Efficient Data Switching in Large Ethernet Networks using VLANs

Ray Liu

Master of Science in Internetworking 709 Project
University of Alberta

Abstract

Large data centers have been widely deployed to adapt to business requirements, and data centers have become very large facilities that contain more equipment like servers, storage units, and networking devices. But the network technology for these large data centers may be using inadequate topologies or protocols that gradually create a bottleneck on performance. In this paper we discuss recently developed technology improvements for data center networks, in particular, we discuss the use of Virtual Local Area Networks (VLANs) to improve performance.

1. Introduction

Traditional data centers are usually facilities used to house computer systems and associated components, such as networking devices and storage units. With the development of modern data centers, many cloud data centers now contain clusters of servers, storage units, configured so that the vast majority of traffic(80-90 percent in some cases) flows between adjacent servers (so-called east-west traffic)[1]. This is a very different traffic pattern from conventional data center networks, which supported higher levels of traffic between server racks (so-called north-south traffic).

There are many different network technologies that are currently in use, but Ethernet has been the most widely used network technology for traditional data centers due to the following advantages:

- All end-host devices have permanent and globally unique MAC address that are pre-configured.
- Ethernet switches are self-learning, which require no need for administrative configurations.
- Flat addressing supports host mobility, so host addresses do not need to be changed.
- Ethernet equipment is more cost-efficient.

Ethernet uses Spanning Tree Protocol as a layer 2 switching protocol used by classical Ethernet that ensures loop-free network topology by always creating a single path tree structure through the network [1]. Figure 1 is a typical data center topology [2]. At aggregation level, servers on each rack connected to Top of Rack (ToR) switch. ToR switch on each row connected to End of Row (EoR) switch on aggregation level, finally the EoR switch is connected to the core switch on Core level.

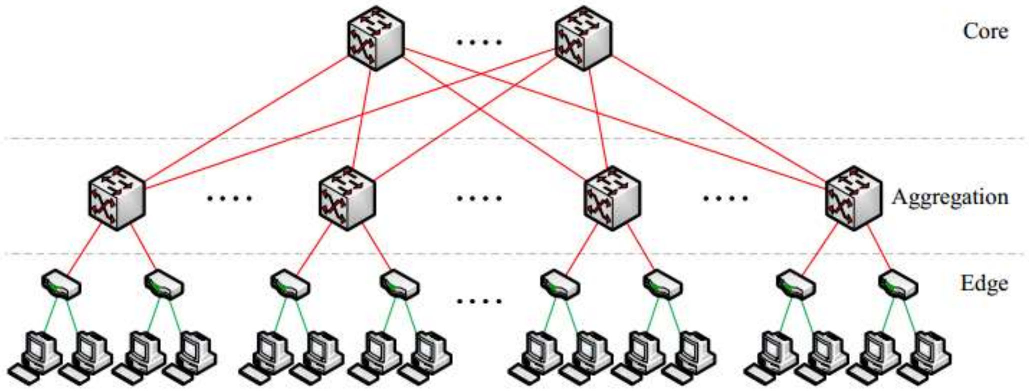

Figure 1. Common data center interconnect topology

In this topology, servers on different racks will need to communicate to each other by EoR switch even when they're close to each other, and servers on different rows will need to communicate through the Core switch. However, the Spanning Tree topology has many disadvantages [3]:

- Flooding-based delivery: Ethernet switch relies on flooding to deliver frames to unknown destinations. Flooding consumes excessive link bandwidth and leads to large forwarding tables in switches.
- Inefficient forwarding paths: Ethernet use Spanning Tree Protocol (STP), which was designed to create a loop free topology. STP will run an algorithm to trim the network by blocking extra pathways and eventually building a single tree topology.
- Broadcasting for basic service: Ethernet relies on broadcasting for address learning, with services such as Address Resolution Protocol (ARP) and Dynamic Host Configuration Protocol (DHCP). This consumes excessive resources, while introducing security vulnerabilities. Also, broadcasting will create broadcast storms which make Ethernet not a suitable choice for large networks.
- No load balancing and fault tolerant support: Each Ethernet node only has one parent node, so there would be only one active link forwarding packet. In the event of a link failure or reconfiguration, the network halts all traffic, and any change of topology would result in reconstruction of a new spanning tree which typically take 30-60 seconds.

