Cross-Over Spanning Trees

Enhancing Metro Ethernet Resilience and Load Balancing (Invited Paper)

Minh Huynh and Prasant Mohapatra
Computer Science Department
University of California at Davis
Davis, CA. USA.
{mahuynh, pmohapatra}@ucdavis.edu

Stuart Goose
Technology To Business
Siemens Corporation
Berkeley, CA. USA
sgoose@ttb.siemens.com

***Abstract*— The economics and familiarity of Ethernet technology is motivating the vision of wide-scale adoption of Metro Ethernet Networks (MEN). Despite the progress made by the community on additional Ethernet standardization and commercialization of the first generation of MEN, the fundamental technology does not meet the expectations that carriers have traditionally held in terms of network resiliency and load management. These two important features of MEN have been addressed in this paper. We propose a new concept of Cross-Over Spanning Trees (COST) that increases the resiliency of the MEN while provisioning the support for load balancing. As a result, the capacity in terms of network throughput is greatly enhanced while almost avoiding any re-convergence time in the case of failures. The gain ranges from 1.69% to 7.3% of the total traffic in the face of failure; while load balancing increases an additional 12.76% to 37% of the total throughput.**

Keywords- Cross-Over Spanning Trees, Load Balancing, Metro Ethernet Network, MSTP, Resilience, RSTP.

I. INTRODUCTION

The most common technology used for local area networks is the Ethernet protocol, which has been the predominant technology for more than 30 years. Ethernet is a simple and cost-effective protocol that provides a variety of services. Despite the occasional challengers, such as fibre channel and infiniband architecture, the evolution of Ethernet has continued. The recent standardization of Gigabit Ethernet [13] protocol has propelled it for consideration in the scope of metropolitan area networks. Moreover, several companies are focusing their developments, products, and services for Metro Ethernet Networks (MEN).

MENs [12] comprise a metro core network and several access networks. All the access networks connect to the core at one or two gateway Ethernet switches. The customers' networks are connected to the access network, and the metro core helps in interconnecting the access networks. Packets hop through multiple switches in both access and metro core networks. Redundant links are used in the core as well as the access networks.

The main challenges in the context of MEN include resiliency, load balancing, and support for QoS. Current Ethernet solutions deploy the Spanning Tree Protocol and its variants to manage the topology autonomously. However, they are inadequate in all of the three areas. In this work, we address the resiliency and load balancing aspects of MEN. We have introduced a novel approach, called Cross-Over Spanning Tree (COST) protocol, which allows switching between multiple Spanning Trees without forming any cycles. This feature enhances the resiliency as well as facilitates load balancing. In addition to fast recovery, it also increases the capacity of the network in terms of the achievable throughput.

We believe that COST has the potential to provide enhanced services with a low overhead. The encouraging experimental results presented in this paper were obtained using the OPNET [11] simulation product to quantify the resiliency and the gain in terms of the network throughput. The behaviors of the Ethernet switches within OPNET Modeler were modified to imbue the COST approach. In the resilience test scenarios, COST yields an increase of 1.69% and 7.3% of the total traffic comparing to Multiple Spanning Tree Protocol (MSTP) and Rapid Spanning Tree Protocol (RSTP), respectively. In addition, when the network is overloaded and imbalanced, COST gains an additional 12.76% and 37% of the total traffic comparing to MSTP and RSTP, respectively.

The organization of the paper is as follows: a preliminary section explains the current state of Ethernet and the motivation for COST. It is followed by a description of the concept of COST. COST is then evaluated separately in two areas: resilience and load balancing. Finally, related works are presented before the conclusion of the paper.

II. MOTIVATION

Traditionally, Ethernet-based networks use Spanning Tree Protocol (STP)[1], standardized in IEEE 802.1d, for switching frames in the network. STP is a layer 2 protocol that can be implemented in switches and bridges. Essentially, it uses a shortest-path approach in forming a tree that is overlaid on top of the mesh-oriented Ethernet networks. Spanning tree is used primarily to avoid the formation of cycles, or loops, in the network. Unlike IP packets, Ethernet data frames do not have a time-to-live (TTL) field. STP prevents loops in the network by blocking redundant links. Therefore, the load is concentrated on a single link which leaves it at risk of failures and with no load balancing mechanism. The root of the tree is chosen based on the bridge priority, and the path cost to the root is propagated throughout so that each switch can determine the state of its ports. Only the ports that are in the forwarding state can forward incoming frames. This ensures a shortest single path to the root. Whenever there is a change in the topology, switches rerun the protocol that can take 30 to 60 seconds. At

any one time, only one Spanning Tree dictates the network. Although STP has been used for most Ethernet networks, it has several serious shortcomings in the context of its use for MEN. These shortcomings are enumerated as follows:

1. Low Utilization: Spanning trees restrict the number of ports being used. In high-capacity Ethernets, this restriction translates to a very low utilization of the network.
2. Poor Resilience: a very high convergence time (30s to 60s) after a link failure.
3. No Load Balancing
4. No support for QoS.

An improvement of STP is the Rapid Spanning Tree Protocol RSTP [2] specified in IEEE 802.1w. RSTP reduces the number of port states from five in STP to three: discarding, learning, and forwarding. Through faster aging time and rapid transition to forwarding state, RSTP is able to reduce the convergence time to between 1 and 3 seconds. It is understood that depending on the network topology, this value varies. In addition, the topology change notification is propagated throughout the network simultaneously, unlike STP, in which a switch first notifies the root, then the root broadcast the changes. Similar to STP, there is only one Spanning Tree over the entire network. RSTP still blocks redundant links to ensure loop free paths leaving the network underutilized, vulnerable to failures, and with no load balancing.

MSTP or Multiple Spanning Tree Protocol [4] is defined in IEEE 802.1s. MSTP uses a common Spanning Tree that connects all of the regions in the topology. The regions in MSTP are instances of the RSTP. An instance of RSTP governs a region, where each region has its own regional root. The regional roots are in turn connected to the common root that belongs to the common Spanning Tree. Since MSTP runs pure RSTP as the underlying protocol, it inherits some drawbacks of RSTP as well. However, a failure in MSTP can be isolated into a separate region leaving the traffic flows in other regions untouched. In addition, the administrators can perform light load balancing manually by assigning certain traffic sources to a specific Spanning Tree.

III. Conceptual Approach to COST

In this section, we describe the basic philosophy behind the COST protocol and its potential for provisioning enhanced performance and services.

