New Techniques to Route in Folded-Clos Topology Data Center Networks

Peter Willis, Nirmala Shenoy†
ISchool, School of Information
Rochester Institute of technology
Rochester, New York, USA
pjw7904, nxsvks@rit.edu

Yin Pan, Bill Stackpole
Department of Cybersecurity
Rochester Institute of technology
Rochester, New York, USA
yin.pan, bill.stackpole@rit.edu

John Hamilton
School of Mathematics and Statistics
Rochester Institute of technology
Rochester, New York, USA
john0903hamilton@gmail.com

***Abstract*— New architectures and topologies continue to be investigated to address the growing demands and challenges faced by Data Center Networks (DCN) - the communications backbone of the datacenter. We believe that novel techniques that leverage the high redundancy and symmetricity in DCN topologies can significantly simplify DCN protocols and operations and improve DCN performance. We adopted the folded-Clos topology to investigate a novel Multi-Root Meshed Tree Protocol (MR-MTP) and compared its performance to the popular protocol suite adopted in folded-Clos topologies, namely the Border Gateway Protocol (BGP) with Equal Cost Multipath Protocol (ECMP) with and without Bidirectional Forwarding Detection (BFD). We studied the convergence time, packet loss, control overhead and blast radius after an interface failure, introduced in multiple points. Our studies conducted on the FABRIC testbed provide strong validation that novel techniques that leverage the DCN structures can indeed simplify DCN protocol operations and improve performance.**

***Index Terms*—Multi Root Meshed Trees, Path Establishment with Virtual IDs, Auto-configuration, Auto-address assignment.**

I. Introduction

A datacenter is composed of servers, storage devices and networking nodes. The networking nodes form the datacenter network (DCN) which is the communications backbone to route data between servers. High availability, scalability and fault tolerance are critical to DCN performance. With the growing demand on datacenters, a DCN requires careful architectural and design considerations. While several new DCN architectures and topologies are being investigated [3, 4, 6,7,9, 10, 12, 13, 17-21, 24, 25], the increased network operational demands are primarily met by aggregating multiple off-the-shelf network protocols. Given the structured network topologies of today and the advances in related network technologies, novel protocols targeting desired functions can be designed to simplify network operations and improve network performance. Such protocols, however, must be backward compatible to the Internet protocol (IP) and Ethernet, to make it easy for industry to deploy them. Based on this hypothesis, the Multi-Root Meshed Tree Protocol (MR-MTP) [22, 23] was designed with simple techniques to establish multiple loop-free routes in a folded-Clos DCN and forward data (IP packets) between the servers. MR-MTP uses auto-assigned Virtual IDs, to establish paths thereby reducing configuration needs. We conducted a detailed investigation of a C language coded MR-MTP performance versus the performance of Border Gateway Protocol (BGP) with Equal Cost Multipath protocol (ECMP) using the FABRIC testbed [5]. BGP with ECMP is a popular protocol suite used with folded-Clos topology and hence was selected for the performance studies. The C code of MR-MTP [27] and BGP designed for DCN [12] software from FRRouting [1] was downloaded from their respective github repositories to reserved FABRIC test bed [5] slices to conduct the studies. BFD [11] was enabled in BGP to investigate its impact on the failure recovery and convergence. Python scripts [29] were designed to provide easy setup of the folded Clos topologies with desired configuration on the FABRIC testbed. These scripts help 1) set up folded clos topologies with the desired number of tiers, servers etc., 2) to deploy the software (such BGP, BFD, MR-MTP) at the DCN routers, 3) introduce failures at specific points in the folded-Clos topology and record some stats, 4) collect multiple logs from the routers and interfaces during convergence, 5) parse the collected logs to extract the performance data of interest, 6) scripts to compute the performance metrics from data collected and lastly 7) scripts to verify the topology and router configuration for MR-MTP and BGP/ECMP/BFD.

We present the performance study on a 2 Point of Delivery (PoD) and 4 PoD topology, with 3 tiers of DCN routers using the FABRIC testbeds as proof of concept and to validate MR-MTP for use in DCNs. Future work will extend the DCNs to more PoDs and tiers to conduct scalability studies using mininet [30]. To study network availability, we collected convergence time after a failure. To study stability of the DCN we collected the blast radius. Packet loss during an interface failure (at different points in the test topology) provides information on the disruption to service on a failure at these points. To understand the recovery process, we collected the update messages generated during convergence. Frames captured (using wireshark) during the test runs are presented to illustrate the differences in operational overhead between MR-MTP and BGP/ECMP/BFD. Routing tables from selected devices in the test topologies are provided to illustrate the route establishment of MR-MTP versus BGP.

A. Protocol Evaluation

FABRIC allows reservation of interconnected computing resources (at remote sites) on which we installed and executed MR-MTP and the protocol suite BGP/ECMP/BFD. Identical folded-Clos topologies, with devices running Rocky Linux 8

979-8-3503-5554-3/24/$31.00 ©2024 IEEE
DOI 10.1109/SCW63240.2024.00116

virtual machines (VM) were set up at these remote sites. To facilitate reproducibility, a suite of Python scripts [29] were written to configure the topologies and access remote nodes using FABRIC APIs. Included in the test suite is a custom-built traffic generator [28]. The traffic generator allows any number of packets to be transmitted back-to-back (up to a full Ethernet frame payload) between a sender and receiver device (i.e. the servers in the DCN). Part of the traffic generator software running at the receiver analyzes the received traffic to provide a count of duplicated, lost and out of sequence packets. The traffic generator was used to study the impact of interface failures on packet delivery.

The results provide strong validation that simple techniques and protocols can significantly improve DCN performance. Fig. 1 shows the protocol stack in routers running BGP/ECMP/BFD suite compared to MR-MTP. MR-MTP is layer 3 protocol.

Section II provides some background and related work on DCN architectures and protocols. Section III describes meshed trees and the MR-MTP. Section IV, discusses network convergence and availability, BGP/BFD's handling of interface failures, and the techniques used by MR-MTP to speed up failure detection and recovery. Section V introduces the performance metrics of interest in this study and their significance. Section VI introduces the test topologies and test cases. Section VII presents performance results. Section VIII provides conclusions and Section IX discusses future work.