Also, the overall cluster bandwidth is limited by the bandwidth available at the root of the hierarchy [2]. There are performance issues for traditional Ethernet switch using Spanning Tree Protocol as well. A switch could either maintain large forwarding tables to reduce the frequency of sending broadcast frames to a new host, or maintain a small forwarding table but increasing the frequency of sending broadcast frames. A large forwarding table will require powerful hardware, which increases processing delay, and small forwarding tables will result in more frequent broadcast frames. Building large networks will increase broadcast domain, thus worsening the problem. These disadvantages have limited Ethernet's scalability. Modern data centers usually have a large scale of networks, and high demand for performance, as well as reliability. Usually the requirements for large data center network contains:

- Easy configuration and management.
- Fault tolerance, reduced down time to a minimum.
- Distributed network traffic to reduce core switch workload.
- Per-port-cost effective.

In order to improve Ethernet scalability to adapt to modern data center requirements, several changes to Ethernet need to be made:

- Spanning Tree Protocol made a loop free network by blocking links that may cause loop in the network. In order to increase network throughout, all links need to be utilized, while using proper control to prevent loops within the network.
- Ethernet was not built to accommodate large networks. The non-hierarchical layer 2 MAC addressing makes forwarding tables very large and increases processing delay [4]. To solve this problem, it is necessary to reduce the size of the forwarding table.
- Broadcast storms create unnecessary network traffic and increases network traffic load. It is necessary to reduce broadcast domains and make spanning trees as small as possible. Broadcast frames like bootstrapping frames send by hosts and address learning frames send by switch. These are two main sources of broadcast. A proper way of handling these broadcast frames will largely reduce broadcast storm.
- Delay has been an increasing problem in larger networks. Switches equipped with larger memory tend to have longer queues for incoming packets, occupy switch resources, and increase end to end delay. The solution is to assign as much network resource as possible for each data transmission in order for data to be transmitted in the shortest time and minimize delay.

Several techniques have been developed to extend Ethernet functionality and accommodate data center network requirements.

2. Enabling technology

Ethernet relies on broadcasting for address learning, but broadcasting has largely affected Ethernet performance. As network scales increase, broadcast domains also increase. Broadcast domains need to remain small so less end-hosts will be affected by broadcast storms. VLAN (802.1q) was introduced to logically divide Ethernet into a smaller scale in order to reduce broadcast domains.

2.1 VLAN (IEEE 802.1Q)

VLANs address some of the problems of Ethernet and IP networks [5]. VLAN allows administrators to group multiple hosts sharing the same networking requirements into a single broadcast domain. By dividing a large bridged network into several appropriately-sized VLANs, administrators can reduce the broadcast and ensure isolation among different host groups. VLAN uses VLAN tags added in Ethernet frames to identify VLAN IDs, and only the frame that matches the VLAN ID will be allowed to pass to the end-host.

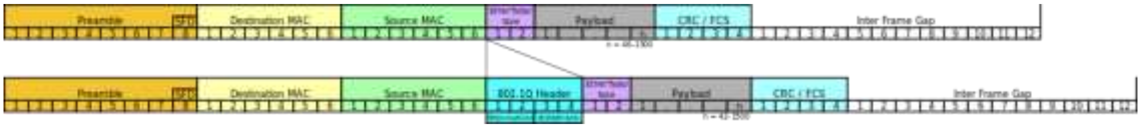

Figure 2. For Ethernet frames, VLAN adds a 32-bit field between the source MAC address and the Ether Type/Length fields of the original frame

VLAN also introduced problems:

- Trunk configuration overhead: Extending VLAN across multiple bridges requires the VLAN to be trunked at each participating switch port, which needs to be done manually by a network administrator.
- Limited control-plane scalability: Switch provisioned with multiple VLANs must maintain forwarding-table entries and process broadcast traffic for every active host in every VLAN visible to themselves.
- Insufficient data-plane efficiency: A single spanning tree is used in each VLAN to forward packets, which prevents certain links from being used. Effective use of per-VLAN trees requires periodically moving the roots and rebalancing the trees. Also inter-VLAN traffic must be routed via IP gateways, rather than shortest paths.