A. *COST Philosophy*

In the Spanning Tree protocol and most of its variants, at any point of time, only one Spanning Tree is used. The use of this Spanning Tree is facilitated by blocked ports in various combinations in each of the Ethernet switches. Although many protocols have proposed the enhancement of the basic STP, they still use only a single Spanning Tree for one flow at any point of time in any segment of the network. These protocols take relatively longer to recover from faults and also have no support for balancing load across the network.

The primary motivation behind the design of COST is to allow the flexibility of using more than one Spanning Tree while a flow is *en-route* to its destination. This flexibility allows the usage of more ports per switches. However, to avoid the formation of cycles in the network, certain restrictions are imposed.

The basic methodology for implementing the COST philosophy is to identify multiple Spanning Trees and number them sequentially to form an ordered list. The VLAN ids [3] can be used as the sequences for the Spanning Trees. Frames of a flow start using one Spanning Tree and if necessary, can be switched over to the next Spanning Tree (none of the other variants of Spanning Tree allow this flexibility) in sequence. This procedure can be repeated until the frame reaches the Spanning Tree which has the highest id in the sequence, as seen in Figure 1. At no point in time is a frame allowed to change from a Spanning Tree with a higher id to a Spanning Tree with a lower id. A flow is switched, or crossed-over, from one Spanning Tree to another whenever there is a link failure, or load imbalance. Note that in rare cases, all flows may reach the Spanning Tree with the highest id in the sequence. This unlikely event happens when there are a large number of failures without any recovery. The handling of such rare events are discussed in Section IIIC. Since a VLAN has a one-to-one mapping to a Spanning Tree, these terms are used interchangeably.

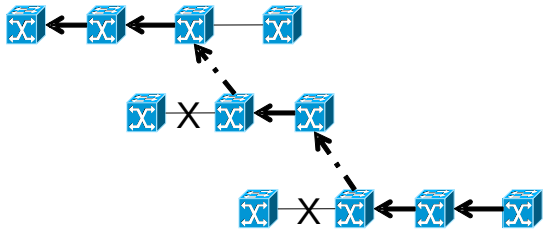

Figure 1 A sketch of the 3 layers of Spanning Tree. Prior to a failed link (indicated by a cross), COST elevates the traffic (indicated by dashed lines) to the next highest Spanning Tree in the sequence.

B. *Loop Free Guarantee*

As mentioned earlier, a cross-over from one Spanning Tree to another is allowed *only* from a lower numbered Spanning Tree (ST) to a higher one; the reverse cross-over is not permitted. Therefore, infinite loops cannot occur in COST because of this monotonic increase property. In other words, COST does not lead frames into an infinite loop when switching between Spanning Trees.

Initially, all traffic in COST starts on the first ST. All the time no problems occur, the frames remain on the first ST. If a problem occurs with a link on the path of the first ST, COST switches to the second ST. Since COST prohibits frames to be switched, or demoted, from a higher ranked tree to a lower ranked tree, a flow cannot be switched to the first ST. Therefore, it remains on this second ST until either delivered or dropped.

COST forms multiple Spanning Trees by creating a set of single independent Spanning Trees by running RSTP for each ST. Each of these STs is guaranteed to be loop free because the RSTP blocks all the redundant links. It is possible that a frame might repeat the link they have traversed once before, but will not become stuck in an infinite loop as it will not repeat any

path supervised by the first ST. The loop free property is guaranteed because the second ST is independent from the first ST, and it is also guaranteed individually by the RSTP loop free property. By induction, the loop free property is guaranteed providing COST switches to different STs.

For example, let there be 3 Spanning Trees on 4 nodes: ST1 (A-B-C-D), ST2 (A-C-B-D), ST3 (D-A-B-C). To go from A to D, initially, ST1 is used. If link C-D is down, the traffic is switched to ST2 at node C; therefore, the path is now ABCBD. We have a local loop at B-C, but it is only transient. Later, if link B-D breaks, at B, the traffic is switch to ST3 so that the new path is ABCBAD. The local loops occur at ABCBA. Even though the frames revisit the nodes B and A creating a loop, it is there temporary so that it can be switch to the next tree exiting the loop. In addition, the local loops do not affect or create problems for the backward address learning process. Since the addresses are learn per VLAN and each VLAN is associated with a Spanning Tree, the switching does not create the ping-pong effect when forwarding frames. Each VLAN only know its own learned addresses on the original port. Therefore, it will not see the local loops.

C. Implementation Issues

When implementing COST, there are other issues that must be taken into consideration. First, COST must be backward compatible with current protocol. As a consequence, the MSTP protocol, 802.1s, was leveraged to implement the functionality and operations needed by COST. Since MSTP is backward compatible with RSTP and STP, COST can interoperate with these and MSTP. Thus, COST retains the advantages of MSTP while providing enhanced features in terms of resiliency, capacity, and load balancing.

The decision as to whether a frame should be elevated to the next Spanning Tree is performed on a per frame basis. The reassignment of a frame to the next Spanning Tree occurs in the same time period as a write to the frame header. As each frame arrives at a switch, the switch exams the outgoing link of the current frame to determine the network condition, as shown in Figure 2. Thus, conditions such as failure and load imbalance are detected locally by each switch remaining faithful to the nature of Ethernet. Therefore, when a problematic link is detected, only a subset of the end-to-end path is needed in rewriting the header for the affected flows. As a result, it is possible for a flow to be on multiple STs at one time. For example, a flow traverses ST1 on path A-B-C-E and a problem occurs at link B-C. Without any user intervention, the flow is adjusted so that a new path is chosen, A-B-D-E. From A-B, the flow remains with ST 1, but from B through E, the flow is crossed over to ST 2.

To optimize the benefit of having multiple paths, the root for each Spanning Tree is chosen to be unique, if possible. In other words, to the extent possible each instance of the Spanning Tree avoids sharing the regional root. Since the Spanning Tree protocol uses the shortest path to the root approach, having unique roots increases the chances of constructing disjoint trees. Works focused on creating robust Spanning Trees include [5], [7], [9], and [10].

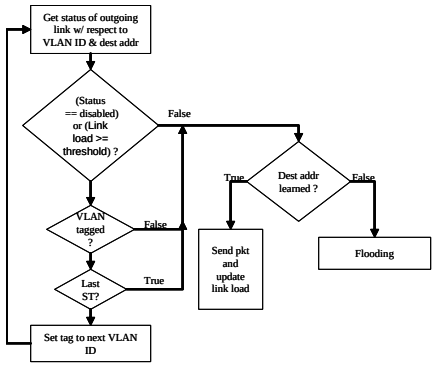

Figure 2 Pseudocode for COST

The primary performance enhancement of COST is the avoidance of the lengthy re-convergence procedure. Therefore, the reconvergence behavior of MSTP is adapted in the following ways:

1. When a switch detects a fault or a link recovery on one of its ports, the Spanning Tree Algorithm (STA) no longer initiates the port state/ role re-selection.
2. The STA no longer flushes entries in the filtering database and forwarding tables. The switch acts as if nothing has happened and the traffic is switched to the new Spanning Tree for a "soft re-convergence".
3. When a link recovers, instead of setting the recovered port to blocking state and performing the re-convergence, the switch reinstates the original role of the port per Spanning Tree.