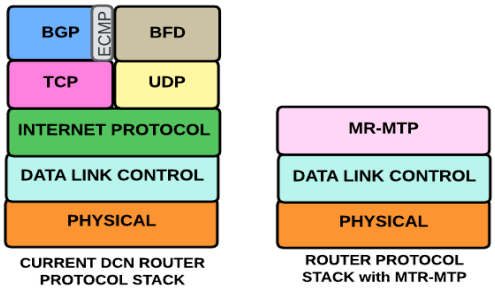

Figure 1: Protocol stack at BGP/ECMP/BFD vs MR-MTP router

II. Background and Related Work

Several studies have been conducted on DCN architectures and topologies to improve its performance, energy and cost. Popular DCN topologies include the folded-Clos, Virtual Layer 2 (VL2) [4,17], Dcell [10] and Bcube [9]. Variations of BGP [12, 18, 21, 25] and Intermediate System to Intermediate System (IS-IS) [24] have been proposed in several of these solutions to simplify DCN operations and improve their performance. DCell includes servers as computing nodes with DCell Fault Tolerant Routing to exploit its recursive structure. BCube uses source routing based off IS-IS. Hyperconverged infrastructure brings server and storage resources into a single appliance managed by a virtualization layer to allow dynamic reallocation of resources as computing demands shift [31]. Software defined networks (SDN) play an integral role in architectures in [21, 25] which earlier used variations of BGP and later opted for an all-optical switching solution to replace spine blocks [37]. Variations of Open Shortest Path First (OSPF), Routing In Fat Trees (RIFT) [20] Link State Vector Routing (LSVR) [18], and several variations of BGP [18, 21, 25] have been proposed and investigated. BGP solutions use BFD [11], a hello protocol that can be tailored to application needs to speed up failure detection, at the expense of two additional protocols – BFD and User Datagram Protocol (BFD messages are carried in UDP datagrams). The concepts underlying protocols used in many of these solutions are variations of protocols designed for other purposes a couple of decades ago. For example, RFC 7938 [12] proposes a reconfigured eBGP for intra-Autonomous System (AS) communications for use in folded-Clos DCN. Investigation of routing schemes for DCNs is ongoing research. We focus our studies on folded-Clos topologies running BGP and, as these protocols have seen wide deployment. We used RFC 7938-compliant eBGP for DCN folded-Clos topologies from the FRRouting [1] site in our experiments. This version of BGP allows enabling/disabling BFD and ECMP.

MR-MTP is Layer 3 and Layer 2 agnostic. It is also backward compatible to IP and Ethernet. It performs essential DCN operations, i.e., 1) establishing multiple routing path (for which BGP is a candidate protocol), 2) forwarding IP packets between servers (currently performed by IP), 3) load balancing (performed by ECMP enabled in BGP) and 4) fast failure recovery (for which we currently use BFD). Thus MR-MTP a simple layer 3 protocol replaces BGP, ECMP, IP and BFD. MR-MTP does not use TCP required by BGP and UDP required by BFD. Thus MR-MTP replaces 6 protocols in a DCN router for folded-Clos topology as shown in Fig.1.

III. Meshed Trees and MR-MTP

We use Fig. 2 to explain meshed trees and how this technique is applied to a folded-Clos topology DCN. A folded-Clos topology is a tiered structure where the server/compute nodes are considered to be tier 0 devices and all networking devices exist at tiers above. Top of Rack (ToR) devices, which connects to the server rack and servers are in tier 1. We consider each ToR to be the root of a tree. The different trees from each ToR mesh at the upper tier spine devices (creating a meshed tree structure- see Fig.2, where a purple tree from TOR VID=11and a blue tree from ToR VID=14 mesh at the top tier spines S2_1 to S2_4) to provide multiple loop-free paths to quickly provision a fallback path in the event an existing path fails. We explain meshed trees using Fig. 2.

In Fig. 2, we assigned each ToR a Virtual ID (VID) (their derivation is explained in section A below). ToRs have been assigned VIDs 11, 12, 13 and 14. Extension of the ToR VIDs will be assigned to the upper tier spines to construct the meshed trees. In Fig. 2, we also show VID tables at some spines, namely S1_1, S1_4, S2_1, S2_4, which will help in explaining the use of the VID tables. The VID derivation for upper tier spines is explained in section B. The VIDs are color coded to help trace the path from the respective ToRs to the top tier spines. Along with the VIDs, an upper tier device also stores the ports on which the VIDs were acquired in the VID table. The ports of acquisition will be used when MR-MTP

forwards an encapsulated IP packet between servers and is explained in section D.

A. VIDs for ToRs

Servers in datacenters are arranged in racks, where the ToR connects the servers in the rack. The ToR shares a subnet with servers [4, 8]. For VMs running in different servers to collaboratively execute a job, we assume that Virtual Extensible Local Area Network (VXLAN) [14] is used for inter-rack VM communication. The VM traffic is encapsulated in an outer IP header, which carries the server's IP address, in which the VMs reside. The current algorithm for ToR VID derivation uses the third byte in the subnet IP address that the ToR shares with servers in its rack. In Fig. 2, notice that the first ToR has a VID 11 derived from the third byte of the subnet IP address 192.168.11.0/24, the second ToR has a VID 12 and so on. This approach simplifies MR-MTP data forwarding in section D. More than 1 byte (or other algorithms) can be used to generate the ToR VID.

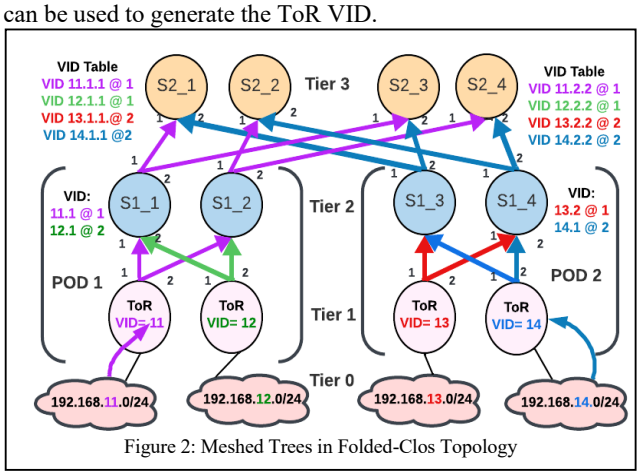

Figure 2: Meshed Trees in Folded-Clos Topology

B. Establishing Meshed Trees

The port numbers of the DCN devices play an important role in establishing meshed trees using VIDs. We explain the construction of the purple tree from ToR VID=11.

- The ToR advertises its VID on its upstream ports.
- S1_1 and S1_2 send in a request to join the tree and are offered VIDs 11.1 and 11.2 by the ToR VID=11.
- The ToR derives the VIDs for the requesting spines, by appending the port number on which a request arrived to its VID (11). S1_1 and S1_2 thus acquire VIDs 11.1 from ToR VID=11 and 12.1 from ToR VID=12.
- S1_1 and S1_2 in turn advertise their VIDs.
- S2_1, S2_2, S2_3 and S2_4 send in join requests and are accordingly assigned VIDs 11.1.1, 11.2.1, 11.1.2 and 11.2.2 respectively (by S1_1 and S1_2) following the same process of appending the port number (on which the request arrived) to their VIDs.