2.2 Hybrid IP/Ethernet architecture

One way of dealing with Ethernet's limited scalability is to build networks out of multiple LAN/VLAN interconnected by IP routing [5]. Each LAN/VLAN contains at most a few hundred hosts that collectively form an IP subnet, and communication across subnets is handled via default gateway. Each IP subnet is allocated an IP prefix, and each host in the subnet is then assigned an IP address from the subnet's prefix.

The advantage about IP network:

- Hierarchical network addressing and subnet-based routing, reducing address table size.
- Location addressing, no flooding for address lookup.
- TTL used in IP packet, allows temporary loops in topology.
- Efficient use of network topology, allows shortest-path forwarding over any topology, and also allows load-balancing.

This compromising solution also have disadvantages:

- Configuration overhead: Network addressing must be configured by administrator.
- Addressing inefficiency: Subnetting inevitably puts barriers between address blocks, making it difficult for administrator to design a network.
- Lack of mobility support: End-host needs to change its IP address when moving to a different link.
- Routing is directed to a link, not a node. Each link has its own block of address. Compare to CLNP, the bottom level of routing consists of routing to an area.
- Per-port-cost for routers is much higher than switch.

2.3 Multiple Spanning Tree

Per VLAN Spanning Tree (PVST) / Per VLAN Spanning Tree Plus (PVST+) / Rapid Per VLAN Spanning Tree (RPVST+) are Cisco's proprietary versions of Spanning Tree Protocol, PVST/PVST+ create a separate spanning tree for each VLAN; each spanning tree will have its own topology. Proper design of network would greatly improve network efficiency.

Multiple Spanning Tree Protocol (MSTP) was originally defined in IEEE 802.1s and later merged into IEEE 802.1Q-2005, defines an extension to RSTP to further develop the usefulness of virtual LANs (VLANs). MSTP allows formation of Multiple Spanning Tree (MST) regions that can run multiple MST instances (MSTI). Multiple regions and other STP bridges are interconnected using one single common spanning tree (CST), then map each VLAN to different MST instances to make efficient use of network topology. MSTP maintain separate trees for different broadcast domain, but forwarding path in each domain remain sub-optimal [3].

2.4 Routing in Layer 2

Transparent Interconnection of Lots of Links (TRILL) [6] is an Internet Engineering Task Force(IETF) protocol standard that uses Layer 3 routing techniques to create a large cloud of links that appear to IP nodes to be a single IP subnet. TRILL is a Layer 2.5 protocol, using IS-IS routing protocol between Rbridges to efficiently make full use of network topology, by adding TRILL header for RBridge incoming packet. TRILL header contain TTL to avoid infinite loops, and ingress/egress RBridge IDs to route packets between RBridges, which are totally transparent to Layer 2 and Layer 3.
The advantage about TRILL:

- Zero configuration required.
- Support Multipathing of multi-destination frames and ECMP (Equal Cost MultiPath) of unicast frames.
- Doesn't limit topology to spanning tree, mesh-like topology could be supported.

TRILL have inherited most virtue about Ethernet, but didn’t solve all problems, it didn’t eliminate Ethernet broadcast flooding while added additional network layer which increased network complexity and delay.

IEEE 802.1aq Shortest Path Bridging (SPB) [7] provides frame forwarding on the shortest path within a Shortest path tree (SPT) region of a network by using ISIS-SPB on all SPT bridges to control the forwarding paths. ISIS-SPB uses the standard IS-IS procedures to construct and update the link state database in each SPT bridge.

There are other technologies that introduced routing into layer 2, but Ethernet is still preferred for its simplicity and economies. In this paper, we focus on improving Ethernet scalability by using VLAN.

3. Related Work

The Viking system [8] use MSTP in conjunction with VLAN to maximize the overall throughput performance of the network by utilizing multiple redundant links, VLAN are used to select the desired switching path between a pair of end-hosts.

Viking relies on VLAN for selection of appropriate switching paths, by using tag based VLANs to select the desired switching path between a pair of end hosts. All paths which can possibly be used as switching paths are absorbed in different spanning trees. Since each spanning tree instance corresponds to a particular VLAN, explicit selection of a VLAN results in an implicit selection of the switching path associated with the corresponding spanning tree.