After a prolonged period of operation, it is possible that a significant proportion of the traffic is flowing on the last available (highest elevation) Spanning Tree due to multiple failures without any corresponding recoveries. As COST is prohibited from switching traffic to a lower order Spanning Tree, in the worst case scenario COST performance degrades to that of standard RSTP. Each switch monitors for this condition by keeping track locally: of any failure resulting in flows being elevated the next Spanning Tree and the load on the highest Spanning Tree. If the load exceeds the predetermined threshold, the switch will broadcast a topology reconvergence on the affected tree. If switches receive at least two of such messages from distinct switches reconvergence will be triggered. Switches are then permitted to enter a self-reconfiguring state by reelecting state/port role, flushing the filtering database and the forwarding tables as before. Therefore, COST remains faithful to the decentralized nature of Ethernet.

The original intention of having VLAN tags is for isolating traffic. As STEP uses VLAN tags as ids for STs, the original objective of VLANs is preserved. Instead of mapping a VLAN id to a traffic group, STEP can be implemented to map a set of VLAN ids that represent the Spanning Trees to a traffic group. The VLAN partition is implementation dependent. The shortage of VLAN ids can be an issue, but there are proposals to perform VLAN stacking or Q-in-Q [14] and [15]. This technique increases the number of VLAN tag from 212 to 224. Viking [5], a related work, also uses VLAN tag as the identification for multiple Spanning Trees.

IV. SIMULATION DESIGN

The OPNET [11] simulator tool was chosen because of its comprehensive implementation of Ethernet. OPNET includes implementation of RSTP, MSTP, and VLAN which are crucial to the evaluation of COST.

COST will be evaluated on two topologies: a topology representative of Metro Area Network (MAN) [16], and a 6x6 grid topology as seen in Figure 3 and Figure 4, respectively. A grid topology which inheritably contains high degree nodes is put to the test to show the impact of COST on various topologies. Providing multiple alternative paths exist, the network performance will yield the benefits of COST.

In the MAN topology of Figure 3, RSTP has only a single Spanning Tree configured on each side of the router. The initial RSTP Spanning Tree is shown in Figure 20 (Appendix). The root of the Spanning Tree is located at the switch **core6**. By contrast, MSTP and COST have four Spanning Trees configured: the common root is at **core6** but the regional root for **vlan10 (ST 1)** and **vlan40 (ST 4)** is at **core1**, the regional root for **vlan20 (ST 2)** and **vlan30 (ST 3)** is at **core2**. The Spanning Tree configuration can be viewed in Figure 20 through Figure 23 (Appendix). Each VLAN represents one Spanning Tree. Similar to RSTP, the Spanning Tree stops at the router.

Likewise, RSTP has one Spanning Tree operating in the 6x6 grid topology. The root of the tree is located at **node_14** which is center of the topology. Conversely, MSTP and COST is configured with six Spanning Trees. The common root is at **node_33**. The regional roots for **ST 1** through **ST 6** are in the following order: **node_33**, **node_9**, **node_3, node_15**, **node_21**, **node_27**. As specified in the 802.1D standard, there are a maximum of 7 hops. In order to form a stable Spanning Tree for a topology of this size, the "hop count" parameter is increased.

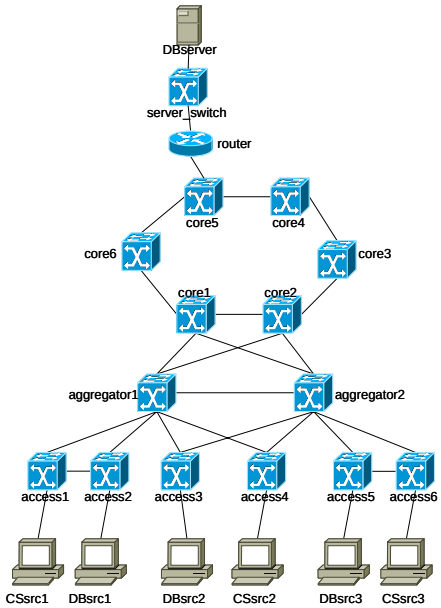

Figure 3 A representative Metro Area Network (MAN) topology

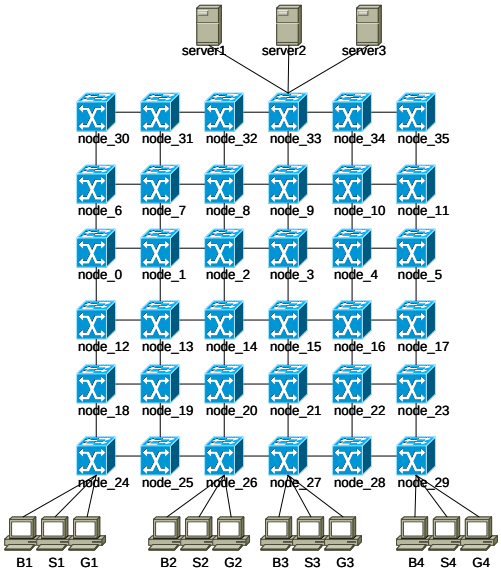

Figure 4 A 6x6 grid topology

A. *Metro Area Network Topology*

Using the MAN topology in Figure 3, resilience and load balance is evaluated. The description and specific parameters used are included. The notation <–> indicates the link between two objects.

1) *Failure Scenarios*

There are 6 traffic flows from CSsrc{1, 2, 3} and DBsrc{1, 2, 3} to DBserver with each flow is a video conferencing session that starts after 100s has elapsed, thus allowing the standard Spanning Tree initialization to complete. All links have capacity of 1Gbps. The simulation runs for a duration of 240s. The link failures and link recovery are scheduled as follows:

120s: aggregator1 <–> core1 fails
140s: aggregator1 <–> core2 fails
180s: aggregator2 <–> core1 fails
220s: aggregator1 <–> core1 recovers

The results of the simulation experiment for RSTP, MSTP and COST are presented in the next section. Cumulative throughput is used as the metric for comparing the resilience as the difference in throughput clearly illustrates the performance loss for each of the protocols.