The messages (arrows) sent by the downstream devices are color coded to show the growth of the purple tree from the ToR VID=11. In a similar manner, a green tree from ToR VID=12, a red tree from ToR VID=13 and a blue tree from ToR VID=14 will be established. Continuing this process, all the top tier spines S2_1, S2_2, S2_3 and S2_4 will have one VID from each of the four ToRs. i.e., they will have a VID starting with 11, 12, 13 and 14 (see the VID tables next to S2_1 and S2_4). The VID table records all the paths between the ToRs and the top tier spines. A close inspection of VIDs at the spines will show that they carry the path or route information. Inherently VIDs also help with loop-avoidance. *The approach mitigates the need for address assignment to spine devices or networks. No routing protocol was used to establish the multiple paths.*

MR-MTP's auto-assigned VIDs address a major challenge faced in networks - configuration errors [8]. It provides a simple mechanism for auto-naming and self-configuring [8, 24]. Because of these features, the scheme can easily scale to any number of spine tiers.

C. MR-MTP Operations

MR-MTP guarantees reliability through request-response and accept-acknowledge messages between peers connected on a link. In its current implementation, MR- MTP messages are carried in Ethernet frames. Included in MR-MTP operations is a hash algorithm to load balance traffic from a downstream router to upstream routers. MR-MTP also forwards IP packets between servers. This is explained in the next section.

D. IP Packet Forwarding by MR-MTP

We use Fig. 2 to describe how MR-MTP forwards IP packets between servers. Let us track an IP packet from server 192.168.11.1 to server 192.168.14.1 (individual servers not shown in Fig. 2, we see only a subnet).

- When the IP packet arrives from server 192.168.11.1, the ToR VID=11 first checks the destination IP address and then uses the VID derivation algorithm (from section A) to derive the destination ToR VID.
- In this case, the destination ToR VID is 14.
- MR-MTP running in ToR VID=11 creates an MR-MTP header with the source ToR VID = 11 and destination ToR VID = 14.
- It then encapsulates the IP packet and forwards it to a tier 2 spine after executing a hash algorithm to load balance.
- Tier 2 spine checks its VID table and finds that it has no record for VID 14, but has a default forwarding path to the next tier. The packet will be sent after executing the load balancing algorithm to a tier 3 spine.
- All tier 3 spines will have an entry for VID 14 (see VID tables at S2_1 and S2_4). The top tier spine will check its VID table and the corresponding port noted against that VID will be used to forward the encapsulated IP packet to a tier 2 spine.
- That tier 2 spine (could be S1_3 or S1_4) will also have an entry for the destination VID 14 in its VID table. It will then forward the encapsulated IP packet on the port noted in its VID table.
- The encapsulated IP packet reaches the ToR VID=14. This ToR checks the destination VID and recognizes that the packet has reached the destination. It will de-encapsulate the

IP packet and forward it to server 192.168.14.1 in subnet 192.168.14.0/24

IV. Network Convergence and Availability

Datacenters process high volumes of data at high speeds. Network availability measured in service uptime and reliable data delivery are crucial to datacenters. Several factors impact network availability, such as configuration errors and hardware and software failures, among others [13, 24]. Failure of a network component such as a device, interface or link has a major impact on network availability. The convergence time subsequent to a network component failure is an important performance metric and comprises of failure detection and recovery. Failure detection requires tracking hello or keep-alive messages from a neighbor. Challenges faced during convergence are instability [32, 33] and route flapping [33, 15]. Thus a neighbor is declared inaccessible after a dead timer (normally three times the hello interval) to dampen route flapping. Routing protocols which are restricted to use a minimum hello time, use BFD and link layer failure detection [34] to speed up failure detection. During convergence, packets may get mis-delivered, delivered multiple times or lost thus disrupting service as the routing tables in the routers get updated. A network is said to converge i.e., fully recovered in response to a failure, once all devices within the failure impact scope are notified of the event and have recomputed/updated their routing tables.

A. BGP and BFD

If BGP relies solely on its keep-alive messages, it face high convergence delays, in the order of several seconds. Since links in current DCNs are predominantly point-to-point fiber connections, a physical interface failure is often detected in milliseconds. BGP hence uses BFD for sub-second failure detection. BGP failure recovery however involves dissemination of the BGP update messages and is affected by the spacing of consecutive UPDATE messages by MinRouteAdvertisementIntervalTimer (MRAI) seconds [12].

B. MR-MTP Failure Detection and Dampening

MR-MTP integrates control and data plane operations. Thus, all traffic between MR-MTP routers carry the MR-MTP header and can serve as keep-alive messages. In the event there is no MR-MTP messages to send for the duration of the keep-alive timer, a 1-byte hello message will be transmitted.

MR-MTP adopts a Quick-to-Detect, Slow-to-Accept approach to speed up failure detection and recovery. Devices running MR-MTP assume a neighbor down on missing a single hello message (Quick-to-Detect). The Slow-to-Accept dampens any toggling interface or neighbor i.e. MR-MTP will accept a neighbor up (on a failed interface) only after receiving three consecutive keep-alive messages. MR-MTP failure detection is three times faster than current methods.

Furthermore, devices receiving an MR-MTP update message, only update port entries noted against the VIDs carried in the update message. Recomputing of routes is not required. We were able to increase the frequency of MR-MTP keep-alive messages to be in par with BFD, using 2 protocols less (BFD and UDP

V. Performance Metrics

To assess DCN availability and stability (with MR-MTP and the BGP protocol suite) the following studies were conducted. An interface failure was introduced at multiple points in the test topology (see Fig. 3). Convergence time, control overhead, and blast radius was calculated as the topology converged after the failure. The packet loss incurred for traffic originating closer to the failure points and for traffic originating at the far end from the failure points were recorded. Routing tables at a spine device was used to compare the operational and memory overhead for the two protocols. To understand MR-MTP message efficiency, the keep-alive message overhead with BGP/BFD is compared with MR-MTP. Lastly, the configuration at a BGP router is compared to the configuration (for all devise) in a 4-PoD DCN running MR-MTP. In the sections that follow we explain the significance of each performance metric. The process to collect the performance metrics are described in section VI.

A. Convergence Time

This is the time taken from the instant an interface fails until the routing tables at all DCN routers stabilize.

B. Control Overhead

Subsequent to a failure, the router that detects failure disseminates update messages, which is forwarded to routers in the impact scope. Routers receiving such messages, update their routing tables and may forward to their neighbors. Control traffic (overhead) so generated to recover and re-establish paths, after a failure depends on the protocol. This was recorded in bytes.

C. Packet Loss

When traffic from one server is forwarded to another and the path used to forward this traffic fails, packets can be dropped, misdirected or duplicated. The packet loss depends on the relative position of the router and the interface failed (i.e., an upstream or a downstream interface). If the interface failure is closer to the sender of traffic the impact is different as compared to an interface failure which is farther away from the sender of traffic.

D. Router Configuration

To run BGP in DCN routers requires several configuration steps. The steps increase when the number of interfaces at a router increases i.e. when the DCN size increases. BGP routers require specific AS number assignment to devices at the different tiers to avoid packet looping. The BGP-DCN version used in our test studies required less configuration and incurred low over-head to recover from failure compared to results presented in [23], where we used eBGP configured to operate in DCN. Using BFD with BGP reduces the failure recovery impacts considerably, hence we include BFD configurations in our comparison studies.

