IMPROVING THE PERFORMANCE AND SCALABILITY OF THREE-TIER DATA CENTER NETWORK ARCHITECTURE

A Thesis Presented

by

Yohannes Maru

to

The Faculty of Informatics

of

St. Mary's University

In Partial Fulfillment of the Requirements

for the Degree of Master of Science

in

Computer Science

February 22, 2020

ACCEPTANCE

Improving the Performance and Scalability of Three-tier Data Center Network Architecture

By

Yohannes Maru

Accepted by the Faculty of Informatics, St. Mary's University, in partial fulfillment of the requirements for the degree of Master of Science in Computer Science

Thesis Examination Committee:

____

Internal Examiner

____

External Examiner

____

Dean, Faculty of Informatics

February 10, 2020

DECLARATION

I, the undersigned, declare that this thesis work is my original work, has not been presented for a degree in this or any other universities, and all sources of materials used for the thesis work have been duly acknowledged.

____

Full Name of Student

____

Signature

Addis Ababa

Ethiopia

This thesis has been submitted for examination with my approval as advisor.

____

Full Name of Advisor

____

Signature

Addis Ababa

Ethiopia

February 10, 2020

Table of Contents

List of Acronyms

CapEx	Capital Expenditure
DCN	Data Center Network
ECMP	Equal Cost Multi Path
Gbps	Giga bits per second
IoT	Internet of Things
IPv4	Internet Protocol Version 4
IT	Information Technology
L2	Layer 2
L3	Layer 3
LAN	Local Area Network
MAC	Media Access Control Address
Mbps	Megabits per second
NIC	Network Interface Card
NS-3	Network Simulator 3
OpEx	Operational Expenditure
OSR	Oversubscription Ratio
Pod	Point of Delivery
SDN	Software Defined Networking
ToR	Top of Rack
UTP	Unshielded Twisted Pair

List of Figures

List of Tables

Abstract

The exponential growth of data science, cloud computing, distributed systems, ubiquitous computing, Internet of Things and other emerging applications and services require mega data centers to house hundreds and thousands of computing devices used to store, manipulate, analyze data and disseminate information. These massive number of computing devices are interconnected together through the data center network. The data center network is the core of the data center and it plays a pivotal role in the data center as it interconnects the whole computing devices together and basically the data center network was deploying in Three-tier architecture. The Three-tier data center network architecture is typical designed in a hierarchical tree-based structure which has three layers i.e. core, aggregation and access layers. This architecture utilizes high-end enterprise-class network devices at the core and aggregation layer. The expensive and power consumption characteristics of this high-end components are the challenges while deploying Three-tier architecture. In response to this many research efforts were done, and different types of data center network architectures were proposed. However, these proposed architectures designed to address one or two data center requirements but introduces other challenges.

Some of the proposed architectures are cost effective and less energy efficient since they use low-cost commodity off-the-shelf network devices, but they suffer incremental expansion, cabling complexity, management and operational difficulties. Some of the architectures can support large number of servers but produces poor performance network and can be complex to manage and maintain. Even if many researchers propose different data center network architectures, the Three-tier data center network architecture becomes the de-facto and most promising architecture deployed in the data center currently by data center operators. Since the cost of high-end devices used on the core and aggregation layer is one-time cost because the end-of-life of such devices is long term and also these devices replaced with low power usage chassis-based devices, still the Three-tier data center network architecture inherits low network performance specially for inter-rack traffic flows (the traffic generated by nodes in one rack or pod and flows to nodes which are installed in other rack or pod).

There are many causes that affect network performance, but the main cause of poor network performance is the oversubscription ratio. The oversubscription ratio is the ratio of downlink bandwidth to uplink bandwidth and 1:1 oversubscription ratio means all servers can communicate with their full provisioned bandwidth. In fact, this ratio can be achieved at the access layer where intra-rack traffic flow takes place i.e. the traffic stays within the rack, but moving up, to the upper layers achieving 1:1 oversubscription ratio is impossible. In this paper, the Three-tier data center network architecture is simulated using an NS-3 network simulator by changing the network structure, increasing the number of nodes and the traffic flows generated by each server in order to observe the effect of oversubscription ratio on average packet delay and throughput. This study also discusses some major network components which highly affects the performance of Three-tier data center network architecture and proposes basic solution for these issues.

Key words: Data center network, Performance, Oversubscription ratio, Packet delay, Throughput

CHAPTER ONE: INTRODUCTION

1.1 Background

Data center is a physical facility used to house computing equipment (server, storage, router, switch, firewall, load-balancer, power distribution equipment, cooling and humidification system, lighting system, and other related equipment etc.) of enterprise mission-critical and business-critical system for storing, manipulating and distributing large amounts of data. An enterprise business mainly depends on the business-application and services deployed in the data center, which makes data centers are typically a key enterprise business parameter and core asset of IT infrastructure to facilitate day-to-day business operations efficiently. All these infrastructures within a data center are orchestrated by the data center network to work as an organic cohesive whole [1].

Data center network (DCN) is an interconnection of different types of network-based components through a cable within a data center facility, so that they can communicate and distribute data between each other and to the external network outside the data center facility. The data center network architecture is a general framework which serves as a blueprint for the established network in the data center. The architecture contains the type of hardware devices deployed, the type of cables used for interconnecting those hardware devices, the physical and logical network topologies and the location where each hardware components are physically placed.

The Data center network (DCN) architecture is a core resource of an enterprise IT architecture to have an efficient information exchange, so proper design and implementation of DCN architecture requires special focus and scalability, resiliency and network performance metrics must be considered carefully. In addition to this the need for a robust data center that is fault tolerant can never be overemphasized, especially nowadays the advent of big data traffic, Internet of Things and other on-demand internet applications are on the increase [2].

In this inter-networked world, more than hundreds and thousands of terabyte data is generated daily and, in the future, much more data will be generated, and data centers will continue to play a vital role in the storing, processing, manipulating and distributing of data. Data centers are the backbone of large IT enterprise and telecommunication industries and they built big data center to provide different business services and the deployment trend shows that the number of computing devices installed in data center are continuously growing, so the network infrastructure should be a first order design concern for data centers.

Today the Internet has become ubiquitous and we are entering an era of the Internet of Things. The Internet of Things (IoT) is defined as a paradigm in which objects equipped with sensors, actuators, and processors communicate with each other to serve a meaningful purpose [7]. This new technology innovation i.e. Internet of Things (IoT) has a potential effect on the data center network design and deployment because the Internet of Things (IoT) will require a network that can handle increased demand of big data manipulation and analysis. Therefore, the data center network (DCN) is becoming a very important component for the deployment of Internet of Things (IoT). So, IT enterprises and telecommunication industries must give a critical attention for their data center network.

As the size of the network increases, lowering the cost of the overall system infrastructure and achieving higher level of performance have become first order concerns for data center operators [4]. In addition to achieving high level performance, data center operators must consider for scaling critical infrastructure components such as power distribution systems, cooling systems and the space to accommodate the exponential growth of data center components. Scalability of the data center network is the ability to construct and expand a network with simple, repeatable designs that can accommodate increased traffic or new devices without impacting applications, workflows or the cost per port.

1.2 Statement of the problem

As the emerging Internet technologies rapidly increase and as computing has become more pervasive, the demand for digital data has growing rapidly and the volume of data becomes huge. To keep up with this Internet technologies, IT industries and data center operators expand their data center network by continuously installing different computing devices in the data center. As a result, the data center network become large, more complex and the performance and scalability of the data center network continuously decrease. This poor performance and low scalable data center networks don’t provide high network convergence, fast route filtering and packet forwarding, efficient bandwidth utilization, high service availability, easy management and maintenance which leads to have high packet delay and low network throughput, finally affects the revenue of the IT industries and data center operators.

A data center hosts these emerging Internet technologies and house massive amount of computing devices that are interconnected through a communication network called data center network (DCN). The Data Center Network (DCN) lies at the core of a data center as it connects a large number of servers, storages, load-balancers, firewalls and other computing devices in various hierarchies through network switches. It is also a key enabler for the rapidly growing demand of network services and big data analysis. The data center network is characterized by the network architecture it employs and the data center network architecture is a general framework which serves as a blueprint for the established network in the data center.

Basically, the data center network is designed and deployed using the Three-tier DCN architecture in which network devices installed in three layers i.e. core, aggregation and access layer. In the core layer routers and layer 3 switches are used, in the aggregation layer load-balancers, firewalls and other security devices are installed and servers and storages are installed in the ToR (Top of Rack) or access layer. The Three-tier DCN architecture employs high-end enterprise-class network devices at the core and aggregation layer which are more power consumption and expensive. These challenges become bottlenecks for data center operators to design and deploy efficient data center network and in response to these challenges, researchers propose different types of DCN architecture.

Some of the proposed DCN architectures are cost effective and less energy efficient since they use low-cost commodity off-the-shelf network switches to build the data center network, but they suffer incremental expansion, cabling complexity, management and operational difficulties and some of the architectures can support large number of servers but produces low performance network and can be complex to manage and maintain [29] [59] [60]. Due to such and other related challenges these new proposed DCN architectures remain in the simulation stages and doesn't deployed in real the data center environments. This leads the Three-tier DCN architecture as the de-facto standard and most promising architecture deployed in the data center currently.

Since the cost of high-end network devices used at the core and aggregation layer is one-time cost because the end-of-life of such devices is long term and these devices come with low power usage chassis-based devices, still the Three-tier DCN architecture suffers with low network scalability and poor performance specially for inter-rack and inter-pod traffic flows. There are many causes that affect Three-tier DCN performance such as inappropriate use of cable standards, low processing capabilities of devices, but the main cause of poor performance is undefined oversubscription ratio caused by improper topology design.

In the Three-tier DCN architecture, oversubscription ratio is typically defined as the ratio of bandwidth for downlinks interfaces to the bandwidth for uplink interfaces. Network scalability is the ability of the network to be able to scale or enlarge to accommodate the network for future growth and to handle the growing amount of work in a capable manner. Oversubscription ratio is also the major factor that influences data center network scalability[4] because high oversubscription ratio at the access layer and aggregation layer will be a bottleneck to add and install additional computing device on the pod/modules.

Accordingly, the research attempts to answer the following research questions:

- What are the main performance metrics of data center network architecture and what is the main cause of poor performance Three-tier data center network architecture?
- How to increase the performance of Three-tier data center network architecture and efficiently interconnect the newly added computing devices in the data center to handle the increasing demand of new applications?

1.3 Objectives

1.3.1 General Objective

The main objective of this study is to improve the performance and scalability of the Three-tier data center network architecture by evaluating the architecture using different performance metrics.

1.3.2 Specific Objectives

The specific objectives of this study are: -

- To understand the state-of-the-art data center network architecture
- To investigate the performance metrics of data center network architecture
- To examine the performance of Three-tier architecture using different metrics
- To identify the major causes for low performance data center network
- To analyze and provide conclusion on the results obtained from the simulation
- To identify and recommend future research directions for further investigation on improving the performance of Three-tier architecture

1.4 Methodology

To achieve the objective and address the research questions, the following pathways has been used.

1.4.1 Literature Review

Different research documents and white papers were reviewed from International Electrical Electronics Engineering (IEEE), Association for Computing Machinery (ACM) and vendors to understand the state-of-the-art data center network and the performance metrics, challenges, and other concepts.

1.4.2 Simulation

In this study network simulator 3 (NS-3) an open source discreet event network simulator that is installed on HP-640 core-i7 with 500GB hard disk and 4GB RAM computer which has Ubuntu 18.04 operating system.

1.4.3 Analysis

After the Three-tier DCN architecture is simulated, the results obtained from the simulator are analyzed and reported. The main performance metrics that are used to evaluate the Three-tier DCN architecture (average packet delay and throughput) are also discussed.