Viking system is based on client-server model, clients are called Viking Node Controllers (VNC) and the server is known as the Viking Manager (VM). Each end-hosts run a Viking Node Controller module which is responsible for load measurement and VLAN selection, Viking Manager is the central place responsible for traffic engineering and fault tolerance. VM also informs end-hosts about the VLAN information as and when required. VM has the global view of the network resource utilization, conjunction with network topology VM can identify critical portions of the network and carry out appropriate network tuning.

In the paper [4], the author tries to improve the control panel to eliminate broadcast. Using a thin control plane, the control plane could be divided into a decision plane and dissemination plane. The dissemination plane’s task is to gather information about network topology as well as link and host statuses, while decision planes use the above information to calculate forwarding tables and offer host MAC lookup. Using distributed control planes, bootstrapping broadcasts like ARP and DHCP could be solved by querying directory service at local bridge, which is replicated at all bridges.

The SEIZE (Scalable and efficient zero-configuration enterprise) architecture [3] is an alternative design of Ethernet bridging. It tries to overcome Ethernet's limitations in four ways: by avoiding flooding to unknown destinations, restraining broadcast for data-plane scalability, keeping forwarding tables small and by ensuring efficient forwarding paths. SEIZE architecture uses flat addressing and link-state routing, keeping flat addressing for end-hosts. Host locations are detected by an adjacent switch. Each switch will use a hash-based location management scheme to register the end-host.
SEIZE handles ARP and DHCP broadcast using unicast-based resolution, as well as supporting regular broadcast/multicast with a VLAN like scoping option. This architecture is similar to TRILL; both require more powerful switches that are able to run routing protocols, but does not reduce the size of broadcast domain. Compared to Viking system, a centralized controller provides better performance while reducing costs on each switch.

4. Proposed ideas

Our work is based on Viking system. We discussed the implementation of software to assign traffic demands between nodes in Ethernet network to VLAN. This implementation allows us to discover other kinds of network topology. In our system, VLAN is used to not only separate the network to make spanning trees smaller, separate but also to keep each connection separate from each other. In this case, spanning trees could be minimized to only two hosts, and no broadcast packet will be passed to irrelevant hosts. There will be a centralized controller like the Viking Manager which will calculate the best route, allocate resources for each connection, and achieve minimum end to end delay.

Viking Manager is responsible for registering new hosts, calculating the shortest paths, assigning VLAN ID, and the configuration of switch port. The network will run a single STP start from the Viking Manager to connect all switches together. Since Viking Manager have the information of the entire network, broadcast and bootstrapping frames like ARP and DHCP could be sent through the default STP to Viking Manager to avoid broadcast storms. Furthermore, broadcast frames and bootstrapping frames can be sent to the switch's parent host only, which points directly to Viking Manager.

In order to support dynamic creation and destruction of VLANs, the use of GARP VLAN Registration Protocol (GVRP) [9] allows VLAN-aware switches to automatically learn the mapping of VLANs to switch ports without having to individually configure each switch, and allows end stations to register their VLAN membership. In order to support multiple VLANs for a single host on a single switch port, we can either add VLAN tag on Ethernet frame at each host, or use MAC Address-Based VLAN Mapping [9] rule on the switch to dynamically identify the destination of frame and add appropriate VLAN tag.

When end hosts request to send data to another host in the network, the Viking Manager first checks the closest switch location and calculates the shortest path in the network according to the current network load. Next, it updates the network map, then checks the VLAN ID assigned on the switches, and finally determines the next available VLAN ID and configures this VLAN on the corresponding switch port. After VLAN ID has been configured on switch port, a dedicated route is constructed for this connection. The Viking Manager could further send forwarding table to each switch so switches won't need to flood frame to construct a forwarding table.

When a dedicated route is constructed, all links are marked with a higher cost. So calculating the route for next connection will tend to avoid the links that have already been taken.

For setting up a new connection in a crowded network, the choices are either to share certain links with other connections, or take a detour through links that have not been used, depending on the cost added when the link is taken. Higher cost means to use links that are currently not in use, and lower cost means to use shorter paths, reducing hop count. For example, as Figure 3 shows from node A to node B, a straight path will have less hop count, but might have more cost. An alternative path might have a few more hop counts, but link cost is much lower. The value that is marked on links represent the queuing delay and processing delay on a heavy loaded switch compare to a light loaded switch.