E. Keep-Alive Overhead

The BGP/BFD keep-alive messages add to the normal traffic generated by routers. While BGP messages are carried by TCP and IP, BFD messages are carried by UDP and IP. The BGP keep-alive and BFD keep-alive have multiple fields. Introducing BFD adds to the keep-alive message traffic. The MR-MTP keep-alive message is a single byte (carried in an Ethernet frame) to inform a neighbor that it is active. MR-MTP keep-alive is generated only if there are no other MR-MTP messages exchanged between a pair of neighbors.

VI. TEST TOPOLOGIES AND TEST CASES

Fig. 3, shows a 2-PoD test topology with the interface failure points identified as TC1, TC2, TC3 and TC4. A 4-PoD test (dotted boxes in Fig.3) topology was also used with identical failure points. In both topologies, one server was supported at each server rack due to resource reservation constraints at FABRIC. The four failure test points were selected at devices in different tiers, some interfaces are upstream and some downstream.

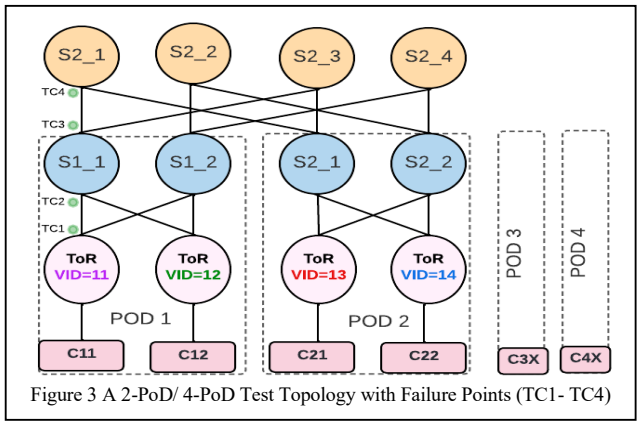

Figure 3 A 2-PoD/ 4-PoD Test Topology with Failure Points (TC1- TC4)

A. Performance Tests

For the performance tests it is important to synchronize clocks in all devices. In our performance studies we synchronized our clocks to record timings to microsecond accuracy. Tools to enable clock synchronization and software to capture frames (such as tshark [15]) and other bash scripts to facilitate data collection were uploaded to the remote site nodes (at the FABRIC test sites) in the test topologies. In the sections that follow we describe the techniques used to accurately record times of interest and collect data for the performance metrics.

B. Convergence Time

To calculate the convergence time, the interface failure time has to be recorded. The system logs provide the most accurate time when an interface goes down. The VM configuration by FABRIC, disallowed recording this event in the system logs. Hence, a bash script was written and uploaded to the target node, to fail an interface. The script, when executed at the remote target node, would bring down an interface and record the time of this event at the node. This gives the start time for the convergence calculations. From this time onwards, the route update messages were monitored. When the update messages stopped, we recorded the end time for convergence. BGP UPDATE messages on all interfaces were tracked to record the end time. For MR-MTP, print statements in the C code recorded the interface down time and the times of update messages. Our automation scripts parsed the logs to calculate the convergence time [29].

C. Control Overhead

The convergence start time recorded above was used to start collecting the update messages exchanged by MR-MTP and BGP. For MR-MTP, the messages and their size in bytes was recorded in the log files. For BGP, tshark was used to capture BGP UPDATE messages on all interfaces. The files from the remote nodes were downloaded and parsed for UPDATE messages. Total bytes transferred during the convergence time was summed up to provide BGP control overhead.

D. Packet Loss

The custom-built traffic generator is executed at both the sending and receiving servers. Sequence numbers in the packets help track lost, duplicated and out-of-sequence packets. At the receiving server, an analyzer software checked all received packets received to record lost, out-of-sequence and duplicated packets [28].

For the four test cases, we sent traffic from the server connected to ToR VID=11 to server connected to ToR VID=14 (in the 2-PoD topology in Fig. 3). We also collected packet loss for the four testcase, by sending traffic from server at ToR VID=14 to server at ToR VID=11. Calculating packets lost, misdirected or duplicated gives a more accurate estimate of the impact on server traffic flows and the loss of service as a result of an interface failure. The relative position of the interface failure with respect to the traffic flow provides insight into how failures at different points in the topology affect network traffic flows.

E. Router Configuration

Configuration errors can adversely impact network performance. Auto-assigned addresses and auto-configuration can significantly cut down such errors and reduce the load on the network administrator, especially for large DCNs. The configurations required to set up BGP in the 4-PoD folded-Clos topology and MR-MTP in an identical topology is compared. With MR-MTP, the devices only need to be configured with the tier value. To enable auto-deriving a ToR VID, the ToRs must be informed of the interface that connects the ToR to the server rack, so ToRs can determine the subnet IP address and then derive their VIDs using the algorithm coded in the ToR's MR-MTP software.

F. Keep-Alive Traffic

The keep-alive timer we used for BGP is 1 second and the dead timer is 3 times the keep-alive timer. By enabling BFD in BGP, the transmission (hello) interval could be reduced to 100 ms. Using the default detect multiplier of 3, a dead timer of 300 ms was configured. Further reduction of the keep-alive interval resulted in false failure detection. In the case of MR-

MTP, the hello timer was maintained at 50 ms and the dead timer at 100 ms. The timers have a dependency on the VMs and resource sharing by the systems at the FABRIC testbed. We tested the lowest possible hello timer for the two protocols and used them in the experiments.

VII. DCN Protocol Performance Assessment

In this section, we present the different performance graphs the four failure test cases, namely TC1, TC2, TC3 and TC4. The plotted values were averaged over multiple runs. Following are the test topologies 1) MR-MTP in a 2-PoD topology, 2) BGP/ECMP in a 2-PoD topology, 3) BGP/ECMP/BFD in a 2-PoD topology, 4) MR-MTP in a 4-PoD topology, 5) BGP/ECMP in a 4-PoD topology and 6) BGP/ECMP/BFD in a 4-PoD topology

A. Convergence Time

Fig. 4 is the plot of the network convergence time measured in milliseconds for the four failure test cases TC1, TC2, TC3 and TC4. Graphs on the left show the metrics collected for the 2-PoD topology. We notice that for the failure at TC2 and TC4, the convergence time is less than the failure detection time. This is because, when S2_1's interface connecting to S1_1 fails, S2_1 will initiate an update message as soon as the router detects the interface is down. However, when there is failure at TC1, S1_1 will detect the failure only after its dead timer expires and then initiate the update messages. Note that MR-MTP convergence times are much lower compared to BGP, even with BFD enabled. This is attributed to the fast-to-detect, slow-to-accept technique adopted by MR-MTP.

The right side of Fig. 4, shows the convergence time for the 4-PoD topology. Convergence time is predominantly controlled by the dead timer. The dissemination time and database update time are relatively low (because of the DCN size), hence there is not much difference in the convergence time calculations for the 2-PoD and 4-PoD topologies. The convergence time for BGP with BFD is better for failure at TC1 and TC3. At TC1 and TC3, S1_1 and S2_1 respectively will start the failure update after the dead time interval.