1.5 Expected Results and Contributions

This study allows to have a basic understanding on data center network architectures and the challenges of current DCN architectures faced. The effect of oversubscription ratio on the performance and scalability of the Three-tier DCN architecture is observed from the simulation results and this study gives a high value how data center operators are beneficial in reducing service interruptions caused by packet drops, packet delays and maximizing network throughput by building a scalable and high-performance data center network. In addition to this, this study will be used as a baseline for future researchers which will study other aspects of data center networks.

1.6 Limitations and Scope

This study mainly aims to show how the oversubscription ratio affects the performance of the Three-tier DCN architecture, specifically the average packet delay and throughput. This study only focuses on the traffic within the data center i.e. the inter-rack and inter-pod/module traffic flow to evaluation and analyze the average packet delay and throughput of the Three-tier architecture. This study doesn’t focus on other types of data center network architectures and doesn’t focus on the network the network convergency and routing scalability, the performance of power consumption and cooling systems, security enhancement techniques, space utilization and other characteristics of the data center network.

CHAPTER TWO: LITERATURE REVIEW

2.1 Overview of Data Center Network

Data center is a physical centralized repository intended for housing computing equipment (server, storage, router, switch, firewall, load-balancer, power distribution equipment, cooling and humidification system, lighting system, and other related equipment) of enterprise mission-critical and business-critical system for storing, manipulating and distributing large amounts of data as per their information technology (IT) needs. According to [16], data centers are a closets, rooms or sometimes entire buildings of storage for a company's processing, distribution of large amounts of data, and server information. With the continuous growth of data volumes and variety of Internet applications, data centers have recently received significant attention as a cost-effective infrastructure for storing large volumes of data and hosting large-scale service applications [15]. For handling the extreme amount of data traffic, data center is a key to an organization. Google handles trillions of data with high traffic and an efficient data center keeps Google up all running [16].

Cloud applications such as Gmail, Google search, Amazon, Facebook have become an important part of our day-to-day activities. Large-scale Data Centers consist of tens of thousands of networked computers that provide services to cloud applications, have become an efficient and promising infrastructure for supporting data storage, band providing the platform for the deployment of diversified network services and applications (e.g., video streaming, cloud computing) [15].

The growth of Internet of Things (IoT) applications, such as smart cars, smart cities, and connected health devices, will also expand data center demands. Emerging applications for data centers like cloud computing, high performance computing and Big data are based on the cooperation of thousands of computing devices which are connected through a data center network. All these applications require the computing devices of the data center to exchange huge amounts of data between them [3]. A data center network is the communication infrastructure used in a data center, and is described by the network topology, routing/switching equipment, and the used protocols (e.g., Ethernet and IP) [15].

The data center network (DCN) holds a pivotal role in a data center, as it interconnects all the entire network-based devices and equipment together within the data center facility. Today's Large data centers are built to provide different services and the deployment trend shows that the number of servers in data centers continues to grow, so the network infrastructure should be a first order design concern for IT enterprises. The data center network which interconnects each computing device in the data center plays a key role in the scalability and performance of the services it runs [1]. To this end, the quest for a scalable and efficient data center network architecture has seen much recent progress [9]. So, future data center networks must be designed and deployed to deliver higher levels of performance, scalability, and availability to meet service-level agreements and maintain continuity of operations.

2.2 Data Center Network Architecture

Network architecture refers to the physical or logical layout of an enterprise's computer network. It describes the way different computer nodes are placed and interconnected with each other. The data center network (DCN) architecture is a blueprint or complete framework which provides a full information for the designed and deployed network. It defines the way in which network devices are interconnected, forming a pathway that hosts follow to communicate each other within the data center.

The data center network (DCN) architecture is a major part of a data center design, acting as a communication backbone, and therefore requires extreme consideration, since it has a great impact on the general data center scalability, performance and throughput [5]. Recent years have witnessed tremendous growth in the number of smart devices, wireless technologies, and sensors. In the foreseeable future, it is expected that trillions of devices will be connected to the Internet. Thus, to accommodate such a voluminous number of devices, scalable, flexible, inter-operable, energy-efficient, and secure network architectures are required [10].

Indeed, Large-scale data centers form the core infrastructure support for the ever-expanding cloud-based services. Thus, the performance and dependability characteristics of data centers will have significant impact on the scalability of these services. Generally, the data center network needs to be agile and reconfigurable in order to respond quickly to ever changing application demands and service requirements.

Advances in data intensive computing and high-performance computing facilitate rapid scaling of data center networks, resulting in a growing body of research exploring new network architectures that enhance scalability, cost effectiveness and performance [4].

To overcome the challenges and problems faced by existing data center networks, Significant research work has been done on designing the data center network topologies in order to improve the performance of data centers and different types of DCN architectures are proposed. The emerging challenges, like simplicity, efficiency and agility, and the new optical-empowered technologies are driving the innovation of the new networking architectures in the data center [17]. According to [18], the following graph shows that some of the data center network (DCN) architectures that have been proposed over the time.

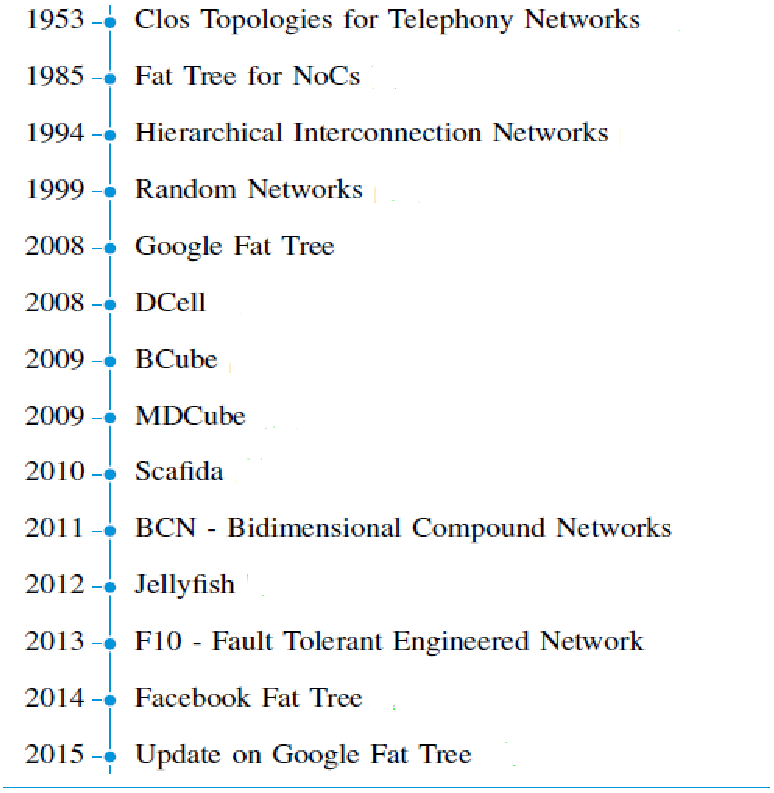

1953 Clos Topologies for Telephony Networks
1985 Fat Tree for NoCs
1994 Hierarchical Interconnection Networks
1999 Random Networks
2008 Google Fat Tree
2008 DCell
2009 BCube
2009 MDCube
2010 Scafida
2011 BCN - Bidimensional Compound Networks
2012 Jellyfish
2013 F10 - Fault Tolerant Engineered Network
2014 Facebook Fat Tree
2015 Update on Google Fat Tree

Figure 2- 1: DCN Architecture Evolution [18]

The data center network that interconnects the network-based devices has a significant impact on the performance, scalability, operation & maintenance and reconfigurability of the data center infrastructure to respond to changing application demands and service requirements.

According to the reconfigurability of the data center network architecture after the deployment of the network, DCN architectures are classified in to two, fixed architectures and flexible architectures [1] [6] [19]. The reconfigurability and network topology of a fixed data center architecture is not changed after it is deployed. However, in flexible data center network architecture the deployed topology will be changed anytime if any updates are necessary.

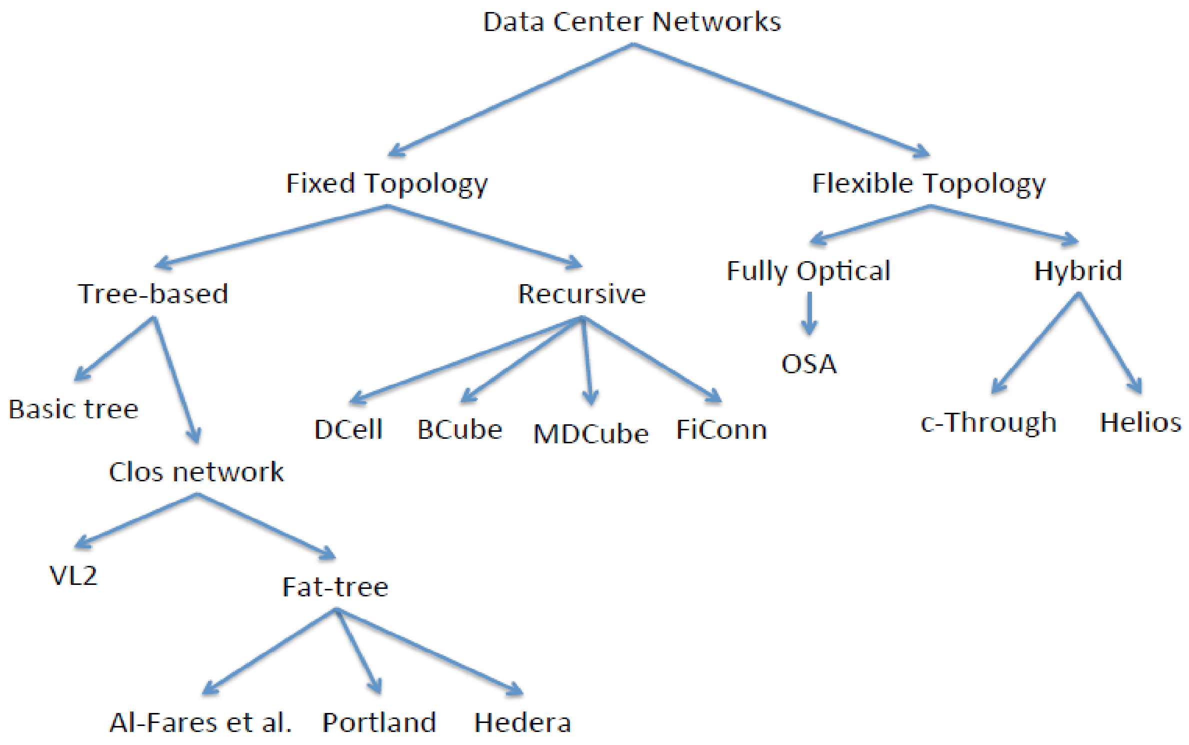


Figure 2- 2: DCN architecture classification [19]

Data center network architectures are categorized in to three classes based on the hardware equipment used to construct the topology and packet forwarding. These classes are switch-based/switch-only, server-based or server-only and hybrid architectures. In switch-based or switch-only data center network architectures, packet forwarding is implemented using only network switches. The well-known data center network architectures which are constructed as tree topology such as Three-tier, Fat-tree, VL2 and Flattened-butterfly are some examples of switch-only data center network architectures.

In server-based or server-only data center network architectures, packet forwarding is implemented using servers. Server-only architectures doesn't rely on switches to construct the network architecture and for network traffic processing rather they server-to-server links. CameCube is an example of a server-only data center architecture. The third category is hybrid-architecture and in this type of DCN architecture both switches and servers are utilized for packet forwarding. DCell and BCube are some examples of hybrid architectures [4] [11] [12].

2.2.1 Multi-tier DCN architecture

The most known and widely deployed multi-tiered data center network architecture is the Three-tier DCN architecture. Three-tier data center network architecture is a hierarchical tree-based structure and in the Three-tier DCN architecture, the switches are primarily arranged in three layers namely, core layer, aggregation layer and access layer. The core layer makes the foundation of the network tree, and each core layer switch is connected successively to all the aggregate layer switches. The aggregate layer switches interconnect multiple access layer switches together and the access layer switches used to connect a pool of servers and storages. High-end enterprise-class switches that can process and forward packets at high speed are usually used at aggregation and core layers, this makes Three-tier DCN architecture an excessively expensive and power-hungry architecture [4] [11].