For MSTP, each traffic source is assigned to a Spanning Tree in a round robin fashion from left to right in Figure 3 as shown in TABLE I. For example, CSsrc1, DBsrc1, DBsrc2, CSsrc2, DBsrc3, and CSsrc3 are assigned to **vlan10**, **vlan20**, **vlan30**, **vlan40**, **vlan10**, and **vlan20**, respectively. **Vlan10**, **vlan20**, **vlan30**, and **vlan40** each correspond to a different Spanning Tree. COST has the same traffic profiles as MSTP except that all sources initially begin with **vlan10 (ST 1)**.

TABLE I. MSTP TRAFFIC MAPPING

Traffic Source	VLAN	Spanning Tree
CSsrc1	10	1
CSsrc2	40	4
CSsrc3	20	2

Dbsrc1	20	2
Dbsrc2	30	3
Dbsrc3	10	1

2) Load Imbalanced Scenarios

To evaluate load balancing, the MAN topology from Figure 3 was used but with two additional sources. CSsrc4 and DBsrc4 are added to **access3** switch and **access4** switch respectively, Hence, there are now eight traffic flows from CSsrc{1,2,3,4} and DBsrc{1, 2, 3,4} to DBserver, where each flow is a video conferencing session starting at 100s. The simulation runs for 240s. However, there are no link failures in this experiment.

The link capacities are shown in Figure 3. The reason that we choose 10Mbps for the links in the access network was for simulation efficiency. It is faster to overload the 10Mbps links, and there is still enough memory to run the simulation.

The traffic load is distributed evenly for MSTP, with each Spanning Tree having 2 sources. CSsrc1 and DBsrc3 traverse on ST1; DBsrc1 and CSsrc3 traverse on ST2; DBsrc2 and CSsrc4 traverse on ST3; and CSsrc2 and DBsrc4 traverse on ST4. In contrast, all of the traffic starts on the same initial ST for the STEP experiment.

B. Grid Topology

Similar to the MAN topology, resilience and load balance are evaluated for the grid topology. The resilience test in this topology is more rigorous in that it includes both node failures and link failures. Whenever there is a node failure, all the links attached to the node also fail as well.

1) Failures Scenarios

There are 4 flows to each of the 3 servers in Figure 4. Each flow is a video conference session starting at 100s. All links have capacity of 100Mbps. This capacity is enough to carry the traffic without causing any congestion. The simulation runs for a duration of 180s. There are a total of 26 failed links and 6 failed nodes. The link failures and node failures are scheduled as follows:

110s: node_7 fails
110s: node_8 <–> node_9 fails
120s: node_10 fails
130s: node_13 fails
140s: node_14 fails
140s: node_27 <–> node_28 fails
150s: node_16 fails
150: node_20 <–> node_21 fails
160s: node_23 fails
160s: node_32 <–> node_33 fails

In the MSTP scenarios, the flows are group by the destination to put into the corresponding tree. For example, since S1, S2, S3, and S4 are going to server1, they are transported on the same Spanning Tree. Again, all of the traffic starts on the same initial ST for the COST scenarios.

2) Load Imbalanced Scenarios

The load balancing experiments are similar to the configuration of the above resilience experiment, except that all links are now 10Mbps. The reason is due to simulation efficiency and resources. Since the bottleneck for the RSTP scenario is the links on the path from node_33 to node_21, for fairness, we upgrade these links to 100Mbps for all the three protocols. The simulation ran for a duration of 170s and no link failures were scheduled in this experiment.

V. ENHANCED ETHERNET RESILIENCE

As alluded to earlier, resilience is of particular importance for carriers and this is one area for which Ethernet is well recognized as being very weak. COST was specifically formulated to address the inherent weakness of Ethernet resilience. An experiment is presented in this section in which RSTP, MSTP and COST are evaluated for their resilience in the face of link failures and recoveries. The results of the MAN topology from Figure 3 are presented first and described in details to illustrate the behavior of COST. Then the results from the grid topology are overviewed to demonstrate the impact of COST on a denser topology.

A. Performance in Metro Area Network Topology

This subsection reports the performance of each protocol in the face of failures separately. Then a superimposed graph shows how they are stacked up against each other.

1) RSTP

The throughput for RSTP as observed by the receiving host during the link failures is depicted in Figure 5. As expected, when a link fails, RSTP re-converges and a dip in the throughput is witnessed before the new link assumes responsibility. Figure 5 shows the effect of failure in the network at different times. The first dip accounts for the failure at 120s. The opened gap takes 10s for RSTP to re-converge. Following the re-convergence, the link **aggregator1<–>core2** is unused, which explains why no dip is observed for the link failure at 140s. The second dip accounts for the link failure at 180s; while the third dip is the result of the link recovery at 220s. Similarly, the second re-convergence takes 10s and the third re-convergence takes 7s.

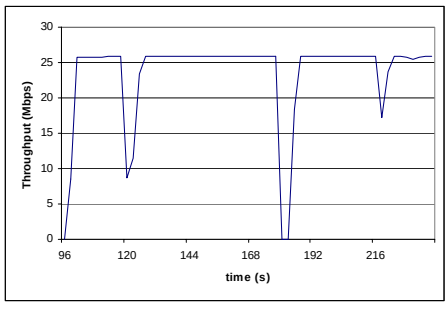

Figure 5 the throughput as observed by the receiving host during the link failures for RSTP

2) MSTP

Figure 6 illustrates the impact of failures in the MSTP network. The first dip accounts for the failure at 120s. The second dip occurs at 140s on account of **aggregator1 <–> core2** failure. Unlike RSTP, MSTP utilizes more links; therefore, it suffers a performance hit on the second failure.

However, the performance hits in RSTP are much more severe thanfor MSTP. It is confirmed in Figure 6 that the dip is not as deep as in Figure 5. The third and last dip are the result of **aggregator2 <–> core1** failure and the recovery of **aggregator1 <–> core1,** respectively. On average, each reconvergence takes 7 seconds.

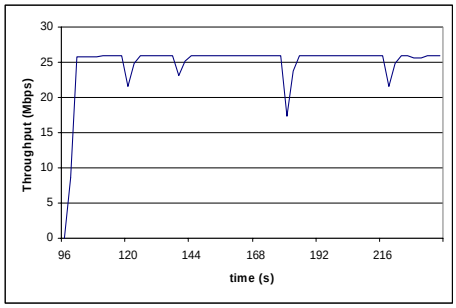

Figure 6 The throughput as observed by the receiving host during the link failures for MSTP

3) COST

The throughput on the intermediate links between the core switches and the aggregator switches can be seen in Figure 7. In this figure, the resolution on the time axis is lower to illustrate the handoff between the links of different Spanning Trees, as shown by the drops in the throughput where the next link assumes responsibility at the same time the previous link fails. These drops do not affect the overall throughput received by the end host. The uninterrupted service is evidenced in Figure 9. As prescribed by the monotonically increasing property of COST, the traffic is initially sent on vlan10 (**aggregator1**<–>**core1** link as shown in Figure 20). Following a failure, the link for vlan20 (**core2**<–>**aggregator1** link as shown in Figure 21) takes over; and when that link fails, the link for vlan30 (**aggregator1**<–>**aggregator2** link as shown in Figure 22) takes over. When **aggregator1**<–>**core1** recovers, the recently arrived frames do not have to crossover so that the rate of **aggregator1**<–>**core1** picks up again showing the dynamic nature of COST.

