backplane Principles The FISSION DC backplane supports an optical bus in an open ring configuration.

In a single fiber backplane, a sector is allocated a specific wavelength range and transmits only on those wavelengths.

In a multi-fiber backplane, a sector sends into only one fiber, but can receive from all the fibers. There is no limitation on the number of fibers in the backplane.

As we add more fibers in the backplane, the FISSION framework requires that there be a drop connection from every fiber to each sector. It is further noted that the number of wavelengths available in a fiber has no impact on scalability as the number of servers supported in a DC is the product of the number of wavelengths in a fiber, the number of fibers, and the muxponding gain.

Specifics: The signal added by an ingress sector is available at every other sector in the open optical ring (bus) (see Fig. 2 and Fig. 3) using optical multicast. This implies no optical switch reconfiguration is required in the backplane. In contrast to adding, for drop-portion of the EOS, at every sector, we drop signals from every fiber ring. To scale the FISSION framework, we simply add more sectors. When the number of sectors saturate a fiber ring, we add another fiber ring in the backplane and connect it to the drop EOSs of all the existing sectors.

There is a limitation on the number of sectors per ring (due to wavelength availability), but there is no limitation on the number of rings in the backplane. This is the principle behind unprecedented scalability of the FISSION framework.

Backplane Design: In each of the F fibers, we assume there are \( H \cdot n \) wavelengths or superchannels (with coherent optics) and each superchannel has \( \sigma \) subcarriers. Alternatively, with non-coherent optics we assume \( W = H \cdot n \) wavelengths per sector. At every sector, a fiber has a \( 1 \times F \) power splitter of splitting ratio \( \psi \). Each port of the \( 1 \times F \) coupler is connected to a \( 1 \times \psi \) coupler. The local drop end of the \( 1 \times \psi \) coupler is connected to a series of cascaded Drop WSS (Wavelength selective switch) of varied configuration, between \( 1 \times \sigma \) and \( 1 \times (H + nF - 1) \) to achieve full non-blocking communication [14]. Using standard \( 1 \times 20 \) WSSs [21], it is possible to build a larger WSS to support the required number of drop ports. We can summarize that \( \psi, H \) is a good lower bound to suffice for the Drop WSS. At the add-side, we use an Add WSS cascaded to produce a WSS block of \( H \times 1 \) ports, corresponding to the total traffic being injected by the servers in the sector. The Add WSS is connected to an add coupler in the fiber backplane.

Since the DC is connected to a service provider network, we assume the existence of datacenter interconnection point (DCIP). Multiple DCIPs facilitate plurality of providers.

Examples: Let us consider a DC with each server supporting 10Gbps line rate. We assume that there are 384 channels (per ring at 25GHz spacing) each supporting 40Gbps and each sector can support a maximum 256 servers. Therefore, each fiber ring can now support a maximum of 6 sectors or 1536 servers. As the number of servers increase, new sectors can be added by adding a new fiber ring. Fig. 4 shows the worst case received power of a channel at a sector with increase in the number of servers in the DC by use of asymmetric power splitter of different split ratio. It shows that with the use of a 20:80 splitter, a power penalty of 6-9dB can be reduced in comparison to other splitters.

| Table 2: Scalability of the FISSION Framework. Table 2a (top) gives required number of sectors and rings for a given number of servers and calculated power consumption and cost of entire DC considering cost and power consumption of a unit given in Table 2b (bottom). The power estimates in Table 2 were consolidated from [34-36], whereas the cost figures were compiled from [33] as well as generalized market estimates based on our regular interaction with the vendors. |
|---|---|---|---|---|---|---|
| No. of Servers | 1000 | 10000 | 100000 | 1000000 | 10000000 |
| No. of Sectors | 4 | 40 | 391 | 3907 | 39063 |
| No. of Rings | 1 | 7 | 66 | 652 | 6511 |
| Transceivers per sector | 3584 | 3584 | 3584 | 3584 | 3584 |
| Power Consumption (W) | 90676 | 820680 | 7586754 | 75236898 | 898030827 |
| Cost of Server ($) | 1408.8 | 1357.9 | 1115.2 | 1082.3 | 2078.7 |
| Cost of DC ($) | 5635 | 54315 | 436031 | 4228456 | 81200949 |

![Fig. 4. Received power profile as a function of number of servers for different splitting ratios of the 1xF drop coupler.](image)
Consider a DC with 100,000 servers and each server's physical network adapter can support 10Gbps. We need 391 sectors with each sector supporting 256 servers with an assumption of 384 channels (per ring across C and L band) each supporting 40Gbps. Each fiber ring can now support a maximum of 6 sectors. Hence, to support 391 sectors, we need 66 fiber rings. Each sector requires 64 wavelengths (subcarriers). Each sector requires 16 ESs and 16 ASs each of 16 × 16 configuration for complete interconnection of all servers within a sector.

For more examples, consider Table 2a for scalability. Shown in Table 2a are also cost and power consumption metrics, the basis of which are shown in Table 2b.

IV. FISSION DATACENTER SPECIFICS AND WORKING

We now discuss the specifics of the FISSION DC. To begin, we desire to create a DC of \( R \) servers. Let \( P_{in} \) be the input power from an EOS: let each Add WSS be of \( 1 \times A \) configuration, implying that we need \( [H/A] \) cascaded Add WSSs, and let \( p_{WSS}^{Add} \) and \( p_{WSS}^{Drop} \) be the power losses at each Add WSS and Drop WSS, respectively. Let \( P_{th} \) be the pass loss at each \( 1 \times F \) coupler (drop coupler). Finally, let \( P_{p-t} \) be the pass-through loss at any sector. If \( P_{th} \) is the threshold power required to obtain an acceptable BER, and \( P_{EDFA} \) is the amplifier gain at each node, then:

\[
n \leq \frac{1}{|P_{EDFA} - P_{p-t}|} \left[ P_{in} + \left( \frac{P_{th} - P_{WSS}^{Add} \log \left( \frac{H}{A} \right) - P_{1xF} - P_{WSS}^{Drop}}{\lambda} \right) \right]
\]

In the above equation, \( n \) is dependent on \( H \), which is further dependent on \( N \). Hence, in order to compute \( n \), we need to know \( N \), which can be done in two ways: (a) \( N \) is the number of ports of a given ES and can be assumed as an input parameter; or, (b) the choice of \( N \) impacts the delay experienced in a sector. For sake of completeness, we choose the second method. For an \( N \)-port non-blocking switch, we need \( 2N \) cross-bar ports. Assume the average delay in a single cross-bar is \( \rho \), then,

\[
2\log_{2}N \leq \delta_{ES}
\]

where, \( \delta_{ES} \) is the maximum permissible ES delay. We can now compute \( N \) and hence design the DC using various numerical methods such as curve fitting.