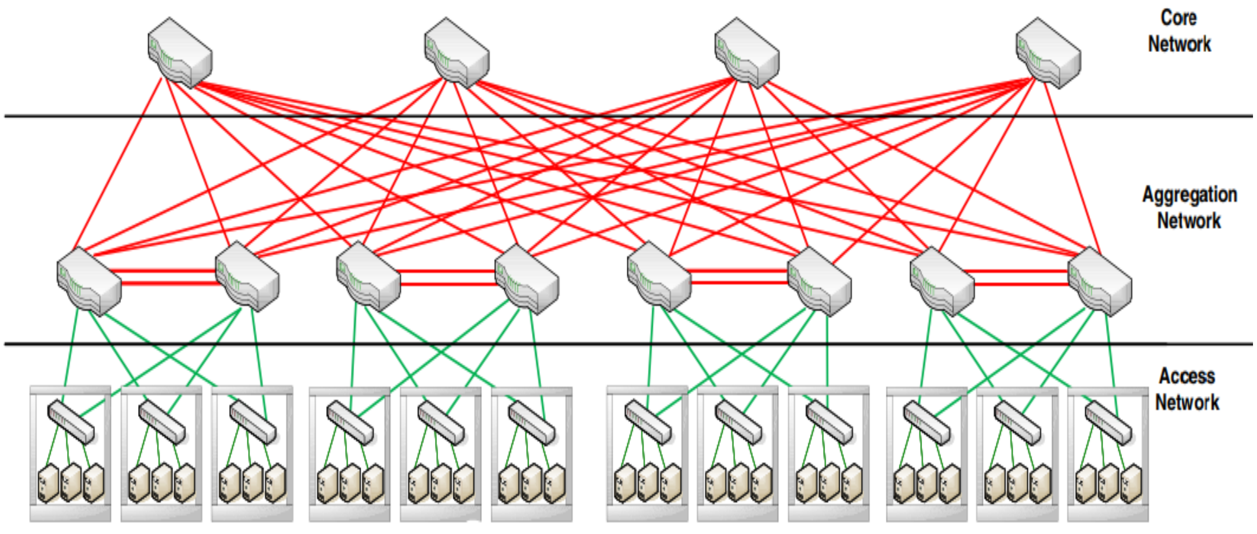


Figure 2- 3: A Three-tier DCN architecture [18]

2.2.2 Fat-tree DCN architecture

The multi-tier DCN architecture approach results in a serious bandwidth oversubscription towards the network core. To overcome this problem, Al-Fares et al. proposes a new DCN architecture called Fat-tree [1]. The main difference in Fat-tree is that all the aggregation level and core level switches are replaced with interconnections of a set of low-end switches. The main idea behind the construction of Fat-tree topology is to replace the high-end switches in multi-tiered topology by interconnecting several low-end switches because the total cost of low-end switches is less that the cost of high-end switches (switches with higher link bandwidth or higher number of ports).

The Fat-tree DCN architecture aims to maximize the end-to-end bisection bandwidth and to provide 1:1 oversubscription ratio [1] [9]. As the number of uplinks and downlinks for each pod are equal, Fat-tree DCN architecture provides a full bisection bandwidth. Bisection bandwidth is the maximum bandwidth that can be transferred across the midpoint of the system. According to [4], since all the switches in the Fat-tree topology are inexpensive low-end access level switches, this network topology is supposed to be highly scalable and economical [4].

The network elements in the Fat-tree topology also follow the hierarchical organization of the network switches in the three levels, unchanged with respect to the traditional DCN: access level, aggregation and core. The number of network switches, however, is much greater than the three-layer DCN architecture. To construct a Fat-tree DCN architecture the information we need is the number of ports present in each switch. Let k be the number of ports that each switch contains. Switches with k number of ports in a Fat-tree topology is called k-ary or k-port Fat-tree network topology. From the value of k, we'll derive the number of core switches, aggregation switches, edge switches and the maximum number of servers that can be attached.

A set of (k/2) number of aggregation switches and (k/2) number of edge switches are combined together and that is known as a Pod. A point of delivery, or Pod, is a module of network, compute, storage, and application components that work together to deliver networking services. The Pod is a repeatable design pattern, and its components maximize the modularity, scalability, and manageability of data centers [31]. A Fat-tree architecture is composed of k number of pods. Each pod contains two layers of k/2 switches. Each k-port switch in the lower layer is directly connected to k/2 hosts in the pod. Each of the remaining k/2 ports is connected to k/2 of the k ports in the aggregation layer of the hierarchy. There are (k/2) ^2 core level switches, each connecting to one aggregation layer switch in each of k pods [4] [13]. For example, Lets' take switches with 4 numbers of port that is the value of k is 4 (Figure 4). A k-ary Fat-tree DCN architecture will have (k/2) ^2 = (4/2) ^2 =4 core switches and number of pods is k which is 4. Each pod contains of (k/2) = (4/2) =2 aggregation switches and (k/2) = (4/2) =2 edge switches. Each aggregation switch within a pod is connected to (k/2) = (4/2) =2 core switches and (k/2) = (4/2) =2 edge switches. Each edge switch within a pod is connected to (k/2) = (4/2) =2 servers and (k/2) = (4/2) =2 aggregation switches. That means each pod will be connected to (k/2) ^2= (4/2) ^2=4 servers. Hence the maximum number of servers that can be connected to the network is (k^3)/4=16 servers.

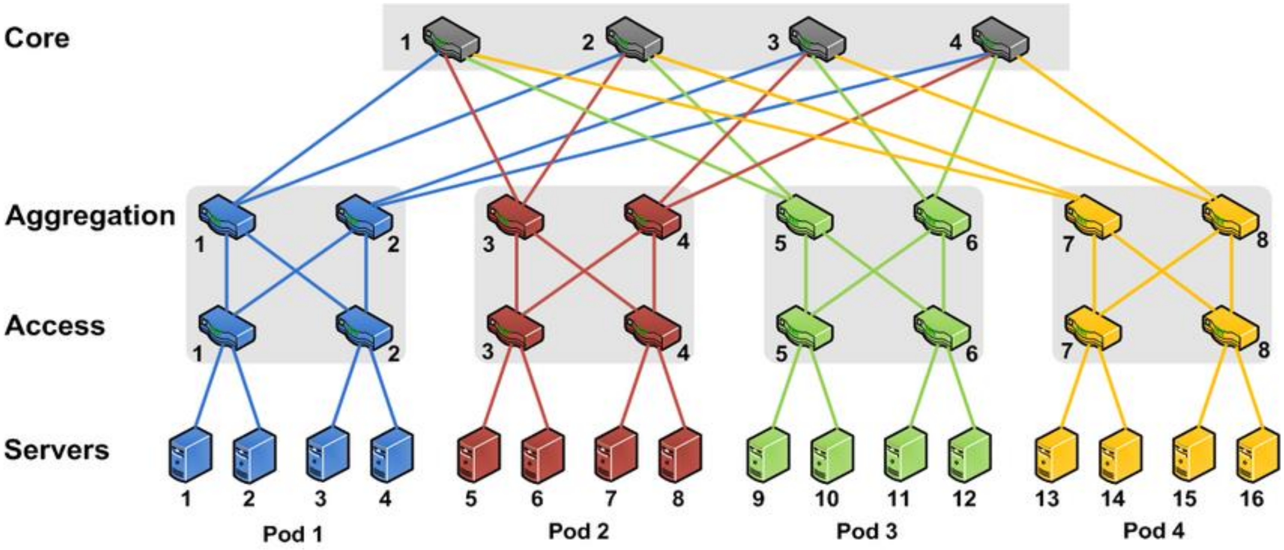


Figure 2- 4: A Fat-tree DCN architecture

2.2.3 BCube DCN architecture

The BCube architecture is a recursively defined structure which uses both switches and servers for routing traffic. BCube$_k$ is recursively defined from BCube$_0$. There are two types of devices in BCube: servers with multiple ports and switches that connect to a constant number of servers. It is recursively defined structure with BCube0 simply being n servers connected to an n-port switch. A BCube1 is constructed from n BCube0s and n n-port switches and a BCube$_k$ network can be constructed using n BCube$_{k-1}$ topologies and n^k n-port switches. In BCube$_k$, there are k + 1 levels, and N = n^{k+1} servers and k + 1 ports for each server, each port connecting a switch at each level. Figure 5 shows an example BCube architecture i.e. a BCube$_1$ with n=4 with 2 levels [21].

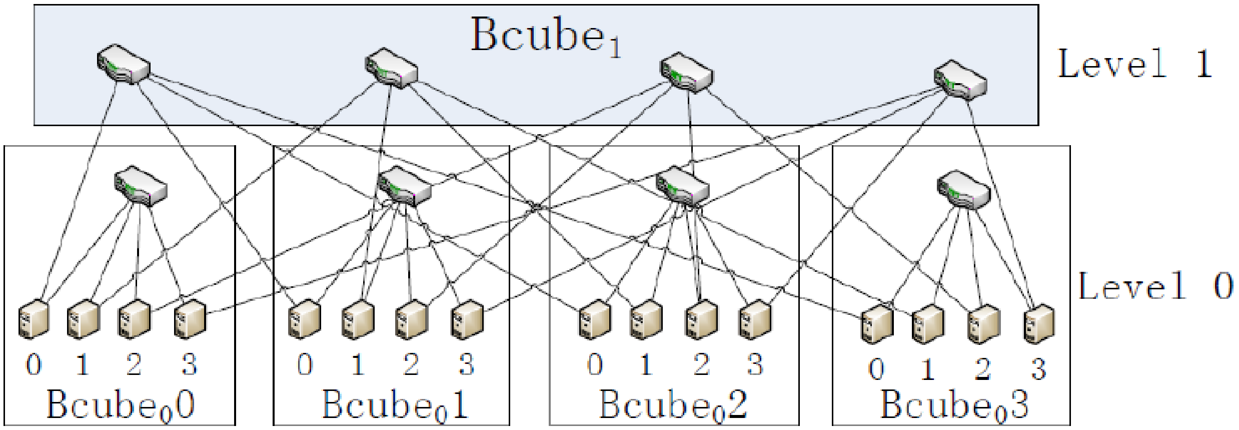


Figure 2- 5: A BCube$_1$ DCN architecture with n=4

2.2.4 DCell DCN architecture

A recursively defined DCell architecture is proposed by Guo et al. [20]. The architecture is extremely scalable and can easily scale to millions of servers in the data center. DCell is defined recursively by building higher level DCells using dcell0 as the basic building block. DCell is hybrid DCN architecture where one server is directly connected to many other servers and it uses a server equipped with multiple Network interface cards (NICs) and a commodity switches to construct its recursively defined architecture [19].

Each server is connected to a different level of DCells via its multiple links, but all the servers act equally. High-level DCells are built recursively from many low-level DCells. DCell uses only mini switches to scale out, and it scales doubly exponentially with the server node degree. Therefore, a DCell with a small server node degree (say, 4) can support up to several millions of servers without using core switches/routers [20]. A DCell$_0$ (level 0 cell) serves as the basic unit and building block of the whole DCell topology and a DCell$_0$ contains n commodity servers and one mini network switch. Higher levels of cells are built by connecting multiple lower level (level$_{L-1}$) DCells. Each DCell$_{L-1}$ is connected to all other DCell $_{L-1}$ in same DCell. The network switch is only used to connect the server within a cell0. A cell1 contains k = n + 1 cell0 cells, and similarly a cell2 contains k*n+1 cell1.