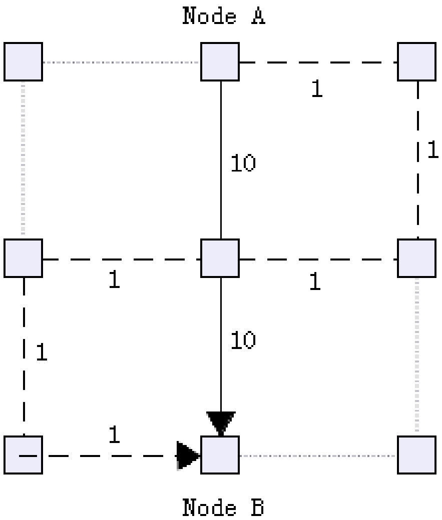

Figure 3. The number represent link cost.

This idea is similar to Multi-Protocol Label Switching (MPLS), instead of using labels, we use VLAN ID to identify each packet so we will not introduce a new protocol. The process of setup connection is also similar to circuit switching, which guarantees full bandwidth for each connection, as well as quality of service. There are disadvantages about circuit switching, such as the inefficient use of BW, but the use of Viking Node Controller is responsible for load measurement, so each link in the network will be used efficiently. Because each switch port was configured with VLAN ID, frames are filtered when passing through each switch port. In other words, data flow with specific VLAN ID will be trimmed all the way to destination, while no extra broadcast frames will be generated. Also the advantage about this design is a lowered performance requirement on each switch. If the connection only contains two hosts, it won't be necessary to check the forwarding table. If the connection contains more than two host, the forwarding table remains at a minimum.

The VLAN identifier space in 802.1q specifications is limited to 4096 entries. The maximum number of spanning trees supported by switches are far less. In this implementation, we try to separate each pair of connection to different switch, so the total number of VLAN that each switch will carry is minimum. Also, the same VLAN ID can be reused by a different switch so the number of connections supported is far more than 4096. A VLAN recycle mechanism should be used to recycle not in use VLAN IDs after a certain time to reduce switch workload. Recycled VLAN IDs should be reused after a certain time to make sure there's no data packet left in the link.

5. Simulation results

The simulation is based on grid topology to simulate a datacenter with N*N server rack, Spanning Tree topology is used to compare results.

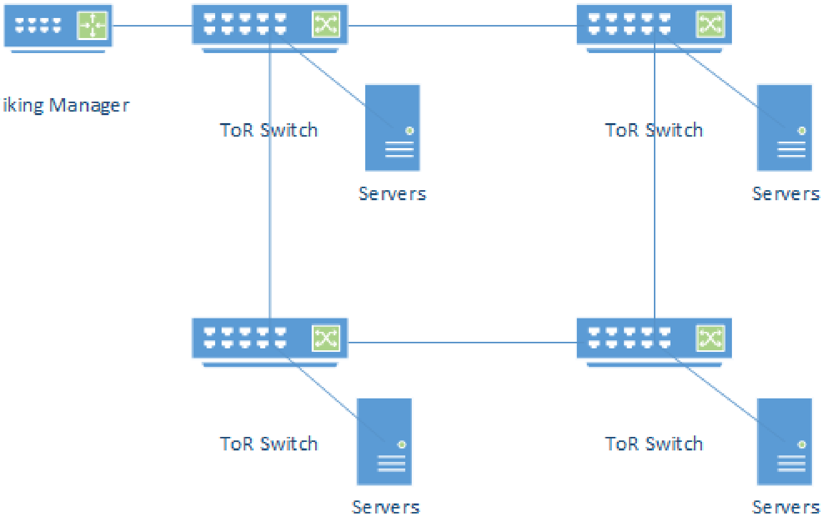

Figure 4. Viking System in Data center

In the topology network shown in Figure 4, each rack has a Top of Rack switch for all the servers. Instead of using end of rack switch and core switch, each ToR switch connects to up to four adjacent ToR switches. Assuming that each link have equal cost, and each connection uses the same amount of bandwidth, hosts in the network will start connecting randomly. The following steps are for a simulation:

1. Two different random numbers will be generated to represent the start point and end node of connection respectively, to simulate random connections in the network. Assume each connection takes 10% of link Bandwidth.
2. Use Dijkstra algorithm as shortest path algorithm to calculate the shortest path between the two nodes, and list all links that will be used by this connection.
3. Cost on all links will be added by 10% to represent traffic density. When cost has reached its maximum, no more data can pass through this link. The cost of the link will be marked as infinite.
4. Calculate the next connection.