A. Designing a Datacenter

Given an \( N \)-port switch, we are able to design a DC in the following manner. Given \( \delta_{ES} \) and \( \rho \), we are able to compute \( N \). Given the WSS losses and input power profiles as well as amplifier gain, we are able to compute the maximum number of sectors possible for a single fiber. By adding a \( 1 \times F \) coupler in the drop portion, we can recursively balance the threshold power computation relationship and compute the dynamics of the FISSION DC.

B. Bisection Bandwidth Computations

In the previous section, we described the FISSION architecture and in this section we describe the engineering aspects of the architecture, namely the wavelength assignment strategies that determine the resource utilization as well as blocking of a request leading to computation of the bisection bandwidth. To this end, various wavelength strategies are described in Fig. 5.

- In Fig. 5, we have two broad variants in routing strategies – single hop (in which an ingress sector directly communicates to an egress sector) and multihop (in which an ingress sector communicates to an egress sector via one or more intermediate sectors).
- The type of technology used is important, as to whether the receiver or the WSS can be tuned fast enough to achieve any-to-any sector communication, or whether we have to rely on a static assignment.
- The traffic distribution is important – we examine two extremes – a symmetric mix, whereby there is a uniform distribution of traffic in the DC, and an asymmetric mix, whereby there is uneven distribution of traffic and which also considers the worst case traffic pattern.

Theory of wavelength assignment: We now describe a base model for wavelength assignment in the backplane. The model assumes the key role of \( \psi \) banks at the EOS drop and its impact on the wavelength assignment scheme. If every wavelength at line rate \( C \) is sped up by a factor \( M \), then we will have the following arguments:

- The gap between being able to achieve any-to-any sector connectivity and what the \( \psi \) banks can provide in a static configuration is denoted by \( \frac{\psi-1}{\psi} \). Intuitively, this is reasoned as: \( \frac{nF - 1}{\psi} \) represents the maximum number of channels that can be dropped at a sector, while \( \frac{\psi-1}{\psi} \) represents the actual number of channels dropped and hence \( \frac{\psi-1}{\psi} \) denotes the blocking rate.
- With the speed up, the capacity of the backplane is now \( F, \phi, M, C \).
- We propose the following worst case condition for both single hop and multihop that facilitates any random wavelength assignment technique to be effective as long as it meets the following condition: \( \frac{M}{\psi} \geq \frac{\psi n}{\phi - 1} \), where \( \phi \) is the average number of hops. Note that for the case \( \frac{\psi}{\phi} = 1 \), i.e. single hop case, the condition changes to \( \frac{M}{\psi} \geq \frac{\psi n}{\phi - 1} \).
- Let \( Y_{ij} \) denote the event that there is a wavelength available from sector \( S_i \) to \( S_j \), and \( Y_{ij}^{-1} \) is its complement event. We next derive \( P(Y_{ij}) \).

\[
P(Y_{ij}) = P\left( \sum_{i=1}^{H} P_{ji} \right) = \prod_{k=0}^{n-2} (1 - P_{ji})^{nF-2-k}
\]

Above is an implicit equation in \( P_{ji} \), which when solved (say, using numerical methods) will offer solutions to \( P(Y_{ij}) \) and \( P(Y_{ij}^{-1}) \). The event of finding a free wavelength at each hop of a given \( \psi \)-hop path \( (i \rightarrow j_1 \rightarrow j_2 \rightarrow \cdots \rightarrow j_{\psi-1} \rightarrow f) \) from \( S_i \) to \( S_j \) is: \( U_{ij} = Y_{ij}^{-1} Y_{ij} (\prod_{k=1}^{\psi-1} Y_{j_{k+1} j_{k+2}} Y_{j_{\psi} j}) \).
$P(U_d)$, which depends on $P(Y_{ij})$ and $P(Y_{ji})$ and can hence be derived. We note that $P(U_d)$ is independent of the given path. Thus, the average hop count of a connection request between a sector pair is $v = \sum_d d \times P(U_d)$.

With the above insight into the model, we can define various scenarios for the FISSION architecture. The goal of this exercise is to compute the conditions of provisioning a connection as well as conditions under which full bisection bandwidth can be obtained.

1. **Multihop symmetric traffic with fixed receivers (MSTFR)**
   In this case, the mean traffic between two sectors is
   \[
   \frac{CoF}{\text{naf}(\text{naf}-1) + v} \approx \frac{c}{npr} \times \frac{\sum d \times P(U_d)}{\text{naf}}
   \]
   which is less than a single wavelength, especially if $F > 1$. Thus, if a wavelength is not tuned between a transmitting and a receiving sector, then the probability that a wavelength is available between two sectors is $M \times P(Y_{ij})$. This is the provisioning probability for the single hop case, which can be extended to the multihop case by scaling with the hop count factor $v$ we get:
   \[
   P_{\text{MSTFR}} = (\frac{M \times P(Y_{ij})}{v})^v.
   \]
   The provisioning bandwidth in this case is $\frac{v}{v-1}$.
   The end-to-end delay is given by $\Delta (\overline{v}-1)$ where $\Delta$ is the average EOS processing time at a sector.

2. **Multihop symmetric traffic with tunable receivers (MSTTR)**
   In this case, we use tunable receivers/reconfiguration feature of the WSS and define $t_k$ as the time required to tune a receiver (or more appropriately configure a WSS) to the desired wavelength drop at the egress sector. Let us define $B_{ki}$ to be the buffer size at a sector $k$ available for communication from sector $i$ to sector $j$. In this case, if $\frac{B_{ki}}{c} > t_k$, then with a probability of 1, we will be able to achieve no blocking between sectors $S_i$ and $S_j$. In this case, the bisection bandwidth due to multihop would be $M/(\overline{v}-1)$, where $\overline{v}$ is the average hop count across all the flows in the DC. The problem arises if $\frac{B_{ki}}{c} < t_k$ in which case the buffer at the ingress cannot hold the data, while the tuning at the egress happens. In such a case, we can use multihop to compensate the tuning time such that $\sum_{v=1}^{\overline{v}} B_{ki}/c < t_k$, which implies that a connection can be set up under the following event: $Y_{ij} \cap (U_{ij} \Delta (\overline{v}-1) + \frac{B_{ki}}{c} \geq t_k)$. This equation implies the case when there is no wavelength between sectors $S_i$ and $S_j$ and the probability of $v$ hops are incurred, leading to a delay such that the combined delay allows tunability at the egress sector $S_j$. This simplifies to:
   \[
   P_{\text{MSTTR}} = P(Y_{ij}) \times \left[1 - \sum_{v=1}^{\overline{v}} \left(\frac{t_k - \frac{B_{ki}}{c}}{\overline{v}}\right) \left(\frac{P(U_d)}{P(U_d)}\right)^v\right]
   \]
   In this case, the average end-to-end latency is $\Delta (\overline{v}-1)$ and the bisection bandwidth is $M/(\overline{v}-1)$.