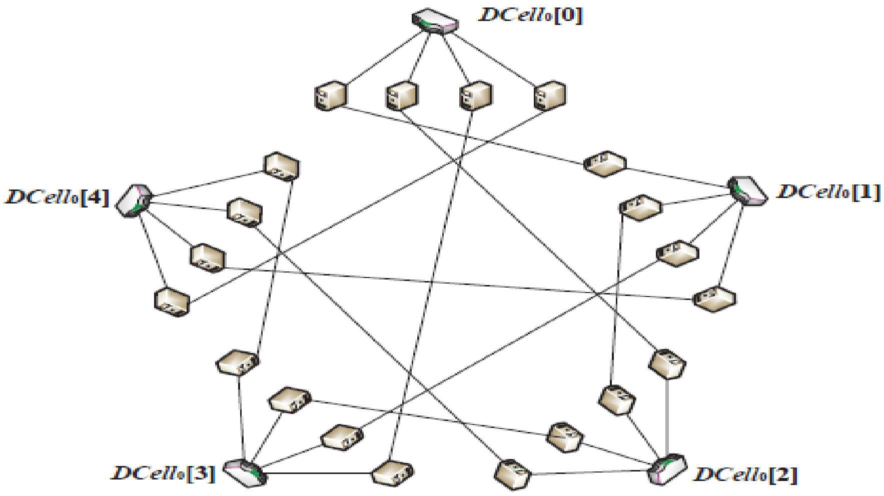


Figure 2- 6: A DCell1 DCN architecture with n=4.

2.2.4 Spine-and-Leaf DCN architecture

As previously stated, a conventional data center network architecture is based on a strict hierarchical tree model that contains a core layer, aggregation layer and access layer. The Three-tier DCN architecture is one of the traditional DCN architecture that was broadly deployed and served the data center well for many years providing effective access to servers within the pod and isolation between the pods and provides a predictable foundation for a data center network. Despite the aforementioned and other advantages, deploying the Three-tier DCN architecture may lead to significant oversubscription, latency and blocking (redundant links) architecture which is unsuitable for the requirements of today's advanced technologies and applications. With the growth of massive data transfers and rapid data travel in the network, the aging Three-tier design within a data center is being replaced with the Clos network–based spine-and-leaf architecture.

Traffic flowing inside the data center can be classified in two types based on the direction of the traffic flow, namely North-South and East-West. North-south consists of traffic that arrives from outside the data center (application query e.g.,) and leaves after required processing within the data center (response). On the other hand, east-west traffic is intra-data center; which flows from one server to the other inside the data center to complete either computation or storage related task [23]. The most significant reason that a shift needed from the conventional Three-tier data center network architecture to the classic leaf-spine architecture is the change in data center traffic patterns due to compute and storage infrastructures alterations. According to [23], almost 80% of the traffic that traverses within the data center is East-West, thus a new data center network design is required to handle this much amount of traffic.

In a leaf-spine data center network architecture every leaf switch is connected to each of the spine switch in a full-mesh topology which makes the DCN architecture flatten and eliminate bandwidth aggregation/oversubscription, because of bandwidth being the same at every tier. As a result, this architecture provides a high degree of network redundancy and a simplified design and supports a large volume of bi-sectional traffic (east-west).

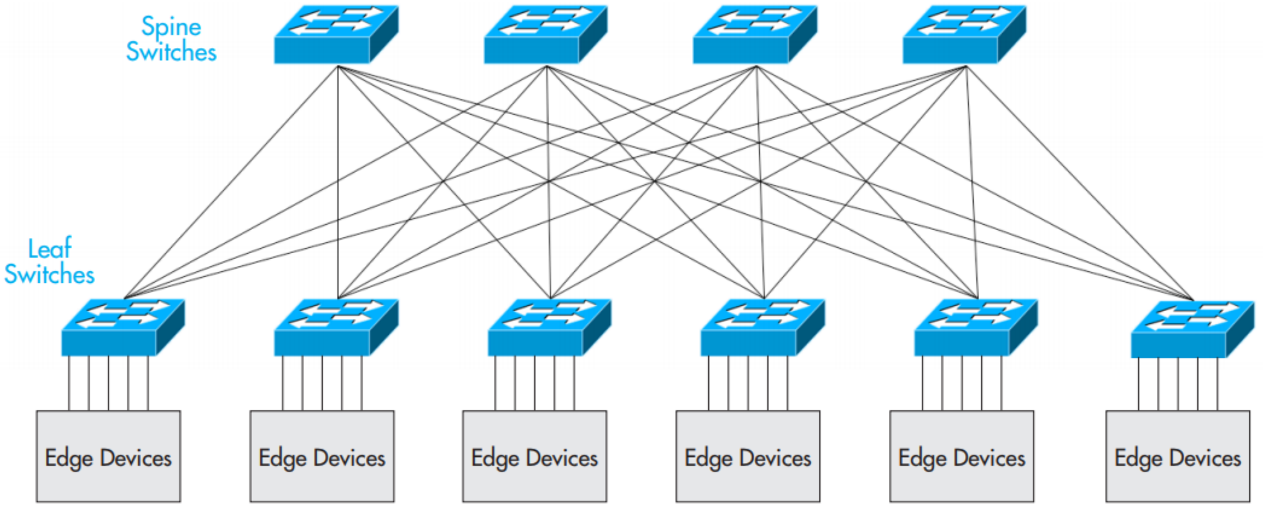


Figure 2- 7: A Leaf-spine DCN architecture

The spine-leaf mesh can be implemented using either Layer 2 or 3 technologies depending on the capabilities available in the networking switches, meaning that the links between the leaf and spine layer could be either switched or routed. In either design, all links are forwarding; i.e., none of the links are blocked, since spanning-tree protocol is replaced by other protocols. Layer 3 spine-leaf require that each link is routed and is normally implemented using Open Shortest Path First (OSPF) or Border Gateway Protocol (BGP) dynamic routing with equal cost multi-path routing (ECMP). Layer 2 utilizes a loop-free Ethernet fabric technology such as Transparent Interconnection of Lots of Links (TRILL) or Shortest Path Bridging (SPB) [24].

2.4 Traffic Flow in Data Center Network

2.4.1 North-South traffic

Data center network traffic is characteristics by two main traffic patterns , north-south traffic and east-west traffic flow. The north-south traffic flow refers the ingress traffic coming from outside the data center examples from the end-users and the traffic that leaves the data center through the data center core layer or data center perimeter (edge routers) network devices. The traffic that comes from the end-users includes web searching, audio and video online streaming , uploading and downloading data. The north-south traffic flow is not only the traffic coming from the end-users to the data center and from data center to the internet network, but also includes the traffic that flows from another data center to data center so called inter-DC traffic. The inter-DC traffic flow refers moving data between the data centers in order to sore data, increase resource utilization and disaster recovery purpose data prevention and backup strategies.

2.4.2 East-West traffic

The east-west traffic flow refers the traffic within the data center. This includes intra-rack (machine to machine communication within the same rack), inter-rack (machine to machine with different rack), intra-pod/module (machines which is installed to serve a specific to different services) and inter-pod/module traffic flow. Some examples of east-west traffic include queries and commands to retrieve data between web servers, application servers and database servers, moving data from test-bed environment to production environments and vie-versa, coping and storing data to storage machines, sharing routing table among routers and layer-3 switches when there is triggered updates to have the same copy of inter-network information for network convergence.

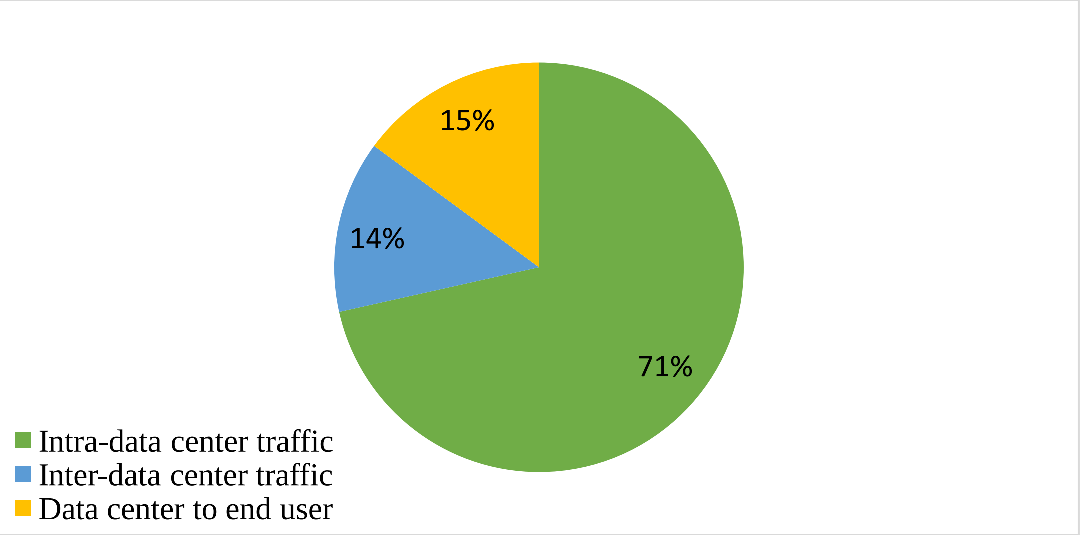


Figure 2- 8: Global Data center Traffic [21]

The east-west traffic is the maximum and noticeable volume of traffic produced in the data centers and it is the predominant traffic category of data center traffic flows. The advancement of server virtualization technologies has a significant impact on the increasing amount of east-west traffic. Big data server to server communication also drives the great increase in east-west traffic. According to [21], the east-west traffic flow is the largest volume of traffic i.e. 71.5% that remains within the data center and the 14.9% traffic flow belongs to data center to end user. The remaining traffic percentage is inter-data center which is approximately 13.6% and this type of traffic will increase gradually.

As [21] the Cisco global index forecasting 2016-2021, Traffic between data centers is growing faster than either traffic to end users or traffic within the data center, and by 2021, traffic between data centers will account for almost 14% of total data center traffic. In addition to this, east-west traffic (traffic within the data center and traffic between data centers) will represent 85% of total data center by 2021, and north-south traffic (traffic exiting the data center to the Internet or WAN) will be only 15% of traffic associated with data centers. This large volume of data center traffic is due to the rapidly growing demand of modern Internet applications such as cloud computing, Network virtualization solutions such as software define networking solutions, social networking and online audio and video streaming. Therefore, it is important to give high emphasis and re-engineered the current data center network infrastructure to support the continuously growing traffic demand.

2.5 DCN Performance Metrics

The main performance metrics or measurements of a data center network are throughout, delay, packet lose and fault-tolerance. The major driver of these measurements is the oversubscription ratio of the network which quantify the network bandwidth between all the host sharing [11]. As more new application needed to install on the data center, the computing device such as servers, storages, switches and cables added to support those applications also increase. These network components are prone to failures and a single point of failure can lead to subsequent performance degradation by blocking the usage of many servers [55]. Also, if the network is blocking and congested, the dropped packets will be increase and which in turn increases the packet delay time. This high delay time produces low network throughout and degrades the overall performance of the network. Hence, the data center network is designed and deployed to achieve a network with a high throughput, low latency and high fault-tolerance on maximum triggered traffic flows on peak-time. The main performance measurements or metrics of a data center network are discussed int the following sub-sections.

2.5.1 Packet delay

Delay is defined as the time required for a packet to travel from a source to destination along its entire path [11]. It is a collection of the transmission delay, propagation delay, queuing delay and the processing delay. The processing delay is the time required by intermediate routers to decide where to forward the packet or discard the packet and the queuing delay refers the time a packet is enqueued while the link is busy sending other packets.

The transmission delay is the time required to push all the bits in a packet on the transmission medium and once a bit is pushed on to the transmission medium, the propagation delay refers the time required for the bit to propagate to the other end of its physical path. The major contributor in the packet delay are the queuing and processing created at each participating device in the transmission. Packet delay will increase when devices links congested due to high link oversubscription ratio, packet buffering when high traffic flows arriving at the same time and re transmission of lost packets.