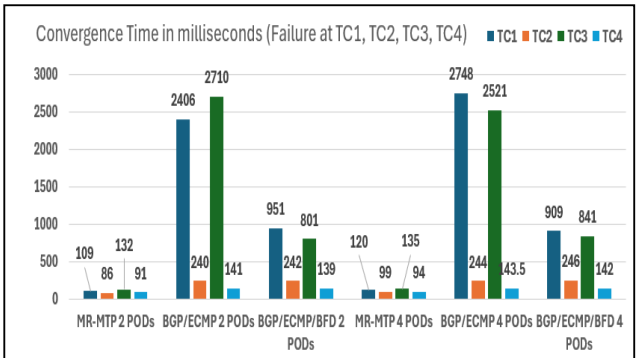

Figure 4 Convergence time in milliseconds for TC1, TC2, TC3 and TC4

B. Blast Radius

Fig. 5 shows the blast radius for the 6 different topologies and 4 test cases. This performance plot provides an insight to the instability introduced in the network on an interface failure. The metric records the number of routers that updated their routing tables subsequent to a topology change. MR-MTP uses VID tables, to route data. When a spine receives an update message that a VID is lost on that port, the VID entry from that port from the device's VID tables is removed. Comparing, BGP disseminates a route withdraw to routers in the impact scope. BGP routers receiving this message, update their routing tables and may forward to their neighbors.

Let us focus on the left side of the graph, which shows the plots for the 2-PoD topology. For a failure at TC3, TC4 (that is between tier 3 and tier 2 devices), S2_1 loses its VID derived via S1_1. Hence, S2_1 will remove any VIDs acquired from S1_1. That is, only one router updates its VID table. Traffic destined to VIDs in the PoD however can be reached via the VIDs derived from S1_2. For a failure at TC1 and TC2, ToRs with VID 12, 13 and 14 will record that a certain port cannot be used for traffic destined to VID 11. Spines along the way only forward the update message, they will not make changes to their tables. Thus, three routers record an update to their VID tables. BGP (with or without BFD) records that 9 routers made an update for a failure at TC1, TC2, while only 3 routers made a routing table update for a failure at TC3 and TC4. In the case of BGP, spine routers that forward the withdrawal message also update their tables.

A similar argument explains the number of routers that updated their tables for the 4 PoD topology with MR-MTP, BGP/ECMP with and without BFD. MR-MTP updated VID tables at 7 routers for a failure at TC1, TC2 i.e., all the ToRs made an update. For a failure at TC3, TC4, only 3 routers i.e., all the tier 2 spines except S1_1 will update a VID table. With BGP with and without BFD (BFD has no impact on the blast radius), we notice that for a failure at TC1, TC2, 15 of the 20 routers updated their routing tables. For a failure at TC3, TC4, 5 routers updated their routing table. The lesser the number of routers that update their routing table– the more stable the network.

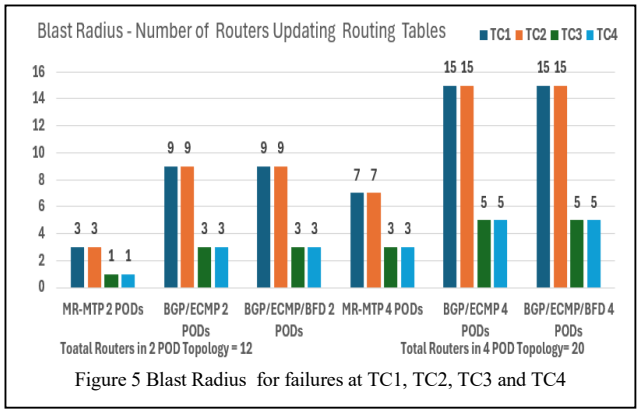

Figure 5 Blast Radius for failures at TC1, TC2, TC3 and TC4

C. Control Overhead

The control overhead in Fig. 6, is calculated by counting the bytes in layer 2 frames that carry the update messages. We summed the bytes in all the update messages following a topology change. The effect of doubling the topology size can be seen in the graph trends. For the 4-PoD topology, the

control overhead is slightly more than double for the 2-PoD topology. This is true for both protocol implementations. However, the control overhead increased to 264 bytes from 120 bytes for MR-MTP and to 2139 bytes from 1023 bytes for BGP. The overhead incurred with BGP is nearly nine times the overhead incurred with MR-MTP. As the topology size increases to realistic DCN implementations, this trend will reflect a significant increase in the control overhead generated by BGP.

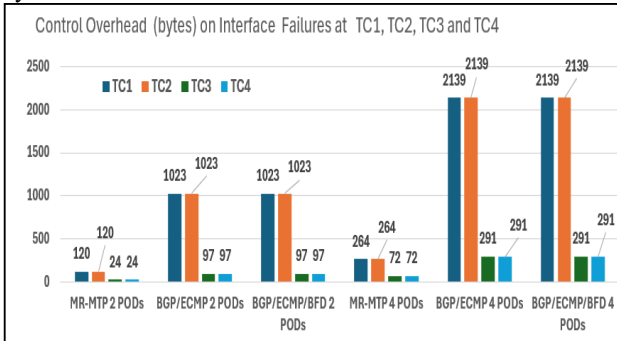

Figure 6 Control Overhead in bytes

D. Packet Loss- Traffic Sender Closer to Failure Point

Fig. 7 captures the traffic lost for the four failure test cases TC1 to TC4, when sending traffic from a compute node connected to ToR VID=11 (see Fig. 3). The significant improvement (i.e. reduction in packets lost) when using BFD with BGP is clear. As before the three plots on the left are for MR-MTP, BGP/ECMP and BGP/ECMP/BFD for the 2-PoD topology, while the three plots on the right are for the 4-PoD topology.

Packets lost with MR-MTP for failure at TC1, TC3 is very low, as the ToR and S1_1 switch the traffic to the other port on detecting a port down. There is no delay due to failure detection. For the failure at TC2 and TC4, the downstream router waits for dead timer before recognizing that the link is down and then changes the routing path for the traffic flow. Thus, more packets are lost.

With BGP/ECMP without BFD the packets lost for failures at TC1, TC3 is low around 30, while the packet loss for failure at TC2 and TC4 is around 1000 packets. BGP with BFD enabled had only one third of the packet loss as compared to BGP without BFD especially for test cases TC2 and TC4. This is because of the faster failure detection provided by BFD. Different sites were used to reserve the VMs for the 2 PoD and 4-PoD hence minor difference can be noticed.

MR-MTP performance is superior compared to BGP with or without BFD for both the topologies. We attribute this to the fast-to-detect, slow-to-accept failure detection and recovery adopted by MR-MTP.