3. **Multihop asymmetric traffic with fixed receivers (MATFR)**
   For this case, we consider the worst case traffic, whereby, we find a set of intermediate sectors such that an ingress sector $S_i$ can send all of its traffic to an egress sector $S_j$. Note however that when sector $S_j$ is sending all of its data to $S_i$, this represents an asymmetric traffic pattern. The worst-case is when we view the DC as a bipartite graph with half the number of sectors sending their entire data to the remaining sectors. In this case the probability of provisioning the asymmetric traffic is the same as in case 1 raised to the power of $H$.
   \[
   P_{\text{MATFR}} = (M \times P(Y_{ij}))^H
   \]
   The bisection bandwidth for this case is $M/(\overline{v}-1)$ and the average end-to-end delay is $\Delta (\overline{v}-1)$.

4. **Multihop asymmetric traffic with tunable receivers (MATTR)**
   This case is similar to case 2, just that the probability of provisioning is $H$ identically distributed connections compared to case 2. Hence,
   \[
   P_{\text{MATTR}} = (P_{\text{MSTTR}})^H
   \]
   The end-to-end delay and bisection bandwidth are same as case 2.

5. **Single hop symmetric traffic with tunable receivers (SSTFR)**
   In this case, the receiver can be tuned and require $t_k$ time to tune. The ingress sector has a buffer of size $B_{ki}$, where $k = i$ for this flow, and the system works with full bisection bandwidth if $\frac{B_{ki}}{c} \geq t_k$. In this case, $\overline{v} = 1$ is sufficient for the best case. If however we use the system with $\overline{v} = 1$, then we need instead of an $M \times N$ WSS, an $M \times N$ multicast switch, which can provide contention-free dropping. The bisection bandwidth in this case is always 100%.

6. **Single hop asymmetric traffic with tunable receivers (SSTTR)**
   In this case, the bandwidth in the core is not a concern, the only issue is that of tunability. The joint probability of the worst-case situation is when the sectors are arranged as a bipartite graph. Hence with a probability of 1 we provision the connections in the case of asymmetric traffic if $\frac{B_{ki}}{c} > t_k$. If however, $\frac{B_{ki}}{c} < t_k$ then there will be sizable drop of data and hence we do consider this case for DC design.

7. **Single hop asymmetric traffic with fixed receivers (SSTFR)**
   This is the second most stringent case. For symmetric traffic, we need $\frac{c}{naf}$ data between any two sectors. Taking the ceiling, this implies that we need at the most one wavelength for large $F$ between any two sectors. Further, the probability that this wavelength is being accessed is given by the following argument: In the base case, the probability is $\frac{v}{c}$. Now, with $\psi$ hops and a speed up of $M$ the probability is $P_{\text{SSTFR}} = M \times P(Y_{ij})$. The end-to-end delay is EOS processing delay while the bisection bandwidth is 100%.

8. **Single hop asymmetric traffic with fixed receivers (SATTR)**
   This is the most demanding case from a provisioning perspective. It denotes the case of traffic corresponding to $H/M$ wavelengths being available between any two sectors. This is hence represented by $P_{\text{SATTR}} = (M \times P(Y_{ij}))^H$. The end-to-end delay in this case is the same as case #7 and the bisection bandwidth achievable is 100%.

In Fig. 6, we have shown the different wavelength assignment schemes for different sized DCs with $\psi = 3$ (we have explained why measurements are taken for $\psi = 3$ in section VII). The speed up factor is 4. The WSS configuration time is 3 milliseconds, and when averaged over flows, it leads to a per-packet extra delay of 20μs. Results are for 10K to 1 million servers. The single hop schemes with tunability have 100% bisection bandwidth, while multihop schemes work well for symmetric traffic. The worst-case is multihop with asymmetric traffic. The key takeaway is that bisection bandwidth is close to 100% for most of the cases and in the worst-case it drops down to about 65%.
V. PROTOCOL IN THE FISSION DATACENTER

In this section, we describe a protocol for scalable carrier-class communication in the FISSION DC, based on the dual premise of scalable and flat addressing and carrier-class communication. A key goal of the protocol is to enable homogeneity among a diverse range of services. Flows can be provisioned using user-defined services and the protocol provides for switch/routing/fault tolerance. As a representative example, the protocol is capable of indistinguishably routing on IPv6 tunnels as well as Ethernet E-LINE between servers and the DCIP.

A. Protocol Working

The protocol acts within the DC boundaries i.e. between the ES ports (servers) and the DCIP(s). Any flow that enters the DC is converted into DC-specific addresses. At the edge of the DC-boundary, incoming protocol identifiers are mapped to DC-specific labels. We adopt the recently proposed segment routing approach [19, 22] as a philosophy for label creation and forwarding. Segment routing is a technique to identify network segments and interfaces using domain-specific identifiers, and conjoin these to obtain a source-routed path to the destination.

The protocol works as follows: At the boundaries of the DC, incoming “relevant” identifiers are extracted from packet headers and mapped to a series of flow-tables. Since the DC is a controlled environment from the perspective of the address space, a match of an incoming packet identifier with an entry in one of the flow-tables is guaranteed. The flow tables are similar to the ones defined in the various SDN implementations [23]. Each flow table is for a specific protocol identifier and consists of two columns. The first column contains pre-stored identifiers (IP addresses, MAC addresses, VLANs, etc.), while the second column contains “segment routing” for each corresponding segment identifier value. An optional third column provides a protection segment.

A controller performs three functions: (a) Compute segment IDs of all the interfaces in the DC; (b) Update the flow tables with the corresponding segment IDs, and, (c) Allocate wavelengths in the backplane.

Due to the restricted address space of a DC, we assume a proprietary addressing scheme. The controller first allocates unique addresses to each port of every ES/AS/EOS in the DC. It does so as follows:

1. For every sector, we construct a binary tree (for that sector) with an imaginary root of a tree whose leaves are the ports of all the ES in a sector and which traces back to include the AS and EOS (see ).
2. There is a separate binary tree for every sector. In addition, each switch is represented by a portion of the binary tree that makes up the sector.
3. Since ES’s and AS’s are of \(N \times N\) and \(\lceil W \cdot M \rceil \times \lceil W \cdot M \rceil\) port count, respectively, the controller assumes a cross-bar structure within the ES/AS to create the binary tree.
4. Such a process is detailed in [18, 19] and involves adding dummy nodes to a \(N \times N\) port switch to create a flat topology that consists of real and virtual nodes, all of \(1 \times 2\) and \(2 \times 1\) port count.
5. Addresses are then accorded by traversing from the root to the leaves.
6. The address of a node/port on the binary tree then is essentially the route from the root to the node, assuming a conjoined string of 0s and 1s, with a 0 (or 1) selected for every right (or left) turn made during the traversal from root to the node.
7. The route through a switch is the unique segment identifier for a port in the switch.
8. An ES/AS/EOS box can be generalized to as a \(x\)-port switch. We label each of its ports using \(\lceil \log_2 x \rceil\) bits. Port addresses are thus local to an ES/AS/EOS switch, and are unique within the sector. We have detailed in [18,19] how switching becomes trivial using such an addressing scheme, i.e. input packets can be switched to their output ports in a switch, by simple operations on the port addresses.