2.5.2 Throughout

In a network, throughput (or accepted traffic) is the rate (bits/sec) at which traffic is delivered to the destination nodes or it simply refers the capacity of the network to transfer data in a given period of time [55]. The main factors of throughout are the provisioned bandwidth of the link, network congestion, packet loss and errors and packet delay [13]. To design a data center network with high throughout, the data rate between devices that packet traverses should be set to high specially on the uplinks to the middle layer and if there are multiple routing paths for the packet between the devices, traffic flow will load-balancer on these paths which minimizes less traffic congestion improves the throughput of the data center network [51].

2.5.3 Oversubscription ratio

A basic parameter for data center network topologies is the oversubscription ratio and it is a major factor that influences data center network scalability also. It is basically a metric to quantify how network bandwidth is shared among all hosts [4]. The amount of oversubscription is expressed by the oversubscription ratio x:1 where x denotes the factor by which the full bisection bandwidth is reduced and oversubscription between two layers in a DCN means that the sum of the capacities of all uplinks at a layer is less than the sum of the capacities of its downlinks.

In a tree-like topology, oversubscription ratio is typically defined as the ratio of the downlink bandwidth capacity to the uplink bandwidth capacity at any layer of the DCN network topology [4]. For instance, an oversubscription of 1:1 indicates that all hosts may potentially communicate with arbitrary other hosts at the full bandwidth of their interface bandwidth. An oversubscription value of 5:1 means that only 20% of available host bandwidth is available for some communication patterns and the oversubscription 4:1 means that the communication pattern may use only 25% of the available bandwidth. Different layers of Three-tier DCN architecture are oversubscribed at different threshold values and variation in the oversubscription ratio at the various network layers is based on the physical infrastructure [11].

In the DCN north-south traffic patterns, statistically, not all connected users and running application consume the maximum allocated bandwidth at any particular instant of time rather there is an intermittent and triggered way of sharing their bandwidth. Because most application in the data center have short-lived/bursty flow patterns, meaning the actual bandwidth consumption may triggers the maximum threshold value when a large amount of data transmission is occurred in short time [24]. To handle such unbalanced bursty flow patterns, switch links will be congested and if such bursty flow patterns increased, the DCN performance will be affected highly. In the data center servers install in one ToR or access switch should have a 1:1 oversubscription ratio to communicate with full interface bandwidth. But this ratio can’t achieve on the aggregation and core layers since it is expensive, hence the oversubscription ratio is defined for the purpose of optimizing the cost of the network design. Designing networks for full bisection bandwidth is costly and unnecessary for smaller companies or enterprises.

As a result, many data centers networks may be oversubscribed, meaning the total inter-rack network capacity may be less than sum of intra-rack capacities across all racks. The underlying assumption is that applications are mostly rack local [56]. In realty the nature of traffic flow in the data center is intermittent, meaning sometimes the traffic flow will be extremely high and requires more bandwidth than the provisioned and sometimes there will be minimum percentage of bandwidth usage. By taking this behavior of traffic flow, Internet Service Providers and enterprises introduce oversubscription ratio to minimize the cost of hardware components and network deployment.

According to Al-Fares et al. [1], cost is a major factor that affects the data center network design and related decisions. As stated earlier one method to reduce costs of data center components is to oversubscribe data center network. However, oversubscription leads to have poor performance and low scalable data center network. Paper [52] reports the amortized costs of data center components as describe in the following table.

Table 2- 1: Cost of data center components [52]

Amortized Cost	Components	Sub-Component
~45%	Servers	CPU, memory, storage systems
~25%	Infrastructure	Power distribution and cooling
~15%	Power draw	Electrical utility costs
~15%	Network	Links, transit, equipment

According to [52], the highest percentage of data center costs go to server components compared with the other components and as the report reveals, network components are not much expensive than servers and cooling systems which are the key to minimizing investment costs. It is a cost-effective way to attach more devices to the network, provided that the applications can tolerate the risk of losing a packet or occasionally the network unavailable, but this risk gets progressively worse as the level of oversubscription is increased [54]. Oversubscription can overwhelm the hardware packet buffers and lead to packet loss, if the queue exceeds the size of the physical hardware buffer, packets are dropped. Performance degradation and increased latency may result in significant revenue loss. Goggle reported 20% revenue loss because of an experiment that added an extra delay of 500ms in displaying the search results. Amazon experienced 1% sales decrease because of 100ms additional delay [53].

2.6 DCN Simulation

2.6.1 Network simulator

Simulation in computer science, is the technique of representing the real-world condition, event, or situation by a computer program or it is an animated model that mimics/imitates of an existing or proposed system to find a cause of a past occurrence or to forecast future outcomes [39]. By mimicking the behavior of each part of the process as it interacts with other parts, it helps to understand how the whole system will perform and try alternative ways to provide resource capacity or innovative ways to improve performance.

Networking community is largely depending on simulation to evaluate the behavior and performance of network protocols for various networks. Simulators are used for the development of new networking architectures, protocols or to modify the existing protocols in efficient environment [32]. A Network simulator is software that predicts the behavior of a computer network. Network simulator allows researchers to test the scenarios that are difficult or expensive to simulate in test-beds environments because it is very costly to deploy a complete test bed containing multiple networked computers. While testing a scenario, it is very difficult to setup a complete network containing computers, routers and data links to see the feasibility of the network. In these circumstances, network simulators are used to set up, test and improve performance of any computer network [33].

Most of the commercial simulators are GUI driven, while some network simulators are Command-Line Interface (CLI) driven. The network design and configuration describe the state of the network (nodes, routers, switches, links) and the events (data transfer, transmission delay, packet error etc.). An important output of simulations are the trace files. Trace files log every packet, every event that occurred in the simulation and are used for analysis. Most network simulators use discrete event simulation, in which a list of pending "events" is stored, and those events are processed in order, with some events triggering future events such as the event of the arrival of a packet at one node triggering the event of the arrival of that packet at a downstream node [35].

Discrete event simulation is a computer-based simulation method which is particularly effective for modeling the performance of systems which are driven by activities occurring at discrete instants in time (events). In discrete system is a system whose state may change only at discrete point in time i.e. state variables change instantaneously at separated point in time. Discrete event simulations are powerful techniques for optimizing processes and making confident and evidence-based decisions. Such simulations can serve as an effective tool to evaluate the performance of communication systems carrying a diverse mix of traffic [38]. Most researchers and network designers make use of discrete event simulation in cellular networks for addressing issues related to the cellular features, mobility, handovers between neighboring cells and traffic [34].

2.6.2 Types of Network Simulators

Currently there are many both free/open-source and proprietary network simulators that have different features in different aspects. There are different network simulators with different features. Some of the network simulator are NS2, NS3, OPNET, OMNeT++, DCNsim, NetSim, REAL, J-Sim and QualNet. These network simulators are classified in to two, commercial and open source. Some of the network simulators are commercial which means that they would not provide the source code of its software or the affiliated packages to the general users for free. All the users have to pay to get the license to use their software or pay to order specific packages for their own specific usage requirements. One typical example is the OPNET(Optimized Network Engineering Tool). It is extensive and powerful simulation software with wide variety of possibilities to simulate entire heterogeneous networks with various protocols. However, OPNET as some disadvantages such as Costly, high memory consumption, complex GUI operations, Insufficient tutorial and document since it is less usability [62].

The open source network simulator has the advantage that everything is very open and everyone or organization can use it without payment. Typical open source network simulators include NS2 and NS3. NS-2(Network Simulator version2) is a discrete event simulator targeted at networking research and provides substantial support to simulate group of protocols like TCP, UDP, FTP and HTTP, routing, and multicast protocols over all wired and wireless networks [61].

Some advantages of NS2 simulators includes availability of analysis tool and network visualization tool, Complex scenarios can be easily tested, and it has large numbers of models. But, in NS2 simulation process may slow down, if large numbers of nodes are simulated, difficult to understand and analyze the scripts, tracing system is difficult to use and difficult use it on real infrastructure [63].

NS3 is also an open sourced discrete-event network simulator which targets primarily for research and educational use. NS3 acts both as simulator as well as emulator that allows network administrators for integration with real networks. Some of the advantages of NS3 includes, it is more flexible than any other simulators, support large-scale networks, has large number of modules and supports many protocols to simulate.

OMNeT++ (Optical Micro-Networks Plus Plus) similar with NS2 and NS3, OMNeT++ is also a public-source network simulator with GUI support in the simulation of both wired and wireless network. OMNeT++ has advantages such as it provides a powerful GUI environment, simulation can be performed or executed under graphical user interface and tracing and debugging are much easier than other simulators. But, like the other simulators it has some limitations such as poor network performance analysis, provides few numbers of models, it does not offer a great variety of protocols and very few protocols have been implemented [63].

Generally, each network simulator has their domain of relative strength and weakness compared to other simulators. Amongst the available open source network simulators, OMNeT++ and NS-2 are the most appropriate one. The main strength of OMNeT++ is its GUI support, while the strength for NS-2 is that it provides a large number of models, also NS-2 is most popular in academic research. OMNeT++ and NS-3 can be used for large-scale network simulation. Also, NS-3 provides better simulation results, utilization of IP addressing to and greater arrangement with Internet protocols and simple tool appropriate for simulation of wireless network.

2.7 Related works

Data center network architecture is a major part of data center design and has become a backbone of the data center, which needs to provide reliable and scalable communication services, while continuously being challenged in terms of the architectural scalability, energy efficiency, resource utilization, cost effectiveness, Quality of Service (QoS) and other related issues. Data center network (DCN) is typically based on the Three-tier DCN architecture. The Three-tier data center network architecture is a hierarchical tree-based structure comprised of three layers of switching and routing elements having enterprise-class high-end equipment in higher layers of the hierarchy. The increase number of servers connected to the access layer directly impacts the available bandwidth of aggregation layer and core layer and these links will be typically oversubscribed and there will be Poor server to server connectivity i.e., the capacity of the links between access switches and core switches/border routers is significantly less than the sum of the output capacity of the servers connected to the access switches [26].

Over the last decades, a variety of data center network architectures have been proposed and each trying to improve some limitation aspects and to ensure that computations are not bottlenecked on communication. However, as data center networks continue to grow, there are challenges remain in their structure and the fundamental challenge in data center networking is how to efficiently interconnect an exponentially increasing number of servers [20], i.e. scalability and how to achieve a better performance (high bisection bandwidth, low end-to-end delay and high network throughput) from DCN architectures.

In response to this, Al-Fares et al. [13] proposed a k-ary Clos-based data center network architecture, Fat-tree. This DCN architecture utilizes homogeneous low-cost commodity off-the-shelf network switches and servers throughout the whole topology. The Fat-tree DCN architecture attempts to maximize the end-to-end bisection bandwidth and minimize fault-tolerance, since it uses multiple paths between each node. In addition to this, the architecture is highly cost effective and energy efficient in contrast to the Three-tier DCN architecture because it avoids the high-end expensive devices.

The Fat-tree DCN architecture build with k-port homogeneous switches throughout the entire network offers 1:1 oversubscription ratio and the network work with a full bisection bandwidth i.e. same number of uplinks and downlinks at each layer, meaning each core, aggregation and access layer switches used to interconnect the servers have an equal number of ports with same port speed as end host ports.

However, designing and deploying the k-ary Fat-tree topology has same drawbacks and among these wiring/cabling complexities is the major one. In campus or corporate local area networks, wiring complexity is not significant but in data center environment it is a critical issue when many tens of thousands of network nodes are installed. Operation and maintenance are also the major issues of the Fat-tree DCN architecture which are caused by increased number of networking components and links or cables. Due to the strict behavior of the topology and since the topology is built with homogeneous switches, network expansion, upgrading and rewiring cost is expensive.