E. Packet Loss- Traffic Sender Away from Failure

Fig. 8 shows the packets lost when the traffic flows from the servers at the far end – i.e., away from the failure points to the servers closer to the failure points. More packets are lost at the interface failures at TC1 and TC3 for both the protocols. When interfaces failed at TC1 and TC3, the routers forwarding the traffic from an upstream router were unaware of the failure until the upstream router missed keep-alives for the duration of the dead timer. Other trends are similar to the trends observed in the last section. Enabling BFD with BGP has a profound affect on reducing the packet loss. MR-MTP consistently faces low packet loss.

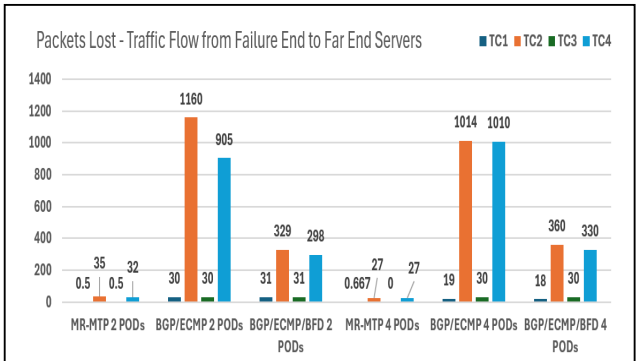

Figure 7: Packet Loss – Traffic Flow from Failure End to Far End Servers

F. Keep- Alive Message Overhead

With BFD, the transmission interval was set to 100 ms. The dead timer was 3 times the keep-alive interval. Further reduction resulted in false failures (there was a dependency on the VM and its resources, the same resource configuration was used both for Mr-MTP and BGP/ECMP with and without BFD). Though BFD takes over the neighbor monitoring, BGP continues to send keep-alive messages. Included in BGP communications is TCP acknowledgements. Fig. 9 shows a capture of the BFD Hello messages. In our captures we include a couple of BFD and BGP messages to show their interleaving. One BFD message is expanded to show the different fields in the message. Each BFD message occupies 66 bytes, and a BGP message occupies 85 bytes (at layer 2). These contribute to the control overhead even during normal operations

Fig. 10 shows the MR-MTP keep-alive messages sent every 50 ms. We used the protocol type 0x8850, (an unused protocol type) for MR-MTP in the Ethernet header. MR-MTP uses one byte (value=06) that informs the receiving neighbor that the sending neighbor is alive. Though MR-MTP may send more keep-alive messages the overhead incurred is very low. In addition, all MR-MTP messages can serve as keep-alive messages. Hence, when MR-MTP carries traffic between servers, all these messages will be considered as keep-alive. In the MR-MTP captures, as there are no wireshark drivers to interpret MR-MTP fields we see only the hexadecimal values. The destination MAC address used in frames carrying MR-MTP message is ff:ff:ff:ff:ff:ff - the broadcast address. In the current DCN scenario this is acceptable because the links are point to point and the frame is delivered only to the device at the receiving end of the link. Using the broadcast address avoids the need for Address Resolution Protocol (ARP).

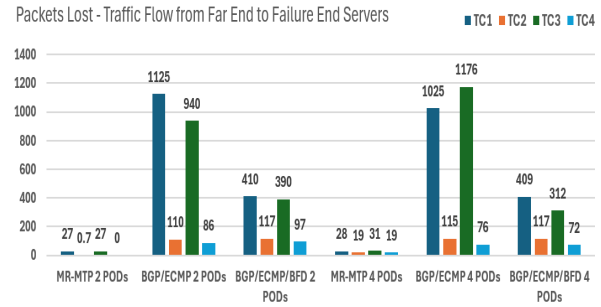

Figure 8: Packet Loss – Traffic Flow from Far End to Failure End Servers

.

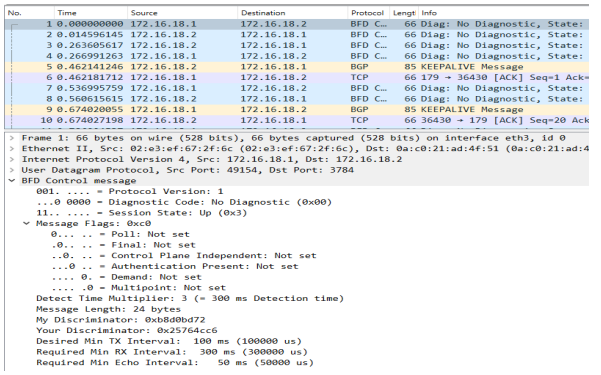

Figure 9: Keep-Alive overhead with BGP/BFD/TCP (Wireshark capture)

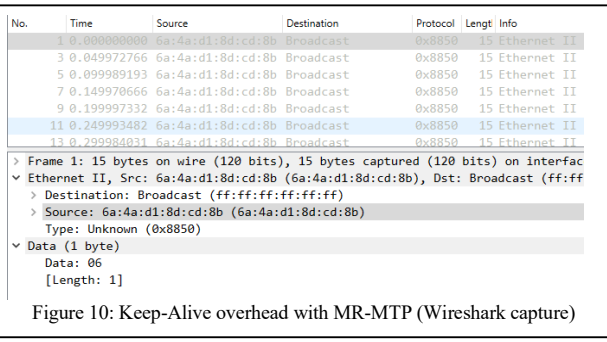

Figure 10: Keep-Alive overhead with MR-MTP (Wireshark capture)

G. Configuration

Setting up a single BGP router (example T-1 in Fig.3) in a folded-Clos topology requires the steps in listing 1. This an abbreviated configuration captured using 'show running configuration' command. As the number of BGP routers increase, the configuration required will increase linearly.

MR-MTP routers in a 4-PoD folded-Clos topology use the configuration in listing 2. The configuration is provided in a JSON file and used to set up MR-MTP in the all nodes at different tiers. This file informs the nodes their tier position in the topology. In its current implementation, MR-MTP running at the ToRs has to be informed of the interface connecting to the server rack. This information is included in the leavesNetworkPortDict and is used by ToRs to derive their VID.

```
frr version 10.0
frr defaults datacenter hostname T-1
log file /var/log/frr/bgpd.log
log timestamp precision 3
no ipv6 forwarding
debug bgp updates in debug bgp
updates out debug bgp updates detail
router bgp 64512
 timers bgp 1 3
 neighbor 172.16.0.2 remote-as 64513
 neighbor 172.16.0.2 bfd
 neighbor 172.16.1.2 remote-as 64514
 neighbor 172.16.1.2 bfd
 neighbor 172.16.2.2 remote-as 64515
 neighbor 172.16.2.2 bfd
 neighbor 172.16.3.2 remote-as 64516
 neighbor 172.16.3.2 bfd
bfd
 profile lowerIntervals
  transmit-interval 100
 peer 172.16.0.2
  profile lowerIntervals
 peer 172.16.1.2
  profile lowerIntervals
 peer 172.16.2.2
  profile lowerIntervals
 peer 172.16.3.2
  profile lowerIntervals
```

LISTING 1: BGP Configuration at Router T-1

H. Routing Table Size

Routing table from a tier 2 spine is provided in listings 3 for a BGP router. The routing table size reflects both the storage needs and the protocol processing time at the routers. As the size of the network increases, a proportional increase in the routing table sizes will be noticed.