The advantage of segment routing conjoined with binary tree-based segments is that for an \(N \times N\) switch, the switch can forward a packet by just extracting the relevant \(2\log N\) bits that would facilitate identification of the appropriate egress port for the packet.

We define three kinds of intra-sector segment identifiers (defined later): (a) SARTAG, (b) ASTAG and (c) DETAG for
passing through an EOS, AS and ES respectively. Each identifier is ingress-to-egress port-specific. A route through the DC is created by the union of multiple segment identifiers at the ingress and stripping them off at ES/AS/EOS. Hence, for a path between two ES’s within a sector, three segment identifiers are used (for the ingress and egress ES’s and the intermediate AS) and are pushed onto the packet by mapping the packet protocol identifier at the ingress protocol identifier.

Though path specific identifiers are generally avoided in large networks, we argue that these have merit in a DC environment which is: (a) controlled; (b) easy to protect from failures; and (c) paths are short (2-3 hop from ES-to-ES within a sector or just 2-hops from an ES-to-EOS).

Hence, a packet has a DETAG, ASTAG and another DETAG inserted in its header. For inter-sector communication, we also provide wavelength information in the backbone. The controller populates the wavelength tables in the EOS with information, mapping the SARTAGs to the WATAGs that denote specific wavelength mapping and enable correct dropping at the egress sector.

The protocol is based on the premise of Carrier Ethernet (CE) leading to a managed networking medium. To facilitate CE, we turn-off spanning tree protocol as well as MAC learning, while the ESs are configured to have a response to ARP broadcast. The ASs and the ESs perform switching/forwarding using a combination of user-defined tags (or labels, in the case of MPLS) switching/forwarding using a combination of user-defined tags (or labels, in the case of MPLS) and IP/MAC addresses. Such switching and forwarding is based on the IEEE 802.1ag protocol.

We define a series of tags that facilitate forwarding within the DC environment as follows:

(a) Source-Address Tag (SARTAG) denotes the unique path represented as multiple conjoined tags, from the Drop WSS interface to the appropriate egress port of the EOS.

(b) Aggregation Switch Tag (ASTAG) is used to determine the correct route through an AS. The ASTAG denotes the path through a particular AS that a packet takes. The ASTAG is added at the EOS or ES depending on whether the packet is moving away from a sector, or to within a sector.

(c) Destination Edge Tag (DETAG) denotes the unique destination edge port path in the ESs. The tag denotes the binary route within the ES that a packet would take en route to its destination.

(d) Wavelength Tag (WATAG) is used by the EOS as it sends packets from one sector to another. It denotes the wavelength number, destination sector number, fiber ring number (if any) that the packet must use from the source sector to the destination sector. The WATAG is identified uniquely by its Ethertype at the egress sector. The WATAG is also used between EOS’s for wavelength mapping, WSS configuration and multihop sequencing.

The CE-based protocol data unit is shown in Fig. 8.

B. Controller Architecture
We now describe the controller architecture and functions. The controller consists of (a) a centralized entity (GEMS), (b) a localized sector-specific entity (LEMS) and (c) a wavelength allocation system for the backbone. While FISSION’s architectural scalability is theoretically infinite, the protocol would have a limitation due to the number of flows that the controller can handle. Even then, a million node DC is easily possible using the FISSION approach.

Generalized Ethernet Management System (GEMS) consists of a cluster of management servers (arranged as per [37]), that controls the entire DC. Globally unique parameters of a packet sent through the DC are reported to the GEMS. The GEMS runs discovery algorithms that facilitate in the discovery of sectors, ESs and ASs as well as systems and VMs using IEEE 802.1ag connectivity fault management. The GEMS stores information as per .

<table>
<thead>
<tr>
<th>VM IP</th>
<th>VM MAC</th>
<th>ES MAC</th>
<th>ES Port No.</th>
<th>WATAG (Sector No.)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The Localized Ethernet Management System (LEMS; localized to a particular sector) contains the same table as the GEMS, with two more columns for DETAG and ASTAG.

The Management of Optical Backplane System (MOBS) is responsible for the optical backbone; in particular, for wavelength assignment, mapping the wavelength to the ports and which channels to add and drop at a sector.

Working: Initially, the GEMS configures all the sectors with sector numbers based on the (f, g) nomenclature described earlier. The MOBS creates an initial wavelength assignment for both add and drop circuits and informs the GEMS as well as the EOS of each sector. The add port of the EOS maintains egress sector number and its associated wavelengths. Whenever a packet arrives at the Add-port of the EOS (from ASs), it will examine the ASTAG of the frame and identify the destination sector number. The EOS will use the wavelength associated with the egress sector and send the packet to the optical backbone either in single or multihop manner.

C. Use cases
We now describe the working of the FISSION protocol using two cases that exhaustively describe the protocol.

1. Communication originating from outside the DC: In this case, a client from outside the DC requests for information. The incoming query/packet with the appropriate destination IP-address (pointing to the destination server/VM) arrives at the DCIP (see Fig. 1). The DCIP checks against its LEMS/GEMS Table for the corresponding SARTAG, ASTAG, WATAG and DETAG and encapsulates the packet into the requisite CE PDU. The packet is then sent into the backbone, from where it is picked up by the appropriate sector (based on WATAG matching) using single hop or multihop techniques. The egress sector then sends the packet to the ES through the AS (both of which forward based on egress ASTAG and DETAG respectively).

2. Communication originating from within the DC: This case represents the server-to-server or server to an external client. The ingress server sends a packet with only the destination IP address encapsulated in an Ethernet frame. The ES checks the destination IP address of the incoming packet against its LEMS table. If an entry exists then the appropriate SARTAG, DETAG and ASTAG are encapsulated in a new Ethernet frame (see Fig. 8), into which the contents of
the incoming packet are transferred. If however no entry exists, then it construes that the destination IP address is either in another sector or outside the DC. The packet is then sent to the EOS using the least-loaded link. At the EOS, the GEMS is invoked and it fetches the corresponding entry for the destination IP address. The packet is now encapsulated with the appropriate WATAG, ASTAG and DETAG. The destination sector (which could have the destination server or the DCIP) will pick the packet up. The WATAG will, in this case, determine the appropriate sector.