It also requires high attention when performing networking component replacements such as switches, servers, media type and converters. Any changes on the lower layer switches highly affect the upper layer network switches which intern affects the scalability characteristics of the architecture by increasing operational and maintenance cost. In the Fat-tree networks which rely on homogeneous switches, dynamic reallocation of any services or the ability to assign any service to any server (agility) requires replacement of these existing homogeneous switches in the network and needs network rewiring activities and such activities are performed manually which are time consuming, expensive and error prone. Generally, even though the Fat-tree data center network architecture designed to deliver 1:1 oversubscription ratio and full bisection bandwidth, deploying this type of data center network will introduces some major issues such as low scalability and cabling complexities.

The Fat-tree data center network build with k port switches will have k^3 wires to interconnect those network devices which is a very complex to manage. The scalability of Fat-tree data center network architecture is limited to the number of switch ports and the maximum number of pods is equals to the number of ports in a switch. In addition to this, the Fat-tree data center network architecture is a non-blocking, fault-tolerance(physically) and have equal cost multiple paths for each traffic reduces the link congestion in the upper layers.

But to support non-blocking communication, the Fat-tree architecture requires a large number of switches at the core layer and aggregation layer which in-turn requires a large number of wires to interconnect those switches, finally increase deployment cost, energy consumption, and management complexity. In addition to this, a Fat-tree network may require a complex routing and other IP protocol stack configurations to be done on each switch to avoid layer 3 loops to effectively use all available paths for load balancing.

Greenberg et al. [43] proposes a Clos-based data center network architecture build with layer2 switches called, VL2 to address that address the oversubscription ratio problem and aims at achieving flexibility in resource allocation. VL2 networks constructed by replacing the low-cost commodity switches used in Fat-tree network by high-speed switches. This type of data center network architecture provides fault-tolerance using its load balancing algorithm called Valiant load balancing algorithm and it also provides easiness routing by forwarding packets using location-specific addresses (LAs) and application-specific addresses (AAs) used by switches and servers.

However, VL2 suffers from low scalability and expensive switches to be implemented (ports capacities are 10 times those of Fat-tree). One limitation of VL2 is the lack of absolute bandwidth guarantees between servers, which is required by many applications e.g., multimedia services [15].

Besides the evolution of Clos-based data center network architectures, Guo et al. [20] designed and proposed server-based data center network architecture called DCell. DCell data center network architecture is constructed using servers with multiple NICs and mini low-cost switches which replaces expensive core and aggregation switches. DCell architecture provides high scalability data center network, but DCell provides low bisection bandwidth that causes high traffic congestion, high packet delay which leads to very poor network performance. In addition to this, deploying DCell architecture requires performing changes on the server's protocol stack because in DCell data center network architectures servers used as both computing nodes and routing nodes.

2.7.1 Performance Analysis of DCNs

Different research works were conduct a comparative analysis of several well-known data center network architectures using main performance metrics and some of these works are presented in the following paragraphs. Paper [5] carried out a comparative study of the major data center network architectures DCell and Fat-tree that were designed to addresses the network capacity and the oversubscription ratio of links respectively by implementing low-cost commodity switches. The paper evaluates the performance of the DCell and Fat-tree architectures using packet level simulator NS-3 and compare the performance of those networks in-terms of throughput and average packet delay.

The simulation results show that for small number of nodes, the average throughput of DCell architecture is greater than the Fat-tree network and for large number of nodes in the network, the Fat-tree architecture gives high throughout than the DCell architecture. For the average packet delay a similar behavior is observed i.e. the DCell architecture outperforms the Fat-tree based architecture for small number of nodes but gradually increases as the number of nodes increase. Generally, the performance of Fat-tree architecture is better than the DCell architecture in terms of average network throughput and packet delay. Therefore, even if DCell architecture supports large network capacity it suffers with poor network performance issues and future incremental expansion will be probability impossible.

Paper [11] presents a performance comparison of the three major DCN architectures, Three-tier, Fat-tree, and DCell architectures and evaluates these architectures based on average packet delay and throughout using one-to-one and one-to-many traffic patterns. According to the simulation results the performance of both the Three-tier and Fat-tee architecture is almost the same and these two architectures outperform than the DCell architecture in terms of average packet delay and throughout. But for small number of nodes in the network, the DCell architecture produces low average packet delay and high throughout which is greater than the Three-tier and Fat-tree architectures for both one-to-one and one-to-many traffic patterns because the oversubscription ratio at the first levels of DCells is minimum i.e. almost non-subscribed. Here also the performance of the Fat-tree architecture is higher than the Three-tier and DCell architecture for both traffic patterns, to produce low average packet delay and high throughout large number of switches are required so as to provide full bi-section bandwidth at each layer of the architecture which leads to have deferent issues. The paper implements ECMP(Equal Cost Multi Path) routing algorithm which used to load-balancer and distribute the traffic to equal cost routing paths and ECMP routing highly contributes to the better performance of Three-tier DCN architecture since having multiple routing path to the same destination can helps to distribute the traffic for those available paths reduces congestion, packet loss and packet delay.

Paper [57] reports a comparative analysis of three switch-based data center network architectures i.e. Fat-tree, Three-tier and flattened-butterfly architectures by varying number of clients that access database servers using OPNET network simulator. The simulation result reveals the Fat-tree architecture offers better throughput than Three-tier and flattened-butterfly (FBFLY) architectures for large number of clients but when small number of clients access the database servers flattened-butterfly architecture has better throughput than multi-tiered and Fat-tree data center architectures. Also, the Fat-tree data center network architecture offers minimum delay compared to Three-tier and flattened-butterfly data center network architectures.

As observed in all the above papers the performance of the Three-tier DCN architecture is less than the Fat-tree architecture but, if the oversubscription ratio value at the aggregation and access layer of the Three-tier DCN architecture was considered during the simulation, may be the performance of Three-tier DCN architecture will outperform than the Fat-tree data center network architecture. However, these papers [5] [11] [57] doesn’t discusses and mentioned how the Three-tier data center network architecture is oversubscribed at each layer of the network structure.

If the oversubscription ratio of Three-tier data center network architecture is high both at the aggregation and access layer, it is obvious that the average packet delay will be higher than the Fat-tree data center network architecture and the throughput will also less than the Fat-tree data center network architecture because high oversubscription ratio causes blocking and congested links which increase to packet loss and packet delay. That is why the Fat-tree architecture outperforms than the Three-tier, DCell and flattened-butterfly data center network architectures in terms of both the average packet delay and network throughput. Since to support large number of clients and to have 1:1 oversubscription ratio at each layer of the network architecture, the Fat-tree architecture must be constructed using more switches than the Three-tier and DCell architectures.

2.7.2 Scalability Analysis of DCNs

Paper [4] evaluates and analyze the impact of oversubscription ratio on the scalability of Three-tier architecture by setting different oversubscription ratios values. The paper sets the overall or total oversubscription ratio to 9:1 (aggregation level 3:1 and access level 3:1), 4:1 (aggregation level 2:1 and access level 2:1) and 1:1. According to the simulation results, the Three-tier architecture can scale better on an oversubscription ratio of 9:1 for large-scale data centers. Paper [4] also evaluates the scalability of different data center network architectures Three-tier, Fat-tree , FBFLY, Camcube and BCube by set the oversubscription ratio of each topology the same since oversubscription ratio is the major factor that influences network scalability.

Paper [4] sets the oversubscription ratio to 1:1 for comparison purposes and a 1:1 oversubscription indicates that all hosts have the capability to communicate with other hosts with full link bandwidth. The simulation result shows that, FBFLY outperforms the other two switch-based topologies by supporting more hosts for a given switch port count and BCube, also has better scalability compared to Fat-tree and Three-tier architectures.

However, deploying a Three-tier architecture with an oversubscription ratio of 1:1 needs high bandwidth switch ports may be 40G at the core layer and the aggregation layer and requires high investment, paper [4] doesn’t consider such challenges. Despite researchers design and propose different types of data center network (DCN) architectures, designing a novel data center network architecture that addresses two or three main requirements is remains challenging. The major challenge that today’s data center networks suffers is scalability i.e. when incremental expansion and upgrading the existing network is required, specially switch-based DCN architectures because these types of architectures has fixed structure properties.

In response to such challenges, Curtis [27] , proposed a new network upgrading and expansion solution called LEGUP to improve the performance of tree-based networks by developing the theory of heterogeneous Clos networks. This heterogeneity theory allows modern and legacy network components like switches to coexist in the network. To design cost-effective network topologies, especially as the network expands over time, updated equipment must coexist with legacy equipment, which makes the network heterogeneous [30].

LEGUP is a flexible optimization framework which attempts to strength the bisection bandwidth to achieve agility (the ability to assign any server to any service) without uninstalling the existing network rather incrementally adding new network components. However, LEGUP's approaches leaves some free ports for future expansion stages which needs to pay significantly cost for the core, aggregation and ToR level switches.

Unless free ports are preserved for such expansion which is part of LEGUP's approach, this can cause significant overhauls of the topology even when adding just a few new servers [29]. LEGUP also doesn't account for the reconfiguration and rewiring cost after modifying the data center network architecture because performing reconfiguration and rewiring a network is more expensive and fault-sensitive since such activities are done manually. In addition to this, as LEGUP changes and selects the location of existing and new networking components, it is will be subjected to space, thermal, and power constraints.

LEGUP, as mentioned earlier designed for restricted and regular Clos network topologies in the data center like Fat-tree DCN topologies and this approach requires a Clos network as input to generate an arbitrary network topology with high performance as output. The restricted behavior of Clos networks also decreases the performance of LEGUP's framework and to deal with such scenarios, Curtis [28] proposes REWIRE, an optimization-based framework for data center networks which avoids the restriction of Clos networks by accepting any unstructured network as input and returns arbitrary topologies as output.

REWIRE is a framework proposed to design new, upgraded and expanded data center networks to maximizing network performance i.e. finding maximum bisection bandwidth and minimum end-to-end latency. However, REWIRE is subjected to different constraints and the main drawback of this framework is budget. During network expansion using REWIRE new cables will be requires and may be existing cables will be moved as per the newly network structure. The main constraint of this approach is the cost of new switch that will be added, and the algorithm doesn't consider the addition of switches into account. This approach also doesn't consider the patch panels which are used to interconnect and manage incoming and outgoing LAN cables because patch panels are the most important components of data center network architecture. It is also subjected to physical constraints such as space, power and cooling systems due to the newly output architecture.

2.7.3 Summary

As discussed in section 2.6.2, performance comparison of DCNs, the Fat-tree architecture is delivers low average packet delay and throughput when compared with the Three-tier and DCell architectures for different traffic patterns. This main reason for such high network performance in the Fat-tree architecture is since the architecture employs large number of switches so as to deliver 1:1 oversubscription ratio and to provide more end-to-end bandwidth in all layers of the architecture. However, to offer 1:1 oversubscription ratio in all layers of the network topology, more network switches required and as the number of these switches in a data center grows to support large number of servers, it becomes difficult to build and maintain these networks. Thus, the Fat-tree architecture may not be suitable in large-scale data centers, because Fat-tree has a high ratio of the number of switches to the number of servers specially the architecture requires larger numbers of switches at the aggregation and core layers.

The DCell architecture delivers low average packet delay and high throughput for small size network as compared with the Fat-tree and Three-tier architectures. However, for large-scale networks DCell architecture suffers to high oversubscription ratio on the highest level of DCells that lead to very high packet delay and low throughput. So, the performance of DCell architecture is heavily dependent of the size of the network deployed in the data center. The performance of Fat-tree architecture doesn’t depend on the network size and the performance of the Three-tier architecture is on the network size and the oversubscription ratio on each layer on the network architecture.

Scalability is one of the major challenges of data center network architectures and data center network should be scalable to support growing number of nodes that are required to handle the new application demands and Internet technologies added in the data center. Even if some DCN architectures are proposed to address this issue such as DCell and BCube, they introduce different challenges. For example, the DCell architecture offers a scalable data center network architecture but, as the network increases it suffer poor network performance due to poor cross-section bandwidth and very high over-subscription ratio. Even, the Fat-tree architecture builds a network with high cross section bandwidth and 1:1 oversubscription ratio, it suffers from low scalability which is limited to the total number of ports in a switch.