In listing 5, MR-MTP's VID table in a top tier spine of 4 PoD topology is shown. The top tier spine has four interfaces connecting to the 4 PoDs. At each PoD, there are two server subnets. The VID table records the VIDs derived from the four ToR VIDs followed by the port numbers on which they were acquired. This information is used to route/forward an MR-MTP encapsulated IP packets to the destination ToR VID.

```
topology:
  leaves: [L-1-1,L-1-2,L-2-1,L-2-2,L-3-1,L-3-2,L-4-1,L-4-2],
  leavesNetworkPortDict:
    L-1-1 :     eth3,
    L-1-2 :     eth3,
    L-2-1 :     eth3,
    L-2-2 :     eth3,
    L-3-1 :     eth1,
    L-3-2 :     eth3,
    L-4-1 :     eth3,
    L-4-2 :     eth2
    topSpines : [ T-1 , T-2 , T-3 , T-4 ],
      pods : [
              topSpines : [ S-1-1 , S-1-2 ]
              topSpines : [ S-2-1 , S-2-2 ]
              topSpines : [ S-3-1 , S-3-2 ]
              topSpines : [ S-4-1 , S-4-2 ]
              ]
```

LISTING 2: MR-MTP 4-PoD json file to configure all Routers

VIII. Conclusion

Networks and related technologies have experienced tremendous growth over the last couple of decades. Network architectures and topologies have evolved accordingly to address the continuously growing communication needs and demands. However, the industry still continues to use well-proven and tested techniques and protocols from yesteryears. A major challenge to adopt new techniques and technologies is the need for backward compatibility with IP and Ethernet.

In this work, we tested a non-IP solution for the DCNs. We introduced a novel routing technique and protocol for use in folded-Clos topology DCNs. The proposed solution, MR-MTP, is IP and Ethernet agnostic, but is backward compatible to IP and Ethernet. We tested MR-MTP's C code implementation on VMs running Rocky Linux reserved on the FABRIC testbed. BGP/ECMP with and without BFD were also executed in identical topologies on the Rocky Linux VMs on the FABRIC testbed. We used BGP/ECMP/BFD software

```
172.16.0.0/24 dev eth3 proto kernel scope link src 172.16.0.2
172.16.8.0/24 dev eth4 proto kernel scope link src 172.16.8.2
172.16.16.0/24 dev eth2 proto kernel scope link src172.16.16.1
172.16.17.0/24 dev eth1 proto kernel scope link src172.16.17.1
192.168.0.0/24 via 172.16.16.2 dev eth2 proto bgp metric 20
192.168.1.0/24 via 172.16.17.2 dev eth1 proto bgp metric 20
192.168.2.0/24 proto bgp metric 20
  nexthop via 172.16.0.1 dev eth3 weight 1
  nexthop via 172.16.8.1 dev eth4 weight 1
192.168.3.0/24 proto bgp metric 20
  nexthop via 172.16.0.1 dev eth3 weight 1
  nexthop via 172.16.8.1 dev eth4 weight 1
192.168.4.0/24 proto bgp metric 20
  nexthop via 172.16.0.1 dev eth3 weight 1
   nexthop via 172.16.8.1 dev eth4 weight 1
192.168.5.0/24 proto bgp metric 20
  nexthop via 172.16.0.1 dev eth3 weight 1
  nexthop via 172.16.8.1 dev eth4 weight 1
192.168.6.0/24 proto bgp metric 20
  nexthop via 172.16.0.1 dev eth3 weight 1
   nexthop via 172.16.8.1 dev eth4 weight 1
192.168.7.0/24 proto bgp metric 20
  nexthop via 172.16.0.1 dev eth3 weight 1
  nexthop via 172.16.8.1 dev eth4 weight 1
```

LISTING 3: Tier 2 Spine BGP Routing Table

from FRRouting. From the results presented it will be clear that MR-MTP a single simple protocol superior in performance compared to BGP/ECMP with and without BFD for the 2 Pod and 4 PoD topologies. The performance improvement with MR-MTP can be noted in all tests that we conducted.

The design and operation of MR-MTP is very simple

```
eth1  33.1.1,  34.1.1
eth2  35.1.1,  36.1.1
eth3  37.1.1,  38.1.1
eth4  39.1.1,  40.1.1
```

LISTING 4: MR-MTP VID table at Router T-1

which makes the protocol robust. Reliability is built in the MR-MTP message exchanges, which replaces the need for TCP. MR- MTP can replace BGP, ECMP, BFD, TCP, UDP and IP (i.e. 6 protocols) and still offer superior performance as provided in his article. MR-MTP requires reduced configurations, reduced memory and processing needs. Some of the benefits of these features are listed below.

- The cost of the equipment will reduce significantly as the hardware and software requirements to implement an MR-MTP router will reduce.
- The energy consumption per router and by the DCN will reduce significantly.
- Autoconfiguration and auto addressing will reduce the configuration steps and this will reduce human errors and misconfigurations.
- The above benefits will increase multiplicatively as the DCN size increases.
- The performance benefits will be more significant as the size of the DCN increases.
- The MR-MTP DCN routers are not running BGP, TCP and IP – which will reduce the possibility of security attacks on the DCN. Simple rules to allow only IP traffic at interfaces connecting to compute nodes and gateways can protect the DCN.

IX. Future Work

The FABRIC testbed offers a powerful network experimentation facility. The testbed provides an API to access and set up topologies as well as conduct tests. We added scripts tailored to our experiment needs. The resource reservation constraints, however, limited our test topologies to a maximum of 4 PoD, where each server rack had one server. These facilities were adequate to provide strong validations of our work. Future work extending and testing MR-MTP includes the following:

- Scaling the DCN to multiple tiers using Mininet [30].
- Tune timers for optimal performance of the protocols.
- Extended failure test cases.
- Future tests will also include overhead calculations of using the MR-MTP header for every IP packet and overhead calculations due to all protocols such as BGP, TCP, BFD and UDP will be considered for comparison.
- Every MR_MTP message will be a keep alive, which will cut down on the keep alive overhead incurred in current protocols

References

[1] FRRouting. https://frrouting.org/ - accessed 2024

[2] Anubhavnidhi Abhashkumar et al, RunningBGP in Data Centers at Scale. In Proceedings of the 18th USENIX Symposium on Networked Systems Design and Implementation. Boston, MA, 65–81.

[3] Hussam Abu-Libdeh et al, Symbiotic routing in future data centers. In Proceedings of the ACM SIGCOMM 2010 conference, New Delhi, India, 51–62. https://doi.org/10.1145/1851182.1851191

[4] Mohammad Al-Fares, Alexander Loukissas, and Amin Vahdat. A scalable, commodity data center network architecture. In Proceedings of the ACM SIGCOMM 2008 conference on Data communication New York, NY, USA, 63–74. https://doi.org/10.1145/1402958.1402967

[5] Ilya Baldin et al, FABRIC: A national-scale programmable experimental network infrastructure. IEEE Internet Computing 23, 6 (2019), 38–47.