3) Service provisioning and carrier-class communication: To provide carrier-class communication, we assume the use of connectivity fault management protocol between every pair of servers in the FISSION DC. The CFM protocol (IEEE 802.1ag) facilitates the creation of management end-points (MEPs) between any two or more service peers in the networks. CFM is adapted to FISSION as follows: The goal is to provide for resilience within the DC. The CFM adaptation enables detection of failure among the various elements of the FISSION DC. At every service end-point i.e. a port connected to either a server or to the DCIP, we assume the existence of an MEP. Each MEP exchanges periodic heartbeat messages called connectivity check messages (CCMs). A CCM is distinguishable by its unique Ethertype and is processed as a control message by the ES/AS/EOS. The CCMs are sent in bidirectional manner between every source-destination pair. The source creates the CCMs, while the destination sinks the CCMs. Intermediate nodes only forward the CCMs at the highest QoS level policy (control messages). Loss of three successive messages indicates to the destination that a failure in route has happened. The control plane (GEMS), allocates for every work path a protection path. Upon loss of three successive CCMs, the destination selects the alternate path.

4) Single, multihop communication and wavelength assignment: The MOBS is also responsible for wavelength assignment. All sectors are allocated H wavelengths for addition into the backplane. For the case of single hop traffic, for static WSS configuration, the MOBS informs the Add EOS on which wavelength it should transmit, corresponding to whether a receiver is available at the egress sector. For the case of tunable receivers, the MOBS facilitates reconfiguration of the WSS to facilitate adequate matching between the ingress and egress sectors. For the case of multihop traffic, the MOBS finds one or more intermediate sectors that facilitate traffic flow. In this case, the MOBS may also facilitate reconfiguration of WSS at the egress sector.

D. Protection and Restoration

While the IEEE802.1ag enables protection for any path within the DC at layer 2, the issue that we must consider is protection against a fiber cut in the backplane. Each fiber ring is assumed as a dual counter propagating ringlet and signal is introduced into both fibers and the rings are kept open by an ON/OFF switch (see Fig. 3) in the OFF state. At the receive side of an EOS (drop), both rings drop signal to the EOS, but due to the switch being OFF, only one copy reaches the EOS. Whenever a fiber-cut occurs, the second copy is allowed to go through by switching ON the wavelength-agnostic switch. We note that reliability of FISSION is comparable to other architectures in the following way:

(a) Intra-sector: There are p independent paths due to the folded Clos network (within the sector) and hence reliability is of the order p − 1 (i.e. up to p − 1 links can go down without losing connectivity).

(b) Inter-sector: At the ingress and egress there are p − 1 independent paths. However, in the core there is a fully dedicated protection fiber ring for every work-fiber ring. Hence, while this lack of more than one path may appear to be a drawback, it is in fact the norm for most telecom networks using carrier-class best practices of 50ms restoration.

(c) Multihop: In the case of multihop provisioning, there are up to n(F − 1) paths available. Of these F − 1 paths are unique (i.e. non-overlapping) using fiber diversity and hence reliability is of the order of F − 1.

VI. FISSION DATACENTER OPTIMIZATION

In this section, our goal is to design a DC that maximizes the total traffic that can be provisioned subject to physical constraints. We also seek to compute the inflection point at which a FISSION-type optical backplane architecture is a better solution as compared to traditional multi-rooted switch fabric. The goal of this optimization exercise is to maximize the traffic by placing it into different types of datacenters, of which FISSION is one of the types. A second goal is compute the dual of the model and check if it is infeasible, which would imply that the primal (whose objective is to maximize traffic provisioned) is unbounded, thus validating our unprecedented scalability claim.

Table 4: List of Input Parameters.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( T_{ab} )</td>
<td>Traffic from node ( N_a ) to node ( N_b )</td>
</tr>
<tr>
<td>( PM^k_{ab} )</td>
<td>Path ( k ) from node ( N_a ) to node ( N_b ): ( PM = {PM^k_{ab}, \forall a, b, v, k} ); (</td>
</tr>
<tr>
<td>( g(PM^k_{ab}) )</td>
<td>Used capacity of path ( PM^k_{ab} ), i.e. residual capacity of ( PM^k_{ab} ), (</td>
</tr>
<tr>
<td>( \Delta )</td>
<td>Delay threshold for every provisioned traffic request in the FISSION DC</td>
</tr>
<tr>
<td>( \rho )</td>
<td>A delay converting parameter for a particular load that can be obtained using Little’s Theorem</td>
</tr>
</tbody>
</table>

Table 5: List of Decision Variables.

<table>
<thead>
<tr>
<th>Notation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \theta_{ab} )</td>
<td>( = 1 ), if ( T_{ab} ) is provisioned in the FISSION DC; 0 otherwise</td>
</tr>
<tr>
<td>( PM^k_{T_{ab}} )</td>
<td>( = 1 ), if traffic ( T_{ab} ) is assigned to path ( PM^k_{ab} ); 0 otherwise</td>
</tr>
<tr>
<td>( \mu_{ab} )</td>
<td>( = 1 ), if ( T_{ab} ) is provisioned over ( \lambda^k_{ab} ); 0 otherwise</td>
</tr>
<tr>
<td>( \beta_1 )</td>
<td>( = 1 ), if only ES is required: 0 otherwise</td>
</tr>
<tr>
<td>( \beta_2 )</td>
<td>( = 1 ), if only ES and AS are required (single sector); 0 otherwise</td>
</tr>
<tr>
<td>( \beta_3 )</td>
<td>( = 1 ), if multisector DC is required using single fiber pair; 0 otherwise</td>
</tr>
<tr>
<td>( \beta_4 )</td>
<td>( = 1 ), if multi-sector multi-fiber DC is required; 0 otherwise</td>
</tr>
<tr>
<td>( \lambda^E_{abh} )</td>
<td>( = 1 ), if edge switch ES, ( E \in (PM^k_{abh}) ); 0 otherwise</td>
</tr>
<tr>
<td>( \lambda^F_{abh} )</td>
<td>( = 1 ), if aggregate switch AS, ( A \in (PM^k_{abh}) ); 0 otherwise</td>
</tr>
</tbody>
</table>

Our objective is to design a DC such that the provisioned traffic is maximized, i.e. \( \max_{\text{s.t. }} \sum \theta_{ab} T_{ab} \theta_{ab} \), while maintaining QoS (bounded delay, full bisection bandwidth). In the process, we investigate bounds on the FISSION architecture.

Claim: The maximum amount of traffic that can be provisioned in the FISSION DC is theoretically infinite.

Proof: Our claim that the FISSION architecture is one, whereby the total provisioned traffic can scale significantly...
will imply that the primal formed in the constrained optimization process is unbounded. We prove that the primal is unbounded, by showing that the dual is infeasible.