The simulation will repeat until a new connection can't be established. The current amount of established connections is the maximum number of connections supported by this network without causing any congestions, which might result in queuing delay. In comparison, spanning tree topology is used to simulate traditional data center network.

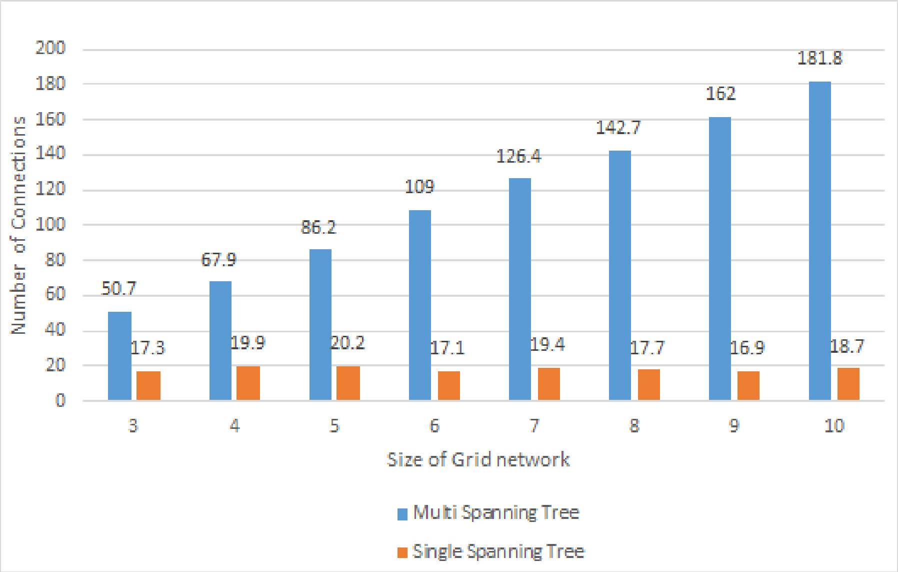

Figure 5. Simulation result

Figure 5 shows the simulation result. Since each pair of nodes are chosen randomly, the number given below is the average number of 10 simulations. From the simulation, we observe the topology using Single Spanning Tree have fixed number of connections due to the bandwidth limit at root. The number of connection supported by Multi Spanning Tree have a linear growth due to the increase of number of available links.

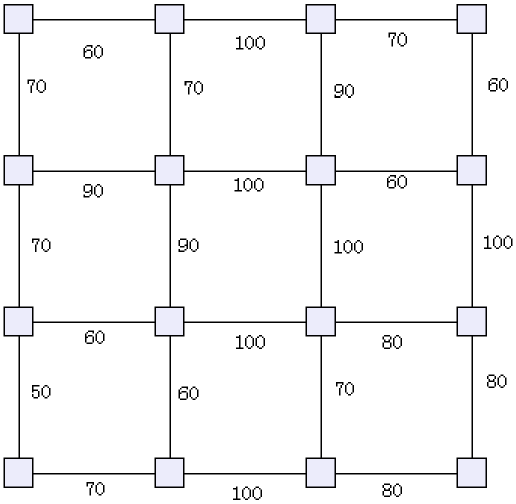

Figure 6. Link utilization on grid topology using Multi Spanning Tree

Figure 6 shows the link utilization in percentage after simulation. In this topology all links are utilized, links in the middle of the topology are used heavily because there are more neighbors nearby, links on the edge with less neighbors are used less heavily. Compare to Spanning Tree topology, links on the root are totally saturated, while links on the leaf are still available.