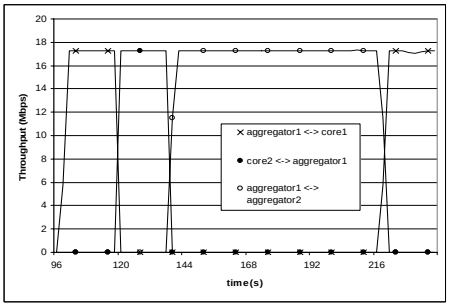

Figure 7 The throughput as observed on various links in the topology as links fail and COST re-routes traffic resiliently to maintain constant throughput

4) Comparison of RSTP, MSTP and COST

For this scenario, the comparative performance of RSTP, MSTP, and COST can be visualized by superimposing the cumulative throughput graphs of RSTP, MSTP, and COST as in Figure 9. While the maximum throughput is the same in this experiment, it can be seen that during the failure periods COST is able to maintain a sustained throughput.

Despite incurring one or more link failures, COST has been designed to minimize the frequent execution of the Spanning Tree re-convergence algorithm. As seen in Figure 8, the reconvergence time for COST is zero at each link failure or recovery event. If one or more failed links recover before COST exhausts the available VLANs (monotonic increase), then it is possible that no re-convergence is ever required despite links failing. Therefore, COST is able to operate continuously without interruption to the service. This is a clear advantage of COST over RSTP and MSTP. To measure the throughput graph, the lost percentage (the area of the dipped region) of RSTP and MSTP is compared against COST. The results show that RSTP loses 7.3% of the total received traffic compared to COST; and MSTP loses 1.69% of the total received traffic compared to COST.

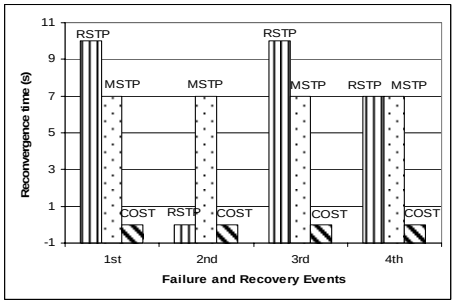

Figure 8 The reconvergence time for the 3 protocols at each event

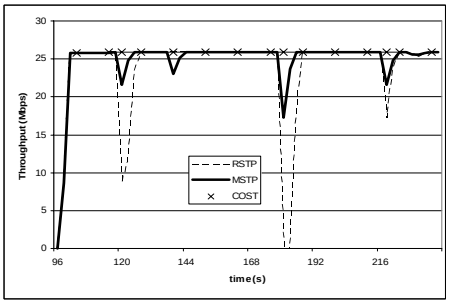

Figure 9 In contrast with RSTP and MSTP, COST maintained a constant throughput to the receiving host despite link failures and recoveries

B. Performance in a Grid Network

In this scenario, Figure 10 shows the cumulative throughput collected at the 3 servers. Both RSTP and MSTP begin to incur a performance hit after 130s, whereas STEP maintains constant throughput until 140s. In contrast to RSTP and MSTP, COST does not re-converge for every single failure or recovery. Therefore, after numerous failures without any recovery, a proportion of the traffic flows are transported on the last Spanning Tree. Evidence for this can be seen in Figure 12 where COST graph line does not return to the maximum rate. Until 145s, RSTP loses 7.69% of the total traffic compared with COST, and MSTP loses 1.69% of the total traffic compared with COST. After 160s, the majority of the traffic in the COST experiment is transported on the last Spanning Tree. Therefore the throughput for COST drops to 3.1MB/s. As

mentioned before, this situation can be detected and simply averted by triggering a reconvergence. However, if one or more of the links recovers, then re-convergence may not be necessary.

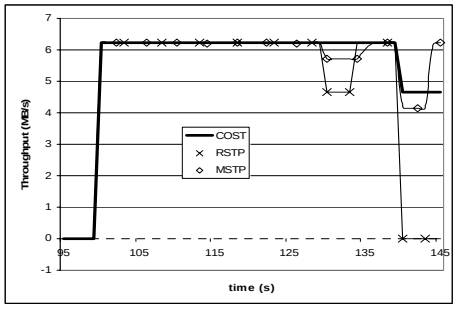

Figure 10 The cumulative throughput of illustes reconvergences for RSTP and MSTP while COST crossover some flows to the last ST.

VI. ETHERNET SWITCH LOAD BALANCING

By being able to distribute the traffic, or to load balance, across various links in a network, it is possible to increase the capacity and utility of the network. However, none of the existing Ethernet protocols allow the carrier to dictate load balancing across all the links in the network. The COST algorithm will facilitate load balancing across all links in the metro Ethernet. The carriers will thus have an option for balancing load across the network, as well as fine-grained control of the load on individual links. This will be an attractive feature for the carriers as they can exploit maximal throughput, and thereby capacity from the network. Similar to the resilience simulation, the results of topology from Figure 3 are presented first and in details showing the behavior of COST. Then the results from the grid topology will be shown from the overall perspective.

A. Performance in Metro Area Network

The load balancing performance for MAN topology is examined in this section starting with RSTP. In order to see the inefficiency in RSTP and MSTP, the utilization of the intermediate links between the core switches and the aggregator switches are shown. As can be appreciated in Figure 3, the "left side" of the network refers to the links connected to **aggregator1**; and the "right side" of the network refers to the links connected to **aggregator2.**

1) RSTP

In this section, the traffic on intermediate links in the network was measured to show the inefficient link utilization of RSTP due to the lack of load balancing mechanism. The metric that we use in the graphs for demonstrating the efficacy of the respective protocols is again the cumulative throughput.