We begin the optimization model by considering the different aspects of the FISSION architecture based on the number of servers supported. As a basic building block, we assume the parameters of Section III to be valid and further assume that an ES is of the order of \( N \times N \). This leads to four DC architectures:

1. If the number of servers is less than \( N \), only a single ES is required, i.e. if \( \sum_{v,a,b} T_{ab} \leq NC \), then, \( n = 1 \), \[ |ES| = 1 \]
2. If the number of servers is greater than \( N \), but less than the number supported in a sector, in which case ESs and ASs are required, they all fit into a single sector, i.e. \( \frac{1}{\sum_{v,a,b} T_{ab}} \leq \frac{WM}{N} \), then, \( n = 1 \), \[ |ES| \leq \frac{WM}{N} \]
3. If the number of servers is such that they need more than one sector, but all the sectors can fit in a single bidirectional fiber ringlet, i.e. \( \frac{1}{\sum_{v,a,b} T_{ab}} \leq \frac{WM}{N} \), then \( n = \frac{\sum_{v,a,b} T_{ab}}{WM} \), \[ |ES| = n \cdot \frac{WM}{N} \]
4. If the number of servers is such that they require more than one fiber pair, and an entire multi-fiber backbone is used, i.e. \( \sum_{v,a,b} P M_{ab}^{k} \), \[ |ES| = n \cdot \frac{WM}{N} \]

Accordingly, four binary decision variables \( (\beta_l) \) that determine the choice of our architecture are also defined. This leads to the architecture constraint:

\[ \sum_{l=1}^{4} \beta_l = 1 \]

The path assignment constraint ensures that a path is assigned to every provisioned traffic:

\[ \sum_{l=1}^{4} \sum_{v,a,b} PM_{ab}^{k}, \beta_l = 1 \]

The granularity constraint holds for each particular traffic request:

\[ \theta_{ab}PM_{ab}^{k} (|PM_{ab}^{k} - g(PM_{ab}^{k})|) \geq \theta_{ab} T_{ab} \]

The above equation becomes linear by pivoting against \( \theta_{ab} \).

We now define the full-bisection bandwidth constraint between two nodes \( N_a \) and \( N_b \) in the DC that supports traffic \( T_{ab} \) using path \( PM_{ab}^{k} \). Since there are four possible architectures \( (\beta_l) \), we will consider all the choices:

1. If \( \beta_1 = 1 \), then \( \forall T_{ab}, N_a \in ES_i, N_b \in ES_j, i = j \) and \( PM_{ab}^{k} = 1 \), \[ \text{i.e. there is a single ES and that the path from } N_a \text{ to } N_b \text{ exists. The following condition guarantees that this path is available:} \]

\[ PM_{ab}^{k} \geq \frac{T_{ab}}{M} \]

2. If \( \beta_2 = 1 \), this leads to two scenarios: the ingress and egress traffic requests may or may not be connected to the same ES i.e. for \( N_a \in ES_i, N_b \in ES_j, PM_{ab}^{k} = 1 \) either, \( i = j \) or \( i \neq j \). Therefore, \( \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} = 1 \), \[ \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} = 0 \] and \( \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} = 1 \), which upon linearizing leads to \( \sum_{v,i,j} A_{ab}^{ijk} + \theta_{ab} = 1 \), \[ \sum_{v,i,j} A_{ab}^{ijk} + \theta_{ab} = 2 \] with the graph constraints for ingress and egress ES as \( ES_i, ES_j \in \{ PM_{ab}^{k} \} \) and \( PM_{ab}^{k} \geq \frac{T_{ab}}{M} \). Therefore, \[ \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} = 1, \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} = 1, \forall T_{ab} > 0 \].

3. If \( \beta_3 = 1 \), we have \( N_a \in ES_i, N_b \in ES_j, i \neq j, \) and \( PM_{ab}^{k} = 1 \), \[ ES_i, ES_j \in \{ PM_{ab}^{k} \} \), and \( PM_{ab}^{k} \geq \frac{T_{ab}}{M} \). The above equations become linear by pivoting against \( \theta_{ab} \).

4. If \( \beta_4 = 1 \), we have \( N_a \in ES_i, N_b \in ES_j, i \neq j, \) and \( PM_{ab}^{k} = 1 \), \[ ES_i, ES_j \in \{ PM_{ab}^{k} \} \), and \( PM_{ab}^{k} \geq \frac{T_{ab}}{M} \). Further, the following two path identities are valid:

\[ \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} = 2 \]
\[ \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} = 2 \] The above equations can be linearized as:

\[ \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} = 2, \forall \theta_{ab} > 0 \]
\[ \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} = 2, \forall \theta_{ab} > 0 \] These equations represent that the ingress and egress sectors are not the same.

The sector arguments hold as follows, implying that the ingress and egress sectors are not the same.

\[ \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} \neq \sum_{v,i,j} A_{ab}^{ijk} \theta_{ab} \]

Wavelength assignment constraint: full bisection bandwidth: There are \( W = h \) add wavelengths per sector where \( h \) is the number of subcarriers per superchannel (\( h = 1 \) if no superchannels are used, and then \( W = h \)). The wavelength assignment per-ring in the backbone is as follows:

\[ \{ A_1, A_2, ..., A_l \} \]
\[ \forall f = 1, 2, ..., F \] (1)

where \( A_i = \{ \lambda_{(i-j)} W, A_{i-1}, ..., A_{i-1} \} \) denotes the set of wavelengths assigned to the \( i \)-th sector in each ring and \( \{ \} \) represents the wavelength assignment of the \( f \)-th fiber ring. The assignment gets repeated across every fiber ring in the backbone.

The constraint for wavelength assignment with full bisection bandwidth is:

\[ \sum_{v,a,b} T_{ab} \leq C_t, \forall \theta_{ab} > 0 \]

where \( C_t \) is the capacity of the \( t \)-th wavelength.

Delay constraint: For every provisioned traffic request \( T_{ab} \), the delay constraint is:

\[ \beta_1 (ES_{delay}) + \beta_2 (ES_{delay} + AS_{delay}) \]
\[ + \beta_3 (2. ES_{delay} + 2. AS_{delay}) + 2. ES_{delay} \]
\[ + \beta_4 (2. ES_{delay} + 2. AS_{delay} + 2. ES_{delay}) \]
\[ < \Delta \]

The delay at an ES/AS/EOS is dependent on: (a) the granularity of the connection, and, (b) the combined granularities of all the connections provisioned at that ES/AS/EOS. Since we assume the same granularity for all connections, i.e. the server line rates are equal, the delay through an ES/AS/EOS is proportional to the load on the device. To compute the individual delays, we have the delay through ES \( \gamma_1 \) as \( \frac{P}{\sum_{v,a,b} A_{ab}^{ijk} \theta_{ab} T_{ab}} \). Similarly, the delay through ES \( \gamma_1 \) is given by \( \frac{P}{\sum_{v,a,b} A_{ab}^{ijk} \theta_{ab} T_{ab}} \).