CHAPTER THREE: SIMULATION IMPLEMENTATION

3.1 Introduction

The cost of building test-bed environments or actual systems for performance analysis is sometimes not effective or even not feasible and the need of simulation comes in first solution. The main purpose of doing a simulation of the data center network architectures is to understand the behaviors of the architectures and to provide a comprehensive insight in a realistic manner.

In this study the Three-tier DCN architecture is simulated under different workloads and network conditions to analyze the performance characteristics of the architecture and Network Simulator 3 (NS-3) is used for simulation. NS-3 is a discrete-event network simulator is written in C++ programming language with Python bindings for Internet systems targeted primarily for research and learning purpose. Using NS-3 we can create PointToPoint, Wireless, CSMA, etc. connections between nodes. PointToPoint connection is same as a LAN connected between two computers and wireless connection is same as Wifi connection between various computers and routers and CSMA connection is same as bus topology between computers. The below figure depicts the software architecture of NS-3 simulator.

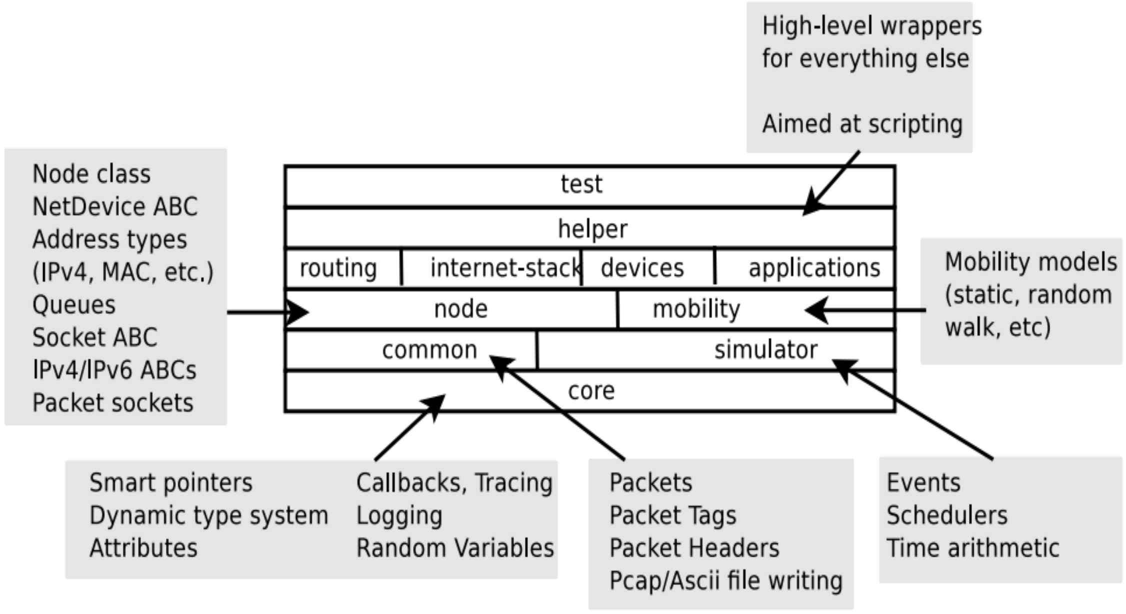


Figure 3- 1: ns-3 architecture [40]

The Software organization of ns-3 has a modular structure, as depicted in the above Figure. The ns-3 source codes are mostly implemented in the ‘src’ directory and the above modules are organized in this directory. All the ns-3 modules are depending on the core module and each module only have dependencies on modules under them. The core module in the lower layer contain all the basic components used to create the simulation. Packets, packet tags, packet headers are implemented in the network/common module. The internet (node) module is aimed at creating all the network nodes inside the simulation. It enables also the definition of protocol stacks, network interfaces and applications for each node. The protocol module enables the utilization of different routing protocols inside the simulated network. Finally, at the top of the model, the helper defines useful interfaces for the user to implement the characteristics of the scenario [40].

Data center networks are usually large-scale networks consisting of thousands of servers thus it requires large amount of memory and computation power(simulation run-time) to simulate a DCN. Paper [41] conducts a performance comparison of recent network simulators and finally the results show that ns-3 outperforms with the performance metrics (effective simulation run-time and memory usage) at different network sizes and ns-3 is capable of carrying out large-scale network simulations in an efficient way as described in Figure 10. Beside low memory usage and fast computation power characteristics, the most important salient features of ns-3 simulator are an implementation of real IP addresses stacks on node, real network bytes are contained in simulated packets and packet traces can be captured and analyzed using tools like Wireshark.

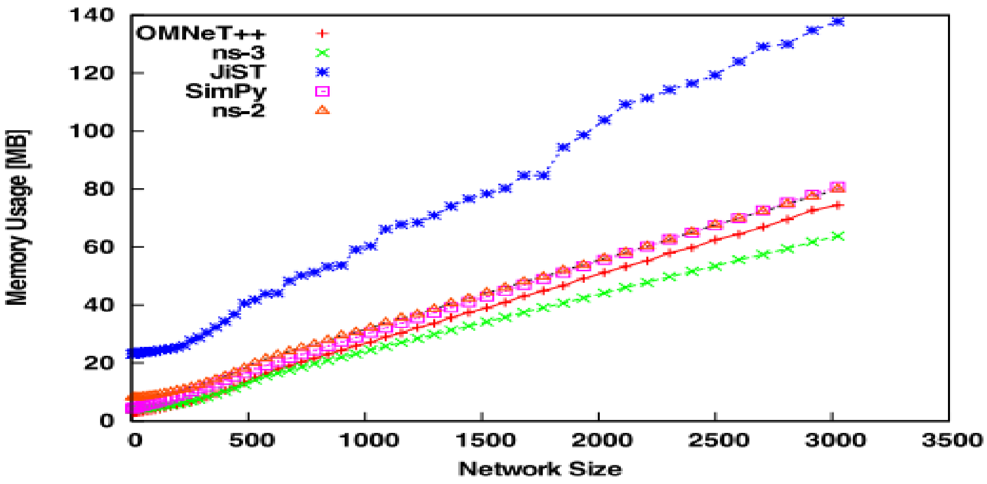


Figure 3- 2: Comparison of network simulators [41]

3.2 Simulation Setup and Parameters

3.2.1 Environment

For simulating the Three-tier DCN architectures, Ubuntu 18.04 operating system and network simulator-3 version 3.27 (ns-3.27) an open-source simulator tool installed on HP-640 core i7 Laptop computer with 4G RAM was used. Due to the processing capability of the Laptop computer, simulating a DCN architecture with large number of hosts was not performed because it takes much time while simulating more than 4600 hosts.

3.2.2 Network architecture setup

The Three-tier architecture is a type of stable architecture and most of today's commercial data center networks use a conventional hierarchical topology. The Three-tier architecture follows a tree-based architecture composed of three layers: the access, the aggregation, and the core layer. The data center core layer provides a fabric for high-speed packet switching between multiple aggregation Pods/modules and responsible for connecting the data center network with the outside network. The aggregation layer is responsible for connecting the access switches or access switches with the upper layer or core layer network and finally the access layer switches are responsible for connecting multiple servers.

As stated in previous section the predominant volume of traffic in the data center is comprised in the east-west traffic category and usually depend on the kind and mix of applications such user-facing applications (e.g. web services) typically exchange data with users and thus generate north-south communication to response the user requests [44]. As the Cisco forecasting statistics shows that the east-west traffic is growing on a larger scale as the increasing demands of applications within the data centers. Thus, understanding and studying the behavior of these types of traffic (east-west) needs high attention and must be included as the first requirement while designing and deploying data center network.

To simulate the Three-tier DCN architecture, the interconnection of network devices is arranged in four layers i.e. the core layer, the aggregation layer, the access layer and the lower layer where the servers are installed. In order to evaluate the performance of the architecture different oversubscription ratios were used between the core layer switches and the aggregation layer switches, the aggregation layer switches, and the access layer switches. In the simulation the following real-world scenario were considered to build the simulated Three-tier data center network architecture.

A) Core layer and Aggregation layer

In the Three-tier data center network architecture, the core layer devices handle the main task of the data center and they are the gateways for massive number of traffic that flows from the clients outside the data center and servers/clients in the data center. Due to these reasons installing one core layer device for one service or multiple services can lead to different disadvantages because the failure rate of these devices are high. To come up with such challenge, active/standby technology features are used. Thus, in the real data center that operates large numbers of services, data center operators install two core layer devices for active standby purpose depending on the services the offer.

The aggregation layer is the middle layer where different types of security devices such as firewalls, IPS(Intrusion Prevention Systems), IDS(Intrusion Detection Systems), Proxy servers and Load-balancers are installed. In this layer different security configurations such as ACLs(Access Control Lists)), PMR (Policy Map Routing) algorithms, QoS (Quality of Service) are also configured and enabled. So, the of aggregation layer switches also plays an important role in the data center network and that is why active/standby scenario is also implemented in this layer for each type of services.

In the Three-tier DCN architecture, the devices at the core and aggregation layer are installed in active and standby scenario, when one core or aggregation devices fails the second automatically handle the task without any system interruption if the active standby configuration efficiently done. However, the standby device not only functions when the active devices stop working but also shares the task because installing such high-capacity devices for redundancy purpose is wastage and requires high capital investments. This helps to install large number of services in the data center until the capacity of the core and aggregation layer becomes optimum. So, in the simulation the core layer and aggregation layer devices are installed by considering such scenarios in order to mimic the rea data center network architecture.

B) Modules/pod

In networking terminology, a module or a point of delivery, pod is a design pattern that consists a collection of compute, storage, and application components that conform to deliver a specific type of service and share the same failure domain [48]. This module/pod used as a standard operating footprint for the service deployed inside it and it has significant advantages specially in data center networking. The central wiring between switches in adjacent layers are relatively easy to manage, and the network can be expanded to arbitrary size by adding stages and these network modules/pods are usually adopted as building blocks to further ease network scalability, deployment and management [49].

In the simulation process, the Three-tier data center network architecture was simulated and evaluated with different number of modules/pods (3, 4, 5,6,7,8 and 9 module) in order to observe the impact of the underlying network connectivity on the network performance. The sample network architecture used during the simulation are depicted in Figure 3-3 and 3-4 which are built with 4-module/pod and 8-module/pod network respectively and in the result and analysis part only these two network types (4-module/pod and 8-module/pod) are discussed. The ToR or access layer switches act as a bridge and used to interconnect the end hosts with the aggregation layer switches and the number of ToR switches and the number of hosts under each ToR switch are varied in order to analyze the differences on the data center network performance.