6. Conclusion

The simulation shows the efficient use of links in the network to achieve higher bandwidth. We are able to increase network bandwidth to increase the number of links. For example, by connecting the edge of a grid network together, we created a grid topology without edge. This topology shown in Figure 7 have a much higher number of connections (average 131.6 connections for 4*4 grid, almost double the number of connections). In this topology, network traffic is evenly distributed.

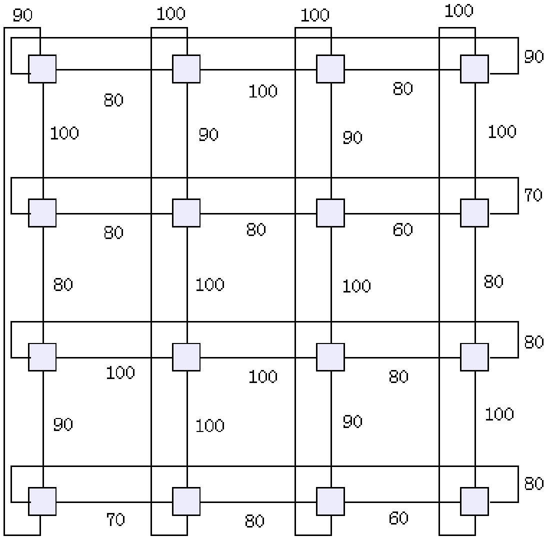

Figure 7, link utilization on grid topology without edge

Traditional Spanning Tree protocol blocks links in the network to avoid loops; therefore utilization efficiency has been limited. The Viking System provides a way to allow all links in the system to be fully utilized without creating loop and broadcast storm. It also allows a larger network with higher bandwidth to be built.

Using centralized controller like Viking Manager to establish connections between two hosts can be used for both layer 2 and layer 3, since Viking Manager have the location of all hosts that registered on it. Also, the use of Dijkstra algorithm to calculate shortest path for each connection is a typical behavior of routing protocol; therefore it is possible to combine layer 2 and layer 3 together within a Viking network.

Ethernet has had a high growth in capacity during the past decades, but the actual response time is decreasing with the increase of more and more layers. Each additional network layer added introduced more functionality but required additional header which lowered network throughput. For example, on the latest 2.0 Gbit/s links, Myrinet often runs at 1.98 Gbit/s of sustained throughput, considerably better than what Ethernet offers, which varies from 0.6 to 1.9 Gbit/s, depending on load [10]. Our future work will focus on combining network layers to reduce frame overhead, in order to reduce delay and improve over all throughput.

Bibliography

[1] Carolyn J. Sher DeCusatis, Aparico Carranza, Casimer M. DeCusatis “Communication within Clouds: Open Standards and Proprietary Protocols for Data Center Networking”, *IEEE Communications Magazine September 2012*

[2] Mohammad Al-Fares, Alexander Loukissas, Amin Vahdat, “A Scalable, Commodity Data Center Network Architecture”, *ACM 2008*

[3] Changhoon Kim, Jenifer Rexford “Revisiting Ethernet: Plug-and-play made scalable and efficient”

[4] Andy Myers, T.S. Eugene Ng, Hui Zhang, “Rethinking the Service Model: Scaling Ethernet to a Million Nodes”

[5] Changhoon Kim, Matthew Caesar, Jennifer Rexford, “Floodless in SEATTLE: A Scalable Ethernet Architecture for large Enterprises”, *SIGCOMM’08, August 17-22, Seattle*

[6] Radia Perlman, “Challenges and Opportunities in the Design of TRILL: a Routed layer 2 Technology”

[7] David Allen, Peter Ashwood-Smith, Nigel Bragg, Janos Farkas, Don Fedyk, Michel Ouellete, Mick Seaman, Paul Unbehagen, “Shortest Path Bridging: Efficient Control of Larger Ethernet Networks”, 2010 IEEE

[8] Srikant Sharma, Kartik Gopalan, Susanta nanda, Tzi-cher Chiueh, “Viking: A Multi-Spanning-Tree Ethernet Architecture for Metropolitan Area and Cluster Networks”, *IEEE INFOCOM 2004*

[9] Rich Seifert, Jim Edwards, "The All-New Switch Book: The Complete Guide to LAN Switching Technology Second Edition", 2008

[10] Wikipedia, en.wikipedia.org/wiki/Myrinet