To compute the delay through an EOS, we break the delay through the EOS into two parts: (a) delay through the electronics, and, (b) delay through the optics.
Finally, we have the following four delay constraints.

\[ \beta_1 \left( \rho T_{ab} + \frac{1}{N} \sum_{v,a,b,k} A_{ab}^v T_{ab} \right) \leq \Delta, \forall i \text{ and } \theta_{ab} > 0 \]

\[ \beta_2 \left[ \frac{1}{N} \sum_{v,a,b,k} (2 A_{ab}^v \theta_{ab} T_{ab}) \right] \leq \Delta \]

\[ \beta_3 \left[ \frac{1}{N} \sum_{v,a,b,k} (\sum_{i=1}^p A_{iab} \theta_{ab} T_{ab}) \right] \leq \Delta \]

\[ \beta_4 \left[ \frac{1}{N} \sum_{v,a,b,k} (2 A_{ab}^v \theta_{ab} T_{ab}) \right] \leq \Delta \]

and \( \theta_{ab} > 0 \)

Nonlinearity: Many of the above constraints are non-linear, but can be linearized using simple mathematical adjustments, such as by converting the L.H.S. from a product of variables to a sum and converting the R.H.S. to an appropriate number that reflects (in most cases, the upper bound) of the sum (and hence the \( \leq \) inequality persists), as was done in the full-bisection bandwidth constraint.

Duality: To compute the dual, we restate the primal objective function by replacing the 0 coefficients for primal variables with abstract entities. The primal objective then becomes:

\[
\max \left( \sum_{v,a,b} T_{ab} \theta_{ab} + \frac{1}{N} \sum_{v,a,b} y_{ab} \delta_{i} + \gamma_1 P_{M} \theta_{ab} + \gamma_2 A_{ab} \theta_{ab} + \gamma_3 A_{ab} \theta_{ab} \right)
\]

The dual of the above primal is then:

\[
\min \left( -z_1 + y_2 + \frac{T_{ab}}{T_{ab}} z_3 \right)
\]

where, the dual variables with their respective bounds are: \( z_1 \) unrestricted, \( z_2 \) unrestricted, \( z_3 \geq 0 \), \( z_4 \) unrestricted, \( z_5 \leq 0 \) and \( z_6 \geq 0 \).

subject to:

\[
\sum_{v,a,b} T_{ab} \leq 0
\]

\[
y_0 \leq 0
\]

\[
z_3 \frac{T_{ab}}{M} \left( |P_{M_{ab}}| - g(P_{M_{ab}}) \right) \geq \gamma_1
\]

\[
z_4 + z_6 \rho \frac{T_{ab}}{M} \geq \gamma_2
\]

\[
z_4 + z_6 \rho \frac{P_{M_{ab}}}{M} \geq \gamma_3
\]

\[
z_5 \sum_{v,a,b} T_{ab} \geq \gamma_4
\]

The first one among the above constraints is clearly infeasible, since sum of traffic is a non-negative entity. Similarly, the last one among the above constraints is also not possible to satisfy as \( z_5 \) is less than zero and \( y_4 \) is zero. Thus, the dual formed is infeasible, which implies that the primal is unbounded, and hence based on the formulation, our DC can accommodate theoretically unbounded traffic.

VII. SIMULATION AND RESULTS

In this section, we show numerical evaluation of the FISSION framework. The results are classified into three parts, namely, (a) simulations pertaining to architecture and protocol; (b) simulations at the optical layer; and, (c) comparative simulations of FISSION and other DC architectures. For a given set of input parameters (number of servers, etc.), we compute the optimal DC architecture using a MATLAB simulation. The resultant DC architecture is then simulated with traffic using a Java-based discrete event simulation (DES) model. The resultant delay, blocking rate, etc. are compared for varying loads and DC sizes. The results consider both single-hop as well as multi-hop traffic patterns and are applied for both elephant and mice flows.

A. DC Architecture and protocol simulations

We built two models to evaluate the architecture and protocols, respectively.

DC Architecture model: The model was developed in Java as a discrete event simulation run over a Quad Core Intel i7 machine. It takes as input the number of servers, the configuration of the ES, AS and EOS, the number of wavelengths in each fiber backbone. Servers are connected to ESs at 10Gbps line rate per server, and packets are generated by servers as well as at two DCIPs.

Traffic is classified into four service priorities with varying delay requirements primarily for the EOS to function as a multi-priority VOQ-based SDN architecture. Packets are generated according to a Poisson arrival process with exponential lengths and an MTU of 9600 bytes (jumbo frames) with average packet size 450 bytes. Packets are assumed to be part of a flow and we associate a Hurst parameter (between 0 and 1) to a flow – whereby Hurst parameter closer to 1 implies extremely bursty traffic i.e. continuous large burst of packets and hence signifies an elephant flow. Unless specified, to compute load, we assume that the traffic mix is 70:30, i.e. 70% of the traffic is contributed by elephant flows, while 30% is contributed by mice or trickle flows, while in terms of number of connections, the ratio is 20:80, i.e. elephant flows are only up to 20% of the number of connections, while mice flows constitute 80%. An elephant flow is of minimum duration of 10 seconds, mean of 30 seconds and exponentially distributed. The number of flows simulated in a DC during the DES simulation is varied from 10E8 to 10E10 in 10 equal increments for a 100K node DC. The number of flows simulated is varied from 10E9 to 10E11 in 10 equal increments for a million node DC.

The traffic pattern is modeled as per true DC traffic from a service provider and is a mix of 70% DCIP to server (and back) unicast flows, 20% server to server flows, and 10% multicast traffic. Destinations are randomly chosen with an equal probability among servers for the first packet of a flow, and subsequent packets are allocated based on the flow duration.

Load is computed as the ratio of the number of bits that are sent into the DC versus the total amount of bits that the DC can handle (equal to the number of servers \( \times \) line-rate of the servers) per unit-time. Load is hence in the range [0, 1].

DC Optimization and Protocol Model: The optimization model is built in MATLAB to compute the DC design and thereafter is validated with the Java-based simulation results with the optimization model. We give as input the traffic matrix \( T_{ab} \) and \( \Delta \) and the model computes the optimal DC. The model then details the architecture as matrices shown in Section VI and these are used by the DES to run traffic.
shown in Fig. 9 is the delay profile for different-sized ES for a 50K DC. The delay profile is a function of multiple ES sizes of ports 16, 64 and 128 respectively. Delay is averaged over all types of flow for a particular load. Delay is computed from server-to-server or from server-to-the-DCIP. The key takeaway from this figure is that the delay profile is identical for different ES configurations, and impacted by specific ES behavior—matching the delay profile for the ES. The figure is important from the perspective of choosing the appropriate ES for the DC. It must also be noted that the ES+AS+EOS delay is almost 95% of the overall delay.