[6] Kashif Bilal et al, Quantitative comparisons of the state-of-the-art data center architectures. Concurrency and Computation: Practice and Experience 25 Dec. 2012, 1771–1783. https://doi.org/10.1002/cpe. 2963

[7] Kai Chen et al, Survey on routing in data centers: insights and future directions. IEEE Network, 4 July 2011, 6–10. https://doi.org/10.1109/MNET.2011.5958002

[8] Dinesh G Dutt. 2020. Cloud Native Data Center Networking: Architec- ture, Protocols, and Tools (first ed.). O'Reilly Media, Inc, Sebastopol, CA, USA

[9] Chuanxiong Guo et al, BCube: a high performance, server-centric network architecture for modular data centers. In Proceedings of the ACM SIGCOMM 2009 conference on Data communication Barcelona, Spain, 63–74. https://doi.org/10.1145/1592568. 1592577

[10] Chuanxiong Guo et al, Dcell: a scalable and fault-tolerant network structure for data centers. ACM SIGCOMM Computer Communication Review, Aug. 2008, 75–86. https://doi.org/10.1145/1402946.1402968

[11] Dave Katz and David Ward. 2010. Bidirectional Forwarding Detection (BFD). RFC 5880. https://doi.org/10.17487/RFC5880

[12] Petr Lapukhov, Ariff Premji, and Jon Mitchell. 2016. Use of BGP for Routing in Large-Scale Data Centers. RFC 7938. https://doi.org/10. 17487/RFC7938

[13] Yang Liu et al, Data Center Networks: Topologies, Architectures and Fault-Tolerance Characteristics, Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-01949-9

[14] Mallik Mahalingam et al, Virtual eXtensible Local Area Network (VXLAN): A Framework for Overlaying Virtualized Layer 2 Networks over Layer 3 Networks. RFC 7348. https://doi.org/10.17487/RFC7348

[15] Zhuoqing Morley Mao et al, Route flap damping exacerbates internet routing convergence. In Proceedings of the SIGCOMM 2002 conference on Applications, technologies, architectures, and protocols for computer communications (SIGCOMM '02). Pittsburg, PA, USA, 221–233. https://doi.org/10.1145/633025.633047

[16] Nicola Modena, Controlling BGP Convergence Time. https://www.ipspace.net/kb/BGPHighAvailability/30-Controlling-BGP-Convergence.html

[17] Radhika Niranjan Mysore et al, PortLand: A scalable fault-tolerant layer 2 data center network fabric. ACM SIGCOMM Computer Communication Review 39, 4 (Aug. 2009), 39–50. https://doi.org/10.1145/1594977.1592575

[18] Keyur Patel, Acee Lindem, Shawn Zandi, and Wim Henderickx. BGP Link-State Shortest Path First (SPF) Routing. Internet-Draft draft-ietf-lsvr-bgp-spf-29. Internet Engineering Task Force. https://datatracker.ietf.org/doc/draft-ietf-lsvr-bgp-spf/29/

[19] Leon Poutievski et al, Jupiter evolving: transforming Google's datacenter network via optical circuit switches and software-defined networking. In Proceedings of the ACM SIGCOMM 2022 Conference Amsterdam, Netherlands, 66–85. https://doi.org/10.1145/3544216.3544265

[20] Tony Przygienda et al, RIFT: Routing in Fat Trees. Internet-Draft draft-ietf-rift-rift-23. Internet Engineering Task Force. https://datatracker.ietf.org/doc/draft-ietf-rift-rift/23/

[21] Arjun Singh et al, Jupiter Rising: A Decade of Clos Topologies and Centralized Control in Google's Datacenter Network. In Proceedings of the ACM SIGCOMM 2015 Conference, Vol. 45. London, United Kingdom, 183–197. https://doi.org/10.1145/2829988.2787508

[22] Peter Willis, Miaoxin Li, and Nirmala Shenoy, Performance of Meshed Tree Protocol in Data Center Networks. In Proceedings of the 2nd ACM SIGCOMM Workshop on Future of Internet Routing & Addressing (FIRA '23). Association for Computing Machinery, New York, NY, USA, 15–22. https://doi.org/10.1145/3607504.3609290

[23] Peter Willis and Nirmala Shenoy, Meshed Tree Routing in Folded- Clos Topologies. In ACM SIGCOMM 2022 Workshop on Future of Internet Routing & Addressing (FIRA '22). ACM, Amsterdam, Netherlands, 86–91. https://doi.org/10.1145/3527974.3545717

[24] Wenfeng Xia, Peng Zhao, Yonggang Wen, and Haiyong Xie, A Survey on Data Center Networking (DCN): Infrastructure and Operations. IEEE Communications Surveys & Tutorials 19, 1 (2017),640–656. https://doi.org/10.1109/COMST.2016.2626784 Conference Name: IEEE Communications Surveys Tutorials.

[25] Kok-Kiong Yap et al, Taking the Edge off with Espresso: Scale, Reliability and Programmability for Global Internet Peering. In Proceedings of the SIGCOMM 2017 Conference of the ACM Special Interest Group on Data Communication, Los Angeles, California, 432–445. https://doi.org/10.1145/3098822.3098854

[26] Gerald Combs. 2024. Wireshark. https://www.wireshark.org/

[27] Vincent Li and Peter Willis. 2024. CMTP. https://github.com/pjw7904/CMTP.

[28] Peter Willis. 2024. Basic Traffic Generator. https://github.com/pjw7904/Basic-Traffic-Generator

[29] Peter Willis. FABRIC Automation. https://github.com/pjw7904/FABRIC-Automation

[30] Peter Willis. https://github.com/pjw7904/CMTP/mininet

[31] CDW Corp. 2017. Hyperconverged Infrastructure: The Power to Simplify IT. White Paper. https://www.cdw.com/content/cdw/en/ articles/datacenter/white-paper-hyperconverged-infrastructure- data-center.html

[32] Nick Feamster and Hari Balakrishnan. 2005. Detecting BGP configuration faults with static analysis. In Proceedings of the 2nd conference on Symposium on Networked Systems Design & Implementation (NSDI'05, Vol. 2). USENIX Association, Anaheim, California, 43–56.

[33] Craig Labovitz, Abha Ahuja, Abhijit Bose, and Farnam Jahanian. Delayed Internet routing convergence. In Proceedings of the ACM SIGCOMM 2000 conference on Applications, Technologies, Architectures, and Protocols for Computer Communication Stockholm, Sweeden, 175–187. https://doi.org/10.1145/347059.347428

[34] IP Routing: BGP Configuration Guide - BGP Support for BFD https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_bgp/configuration/xe-16/irg-xe-16-book/bgp-support-for-bfd.html