At **aggregator1** switch, there are 3 links that can potentially carry the traffic into the core network. However, in order to prevent loops, RSTP blocks 2 of those links. As shown in Figure 11, the **aggregator1<–>core1** link is loaded to its maximum capacity. Despite approximately 25.8 Mbps arriving at **aggregator1**, the output from the switch was only 10 Mbps. RSTP cannot use the other 2 links to transport the remaining traffic as they are blocked which forces **aggregator1** to drop the remaining traffic. At **aggregator2** in Figure 12, 8.6Mbps arrives and since the **core1<–>aggregator2** has the link capacity; no frames are dropped. However, if an overload situation occurs, excess traffic will be dropped because the **aggregator2<–>core2** is blocked.

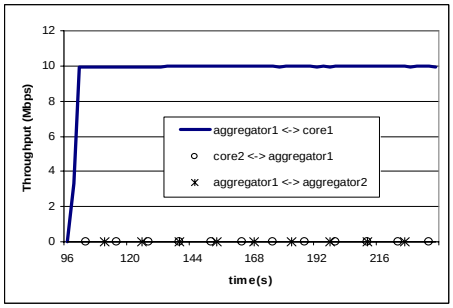

Figure 11 The throughput as observed on various links in the left side of the access network topology shows under utilized links

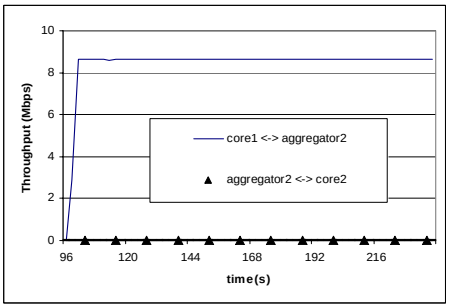

Figure 12 The throughput as observed on various links in the right side of the access network topology also shows under utilized links

2) MSTP

For MSTP, at **aggregator1**, 4.3 Mbps arrives for ST1, another 4.3Mbps comes from ST2, and 8.6Mbps is for ST3. Since ST3 root is at **core2** and the link **aggregator1<–>core2** blocks ST3, the 8.6Mbps must travel via **aggregator2**. At **aggregator2**, there are 4.3 Mbps for ST1, 4.3 Mbps for ST2, 8.6Mbps for ST3 that comes from **aggregator1**, and 8.6 Mbps for ST4. As ST2 and ST3 share the same link, the capacity on the link **aggregator2 <–> core2** is exhausted causing frames to be dropped. The link **aggregator2 <–> core1** only carries ST1 traffic which is 4.3Mbps, as shown in Figure 14. The traffic for ST4 is sent via **aggregator1** because the root for ST4 is at **core1** and only link **aggregator1 <–> aggregator2** allows traffic for ST4. The link **aggregator1 <–> core1** carries the combined traffic of ST1 and ST4 (arriving from **aggregator2**), thus is maxed out at 10Mbps. The link **aggregator1 <–> core2** only carries traffic for ST2, and hence uses only 4.3Mbps. The link **aggregator1 <–> aggregator2** directs ST3 traffic to **core2** as explained earlier, thus using 8.6Mbps. This behavior is captured in Figure 13.

Although this shows that it is possible to perform load balancing in MSTP, it is not efficient. Even if the access ports are reconfigured to distribute the load across the network, it applies only to a specific situation. Due to the dynamic and unpredictable nature of packet switched traffic, there is no

single static configuration that works for all. Unlike STEP, MSTP cannot be responsive to traffic conditions.

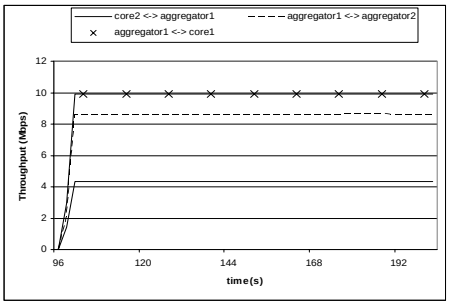

Figure 13 The throughput as observed on various links on the left side of the access network topology

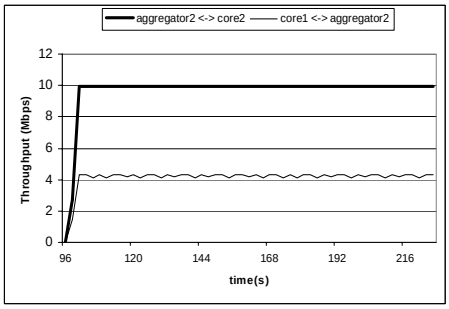

Figure 14 The throughput as observed on various links on the right side of the access network topology

3) COST

As with the RSTP experiment, the traffic on intermediate links in the network was measured to show the resulting link utilization. In this instance of the experiment, the link utilization threshold for load balancing was set at 80%. This means that for a link capacity of 10 Mbps the switch will permit at most 8 Mbps on that link before it will try to switch the traffic to the next ST unless it is the last ST to which the traffic can be crossed-over. In this case, the last ST is ST 4.

Initially, COST starts all traffic in ST 1 with each source sending 4.3Mbps. There are 24.6Mbps arriving at **aggregator1** on ST 1. The other 1.2 Mbps (0.6Mbps from **access3** and another 0.6Mbps from **access4**) is sent to **aggregator2** on ST2, because the load balance threshold for the link is 80%. The 1.2Mbps is now on ST2. In addition, there are 8.6Mbps arriving at **aggregator2** on ST1. Of the 24.6Mbps arrived at a**ggregator1,** 8Mbps is sent to **aggregator1 <–> core1** link on ST1, 8Mbps is sent to **aggregator1 <–> core2** link on ST2, and 8Mbps is sent on the **agrregator1 <–> aggregator2** on ST3 toward **aggregator2**. The remaining 0.6Mbps is elevated to ST4 and is sent via link **aggregator1 <–> core1**. Since the **aggregator1 <–> core1** link is shared by the last ST, it is allowed to transport more than the 80% threshold. On the right side, **aggregator2** sends 8Mbps to **aggregator2 <–> core1** link on ST1 and elevates the remaining 0.6Mbps to ST2. Now, aggregator2 sends the combined 0.6Mbps + 1.2Mbps on ST2 to **aggregator2 <–> core2** link. The 8Mbps arriving to **aggregator2** from **aggregator1** on ST3 needs to be sent out on link **aggregator2 <–> core2,** and this link is shared by ST2 and ST3. However, the **aggregator2 <–> core2** link is capped at 8Mbps, thus, 1.8Mbps traffic must be elevated to ST4 and sent back toward the root of ST4 at **core1**. Therefore, the link **aggregator1 <–> core1** receives additional traffic which totals 1.8+8+0.6 = 10.4Mbps. Since the link is only able to transport 10Mbps, some frames are dropped. The behavior is illustrated in Figure 15 and Figure 16.