It is also pertinent to note that delay is almost stable beyond a load of 0.4. The reason the delay does not increase beyond a load of 0.4 is because of two factors: (a) till a load of 0.4, the main delay contribution is due to EOS scheduling delay. Beyond that load, the carrier-class properties of the protocol along with the presence of 100% bisection bandwidth in the core sets in which makes the delay deterministic. The delay does slightly increase (due to ES/AS queuing), but this increase is minimal (about 1-5 microseconds). (b) Beyond a load of 0.4, the traffic becomes dominated by elephant flow data which once set up does not contribute much to the delay.

To generate Fig. 10, we simulated the architecture for 10K, 25K, 50K, 100K and 1 million servers. Traffic arrivals are as described earlier. The delay performance of the single hop scheme is compared to the multihop scheme with \( \psi = 3 \) receiver banks per sector. The choice of \( \psi = 3 \) is motivated by another set of experiments that were performed (but not reproduced here due to lack of space) that studied the relationship between \( \psi \) and the blocking of flows in different-sized DCs. Load is assumed to be non-atomic. Note that the performance of all the single hop DCs get grouped into a similar set of curves, while the multihop scheme based DCs are grouped into a second set of curves which are initially spaced apart and converge with increase load. It may be difficult to visually spot the difference due to the dense nature of the figure as to how the single hop curves behave as they seem to almost overlap, but there is divergence as the load increases, especially for higher server count DCs.

Shown in Fig. 11 is the impact of elephant flows and mice on the FISSION framework for a multihop model. We simulated a 10K, 50K and 100K node FISSION DC for generating this result. We assume AS/ES delays to be a function of load (20 microseconds at full load). The EOS delay is 15 microseconds. We define OMEGA as the percentage of traffic due to elephant flows compared to mice flows. An OMEGA value of 0 means only mice flows exist and the network provisioning is extremely dynamic. An OMEGA value of 1 means that only elephant flows exist. An elephant flow is defined as a file transfer that is of minimum 1 GB (Gigabyte) size. The interesting (and counter intuitive) takeaway from the figure is that delays converge for heavy elephant flows. This is perhaps due to the optical multicast property that is more suited towards circuit communication than for packet switched communication. This enables us to conclude that the FISSION architecture being backplane-centric is more suited for elephant flows. In fact, it does not matter what the source-destinations are when it comes to elephant flows. In general delay for multihop model is about 25% more than that single hop model with \( \psi = 3 \).

![Fig. 9. Effect of load on delay varying ES sizes](image)

![Fig. 10. Delay versus load for multihop FISSION DC for various number of servers with different backplane capacities](image)

![Fig. 11. Impact of Elephant and Mice Flows](image)
A. Traffic Assignment

We randomly generate traffic value and delay sensitivity. traffic subjected to the optimization process by increasing it from 25 Gbps to 25000 Gbps in 10 non-equal increments and assuming that each server is at 10 Gbps. We have here simulated the backplane capacity. We reserve the first 5 increments for the optimization model. Though there is failure this failure is only about 3.5% for a 1ms delay sensitive traffic assignment in a 100K node DC, and 8% for a 0.5ms delay sensitivity.

B. Optical Layer Simulations

To determine the optical layer performance of the system, a setup was created in VPI TransmissionMaker\textsuperscript{TM}. Since the addition of sectors beyond the initial ring has only an attenuation impact on the FISSION framework, we simulated a 6-sector system and then varied the power levels at the receiver to measure the impact of adding new sectors. Key to the simulation exercise was the use of superchannels. We considered 40Gbps, 60Gbps, 80Gbps and 100Gbps superchannels in the backplane, though all servers were assumed at 10 Gbps. For sake of completeness, we considered QPSK, 4-QAM and 16-QAM modulation formats. Each sector consists of a 3\,dB splitter, various drop couplers and 8\,OFDM Transponders, to drop and add the channels. One output of the splitter drops all the channels in the ring to a 1 \times 8 WSS from where a subset of channels can be selected. We have built a WSS in the VPI TransmissionMaker\textsuperscript{TM} that supports flexible-grids whose center frequency can be tuned to optimize channel spacing and line-rate. The other output of the splitter is used for forwarding data to the next sector. An add coupler facilitates addition of 8 OFDM channels into the ring. The setup is shown in Fig. 14.
Optical OFDM signal is generated as a pseudorandom binary sequence. A sector adds up to 8-channels and the output is measured in terms of system performance at farthest sector (which is 5-sectors away from the ingress sector). Effect of Bit-Rate on SER (Symbol Error Rate) is monitored for different modulation techniques i.e. QPSK, 4-QAM and 16-QAM and is as shown in Fig. 15. As expected, QPSK is found to perform best among other modulation techniques. It is also seen that above 100Gbps data rate SER degrades below the value of 1E-09. At lower bit rates, the difference between the modulation formats is more pronounced than at higher bit rates. Hence, at higher bit rates, it is preferable to use advanced modulation formats with FEC, which provide better bandwidth utilization, subject to acceptable SER limits.

C. Comparative Simulations

Shown in Fig. 17 is a comparison of FISSION protocol performance with that of DCell, Portland and VL2 for 10K nodes, averaged over unicast and multicast traffic with the FISSION architecture modeled in the multihop arrangement. As seen, all other schemes have significantly poor performance. Note that load is computed at different epoch as it is difficult to normalize it across the various schemes. The FISSION architecture delay is consistently better by several orders of magnitude.

Shown in Fig. 18 is a comparison of FISSION architecture with other protocols for packet loss. Packet loss for FISSION due to its 100% bisection bandwidth, is almost negligible. In contrast, packet loss for other protocols such as DCell and VL2 is high. In fact, WaveCube and c-Through are expected to have similar high packet loss due to: (a) absence of full bisection bandwidth support and (b) repeated optical switch reconfiguration and are omitted from further discussion.

VIII. CONCLUSION

We have proposed the FISSION architecture towards a scalable DC fabric. The architecture uses optical buses in the
backplane coalesced with a novel interconnection fabric that facilitates a switchless DC that can scale to unprecedented levels. We have shown how to design a DC up to even a million nodes using contemporary off-the-shelf components. An optimization program is built that shows how such a DC can be built and dual of which being infeasible proves that there are no theoretical bounds to the scalability of the FISSION concept. A protocol for communication within the DC is proposed. Performance results include architectural understanding, optical layer simulations and comparison with other architectures validate our design.

REFERENCES

[20] Online: www.ecil.co.in/ECR/ECR.pdf