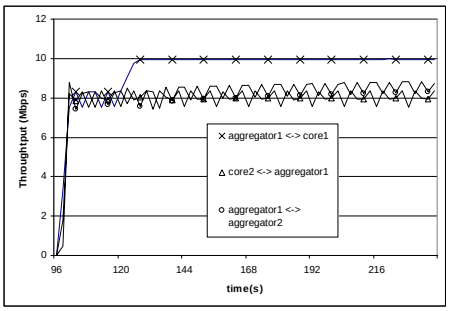

Figure 15 The throughput as observed on various links on the left side of the topology showing COST improving and balancing link utilization

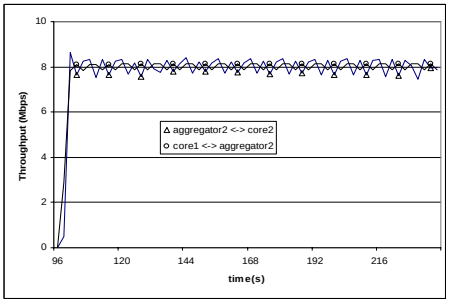

Figure 16 The throughput as observed on various links on the right side of the topology showing COST improving and balancing link utilization

4) Comparison of RSTP, MSTP, and COST

For this experiment, the comparative performance of RSTP, MSTP, and COST can be visualized by superimposing the cumulative throughput graphs of RSTP, MSTP, and COST as in Figure 17.

As the bottleneck for RSTP is at **aggregator1** where only 10Mbps can be sent out, by combining the traffic from **aggregator2** the end host receives approximately only 17 to 18Mbps. Although MSTP is able somewhat to redistribute the load, it is difficult to find an optimum assignment of the loads for a balanced network. Heuristic assignments provide limited assistance in this regard. By contrast, without any static pre-configuration COST dynamically redistributes the traffic if the current link is congested, thus able to accommodate flexibly increased incoming traffic. The fluctuation effect as observed in the throughput for COST in Figure 15 through Figure 18 is the result of COST stabilizing around the link utilization threshold. In our simulation, the current link utilization is measured once per second, and naturally these fluctuations can be smoothed further by selecting a smaller time period. Figure 17 shows the overlaid throughput of 3 protocols at the receiver. Comparing against the traffic throughput of COST, RSTP loses 37%; MSTP loses 12.76%.

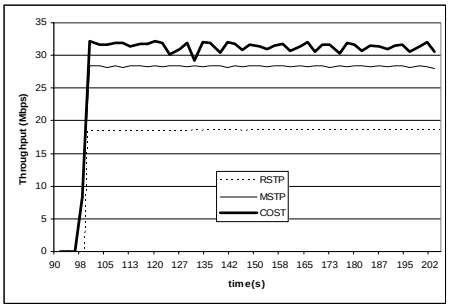

Figure 17 The cumulative throughput of RSTP, MSTP and COST for MAN topology

B. Performance in the Grid Topology

In this grid topology, there is heavy congestion that forces the switches to drop frames and, consequently, of the protocols achieved the maximum throughput which is at 6.45MB/s. Figure 18 shows that COST is able to achieve higher throughput than both RSTP and MSTP. COST delivers 8.8% and 9.2% more of the total traffic than RSTP and MSTP, respectively. It is interesting to note that in this case, the performance of RSTP and MSTP is almost equivalent. Even with multiple Spanning Trees, MSTP still drops the same amount of traffic as RSTP. This is due to inefficient construction of the Spanning Tree. There are works that address this inefficiency in Spanning Tree construction in [5][7][8][9].

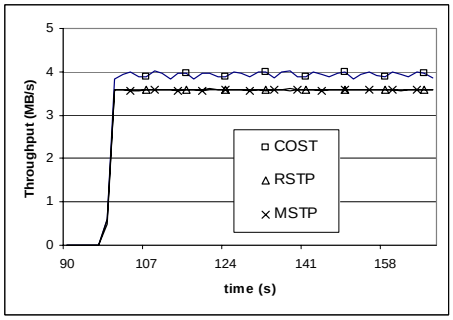

Figure 18 The cumulative throughput of RSTP, MSTP, and COST for grid topology

VII. Out of Order Problem

Since COST allows a flow to traverse multiple Spanning Tree simultaneously, it is conceivable that frames belonging to the same flow may take different paths. Therefore, the potential exists for end hosts to receive out-of-order frames. If such out of order frames force TCP retransmission, it has the potential to diminish TCP performance greatly. We have run a set of simulation to test the effect of out of order packet on RSTP, MSTP and COST. In this scenario, there are 6 sources uploading to a server. Each source uploads a single file of size 200MB. The topology is the one from Figure 3 with each node having 64K receiver buffer. Figure 19 shows that COST completes the file upload before both of RSTP and MSTP. This demonstrates that the out-of-order-frames issues with COST does not impact the performance of TCP.

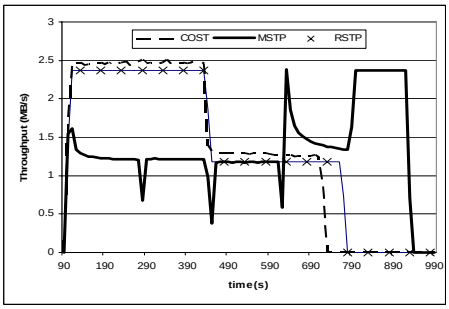

Figure 19 TCP throughput for RSTP, MSTP, and COST

VIII. Related Works

Viking is a Multiple Spanning Tree architecture proposed by Sharma et al [5]. Viking precomputes multiple Spanning Trees so that it can change to a backup ST in the event of a failure. The paths are computed based on the weight that is assigned to each link. The weight is derived from the criticality of the corresponding link. Viking's complexity lies in the computation of the *k* shortest primary paths and the *k* backup paths for each primary must be determined. A path aggregation algorithm is then run to merge the paths into the Spanning Tree. Viking uses a client-server model that needs to be informed by the end-hosts to update the server on the condition of network before the Spanning Trees are periodically recomputed.

Ethereal [6], a real time connection oriented architecture supporting best effort and assured service traffic at the link layer, proposes to use the propagation order Spanning Tree for faster re-convergence of the ST once a failure has been detected. SmartBridge [8] and STAR [9] are also two other approaches to improve upon the STP. They both find an alternate route that is shorter than the corresponding path on the Spanning Tree. SmartBridge requires the full knowledge of the topology. STAR is an overlay approach where STAR-aware switches are the super nodes of the topology. STAR calculates the shortest path from a super node to the next using the distance vector.

Lim et al. [10] address the underutilization of the standard Spanning Tree. They also recognize that the simple priority queuing of 802.1 potentially starves low priority traffic when the high priority traffic dominates a significant fraction of the traffic. Each multimedia traffic flow uses the Spanning Tree that is built for the tuple <traffic type, VLAN>. While non-multimedia traffic flows use the Spanning Tree that is built for a traffic type. Each flow stays in the designated ST and no crossing over is allowed.

Another approach to load balancing is Tree-Based Turn-Prohibition (TBTP) [7]. TBTP constructs a less restrictive Spanning Tree by blocking a small number of pairs of links around nodes, called turn, so that all cycles in a network can be broken. The benefit of TBTP is proportional to the degree of the nodes and MEN access networks have a low node degree. However, TBTP did not improve on the recovery time of the standard Spanning Tree protocol. Since TBTP relies on the standard STP to re-converge before it can re-compute its routing, the recovery time is in the magnitude of seconds.

IX. Conclusion

In this paper, we proposed a new concept of cross-over Spanning Trees (COST) for routing packets in the MEN. We have presented results from a preliminary study that demonstrates the potential benefits that can be offered to the carriers by COST. The implementation of COST has a low complexity overhead and can leverage MSTP support already commonly available in Ethernet chipsets. In this work we have focused primarily on the resiliency and load balancing aspects of COST. Results obtained through simulation experiments using OPNET revealed the following potentials of COST:

- COST is very resilient to failures. It requires no re-convergence in face of multiple link failures. The throughput provided by COST is much higher than that of RSTP and MSTP.
- It helps in seamless re-integration and usage when failed links recover, thereby it may never need to reconverge. The process thus increases the MTBF of switches.
- COST provides load balancing of Ethernet frames throughout the metro access network, which is a new Ethernet layer feature. This gives carriers control over Ethernet that they previously never had.

References

[1] IEEE Information technology - telecommunications and information exchange between systems - local and metropolitan area networks - common specifications. Part 3: Media Access Control (MAC) bridges, ISO/IEC 15802-3, ANSI/IEEE Std 802.1D, 1998.

[2] IEEE Standard for Local and Metropolitan Area Networks — Common specifications Part 3: Media Access Control (MAC) Bridges — Amendment 2: Rapid Reconfiguration Amendment to IEEE Std 802.1D, 1998 Edition. IEEE Std 802.1w-2001

[3] IEEE Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks. IEEE Std 802.1Q-1998

[4] IEEE Standards for Local and metropolitan area networks Virtual Bridged Local Area Networks — Amendment 3: Multiple Spanning Trees Amendment to IEEE Std 802.1Q™, 1998 Edition. IEEE Std 802.1s-2002

[5] S. Sharma, K. Gopalan, S. Nanda, T. Chiueh "Viking: A Multi-Spanning-Tree Ethernet Architecture for Metropolitan Area and Cluster Networks" Proceedings of IEEE INFOCOM 2004.

[6] S. Varadarajan, T. Chiueh "Automatic Fault Detection and Recovery in Real Time Switched Ethernet Networks" Proceedings of IEEE INFOCOM 1999.

[7] F. De Pellegrini, D. Starobinski, M. G. Karpovsky, and L. B. Levitin. "Scalable Cycle-Breaking Algorithms for Gigabit Ethernet Backbones" Proceedings IEEE INFOCOM 2004

[8] T. L. Rodeheffer, C. A. Thekkath, D. C. Anderson. "SmartBridge: A Scalable Bridge Architecture" Proceedings ACM SIGCOMM 2000

[9] K. Lui, W. C. Lee, K. Nahrstedt. "STAR: A Transparent Spanning Tree Bridge Protocol with Alternate Routing" ACM SIGCOMM Computer Communications Review Volume 32, Number 3: July 2002.

[10] Y. Lim, H. Yu, S. Das, S. S. Lee, M. Gerla "QoS-aware Multiple Spanning Tree Mechanism over a Bridged LAN Environment" Proceedings IEEE GLOBECOM 2003

[11] OPNET simulator, http://www.opnet.com

[12] MEF, "Metro Ethernet Networks – A Technical Overview" http://www.metroethernetforum.org

[13] IEEE Std 802.3z-1998, Gigabit Ethernet, http://www.ieee802.org/3/z/index.html

[14] Nortel Networks "Service Delivery Technologies for Metro Ethernet Networks" Nortel Networks Whitepaper Sept. 19 2003 http://www.nortel.com/solutions/optical/collateral/nn-105600-0919-03.pdf

[15] Riverstone Networks "Scalability of Ethernet Services Networks" http://www.riverstonenet.com/solutions/ethernet_scalability.shtml

[16] G. Holland. "Carrier Class Metro Networking: The High Availability Features of Riverstone's RS Metro Routers" Riverstone Networks White Paper #135.

Appendix

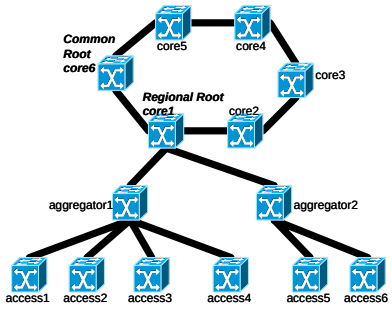

Figure 20 ST 1 configuration for MSTP and COST and the initial ST configuration for RSTP before any failure.

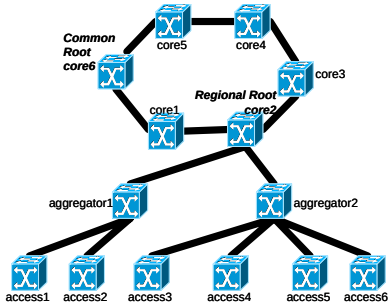

Figure 21 ST 2 configuration for MSTP and COST

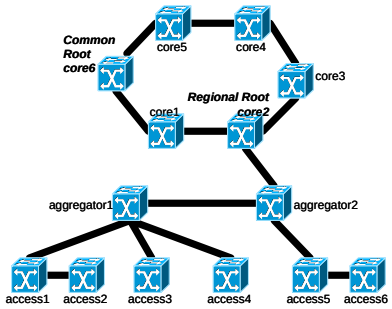

Figure 22 ST 3 configuration for MSTP and COST

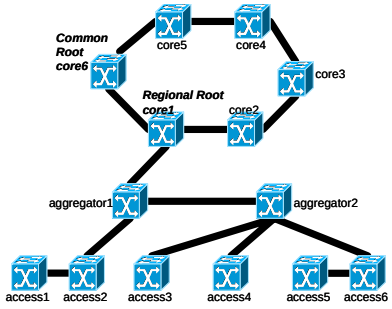

Figure 23 ST 4 configuration for MSTP and COST