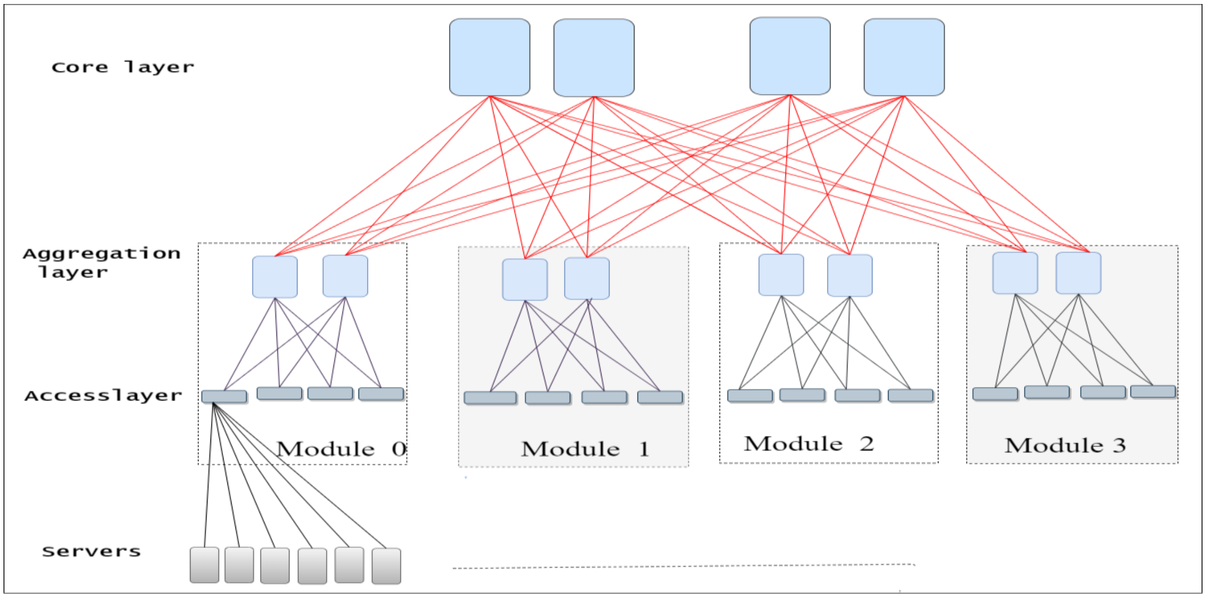


Figure 3- 3: Three-tier DCN architecture with 4-modules/pods used in the simulations

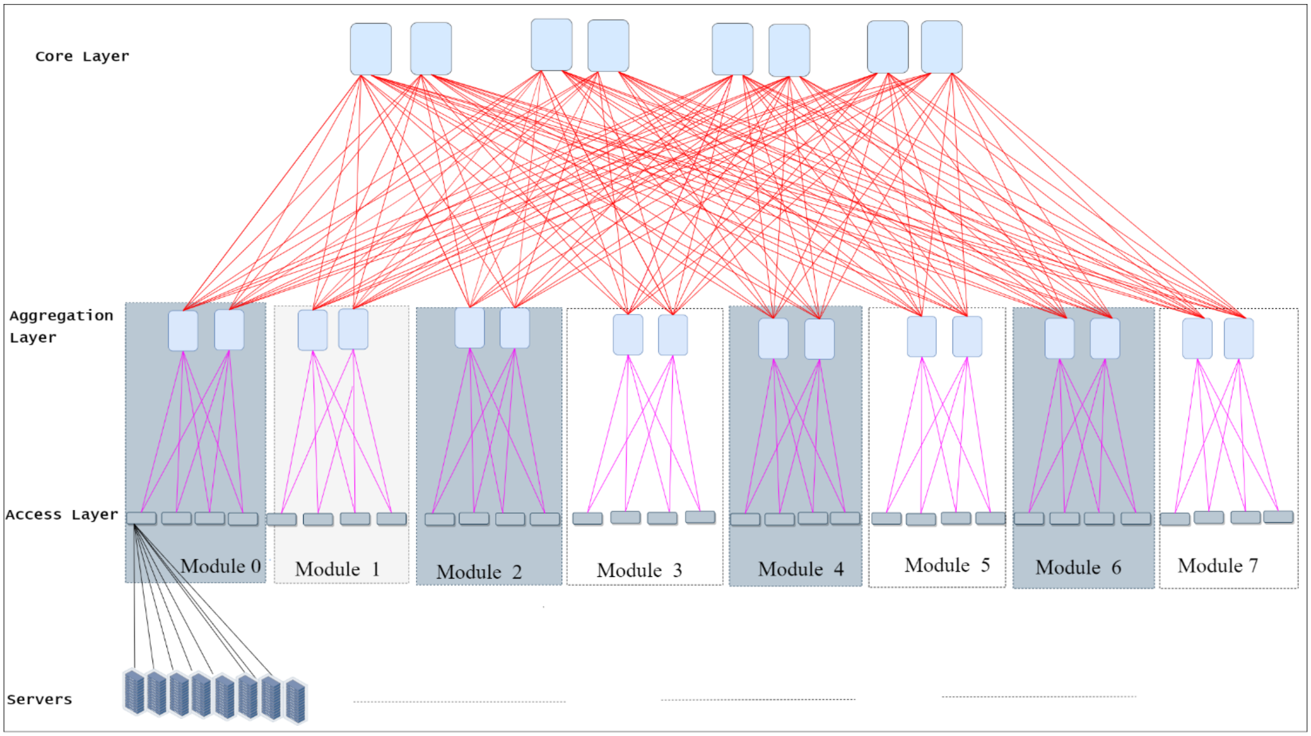


Figure 3- 4: Three-tier DCN architecture with 8-modules/pods used in the simulations

In the Three-tier DCN architecture, the switches/router at the core layer are installed in active and standby scenario, when one core switch fails the second automatically handle the task without any system interruption if the active standby configuration efficiently done. However, the standby device not only functions when the active devices stop working but also shares the task because installing such high-capacity devices for redundancy purpose is wastage and requires high capital investments.

As stated earlier, the main purpose of simulation is predicting the network behavior in the real-world scenario using software. If the number of devices at the core layer are small, then the oversubscription ratio at the aggregation layer will be high and the number of end hosts in the access layer is limited because this high oversubscription ratio at the aggregation layer creates bandwidth bottlenecks. By considering these scenarios the simulated network architecture is designed to have multiple core devices at the aggregation layer. So, the network can support large number of hosts which is similar to the real data center network environment.

3.2.3 Simulation parameters

To be able to properly simulate the behavior of Three-tier data center network as a real data center large-scale network, the following are the main parameter setting that were varied during the simulation process.

A) Link rates

To make the simulation more realistic, the data rates or the interface bandwidth used to connect the devices to the upper layer and lower layer is set to Gigabit Ethernet links.

B) Oversubscription ratio

In the data center the network oversubscription ratio should be ideally 1:1, however achieving this oversubscription requires more numbers switches with high bandwidth and cables. Since the lots of switches and cabling involving extra cost and often such high bandwidth remains under-utilized. Thus, data center networks try to reduce the number of switches depending on traffic since all devices do not communicate simultaneously. So, by considering this scenario, the performance of the Three-tier architectures is evaluated by varying the degree of oversubscription ratio at the aggregation layer and access layer.

In the simulation, two scenarios are used , in the first scenario is performed by increasing the number of ToR/access switches and the number of end hosts attached to each access switch so as to increase the oversubscription ratio at the aggregation layer and access layer. Ideally, if large number of servers installed, multiple number of core switches are needed to handle the tasks efficiently and in the 4-module network simulation the minimum number of access switches are four to get 1:1 oversubscription ratio but to make the network is blocking both at the aggregation and access layer, the number of access switches and the number of end hosts in each access switch were set to more than four. In the second scenario, the total number of end hosts installed at the data center are fixed but the number of access switches were different and the oversubscription ratio at the aggregation and access layer were varying.

C) Routing protocol

The routing protocol used during the simulation is Nix-vector routing protocol (the ns-3 model) and it is a simulation specific routing protocol intended for large-scale network topologies. The on-demand nature of this protocol as well as the low-memory footprint of the nix-vector provides improved performance in terms of memory usage and simulation run time when dealing with a large number of nodes [45].

D) Traffic flow

The traffic characteristics used in the simulation is regular traffic flow where end hosts/servers are just generating random traffic and either one server is sending traffic to another server, one server is receiving from multiple servers or one server is sending to multiple servers which is the same as a real data center network environment. Generally, the simulation parameters used in this paper to simulate the Three-tier data center network architecture are described in the following table.

Table 3- 1: Simulation parameters for 4-module Three-tier DCN architecture

Parameters	Value
Number of Core switches	4
Number of Aggregation switches	8
Numbers of Modules/Pods	4
Number hosts on each edge Switch	4 - 40
Number of ToR switches in a Module	8 - 32
Total number of hosts	48 - 4608
Communication Pattern	Random selection of two hosts and sending data between them
Routing Protocol	Nix-vector routing

For simulating the Three-tier data center network architecture build with 8-module network, the simulation parameters were the same with the 4-module network except the number of core and aggregation layer switches and the total number of end hosts.

Table 3- 2: Simulation parameters for 8-module Three-tier DCN architecture

Parameters	Value
Number of Core switches	8
Number of Aggregation switches	16
Numbers of Modules/Pods	8
Number hosts on each edge Switch	16 - 48
Number of ToR switches in a Module	12 - 24
Total number of hosts	96 - 4608
Communication Pattern	Random selection of two hosts and sending data between them
Routing Protocol	Nix-vector routing

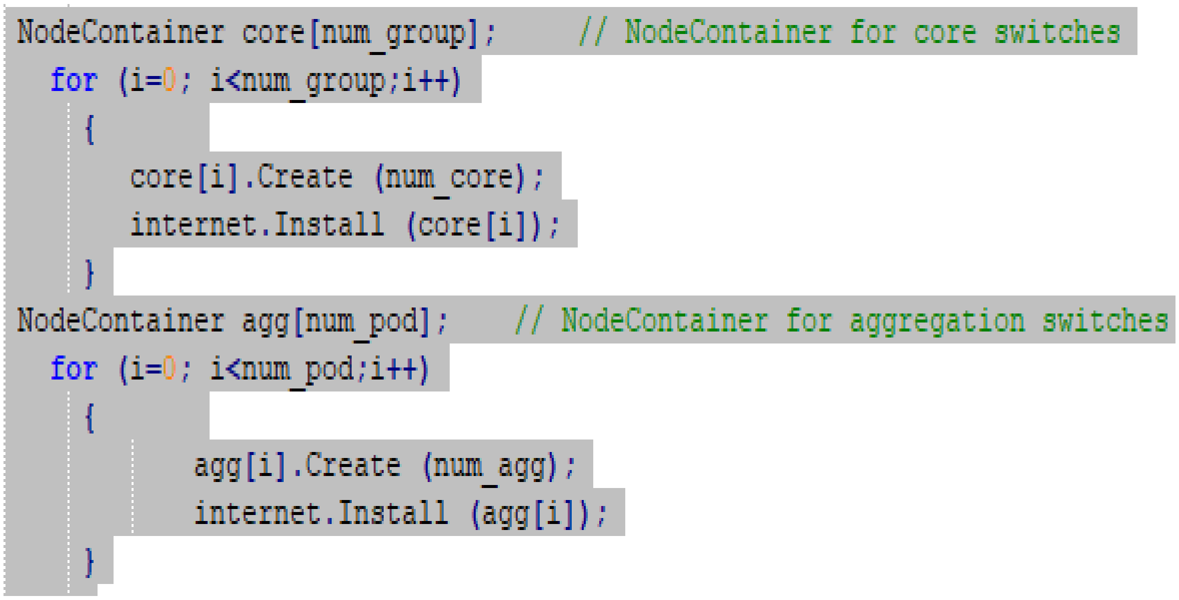
3.3 Implementation detail

3.3.1 Building Topologies

In the Internet world, a computing device that connects to a network is called a host and because ns-3 is a network simulator, not specifically an Internet simulator it does not use the term host, instead it uses a more generic term call node [46]. In ns-3 network tenants i.e. switches and servers are generally called as nodes and nodes are created in node containers.

In the Three-tier DCN architecture, each device in the same layer are created with one node container i.e. core layer switches are created in one node container, aggregation layer switches are created in one node container, ToR switches are created in one node container and hosts are created in one node container. This way of creating those different node containers for each layer of network devices is important to assign the same attributes on the nodes such as link data rates, delay, etc. The following sample line of code depicted in Figure 3-5 was used to create the ns-3 node objects that represents the core layer and aggregation layer switches.

The ns-3 NodeContainer class is a topology helper that provides a convenient way to create, manage and access any node objects that we create in order to run a simulation.



```
NodeContainer core[num_group];     // NodeContainer for core switches
  for (i=0; i<num_group;i++)
    {
      core[i].Create (num_core);
      internet.Install (core[i]);
    }
NodeContainer agg[num_pod];    // NodeContainer for aggregation switches
  for (i=0; i<num_pod;i++)
    {
          agg[i].Create (num_agg);
          internet.Install (agg[i]);
    }
```

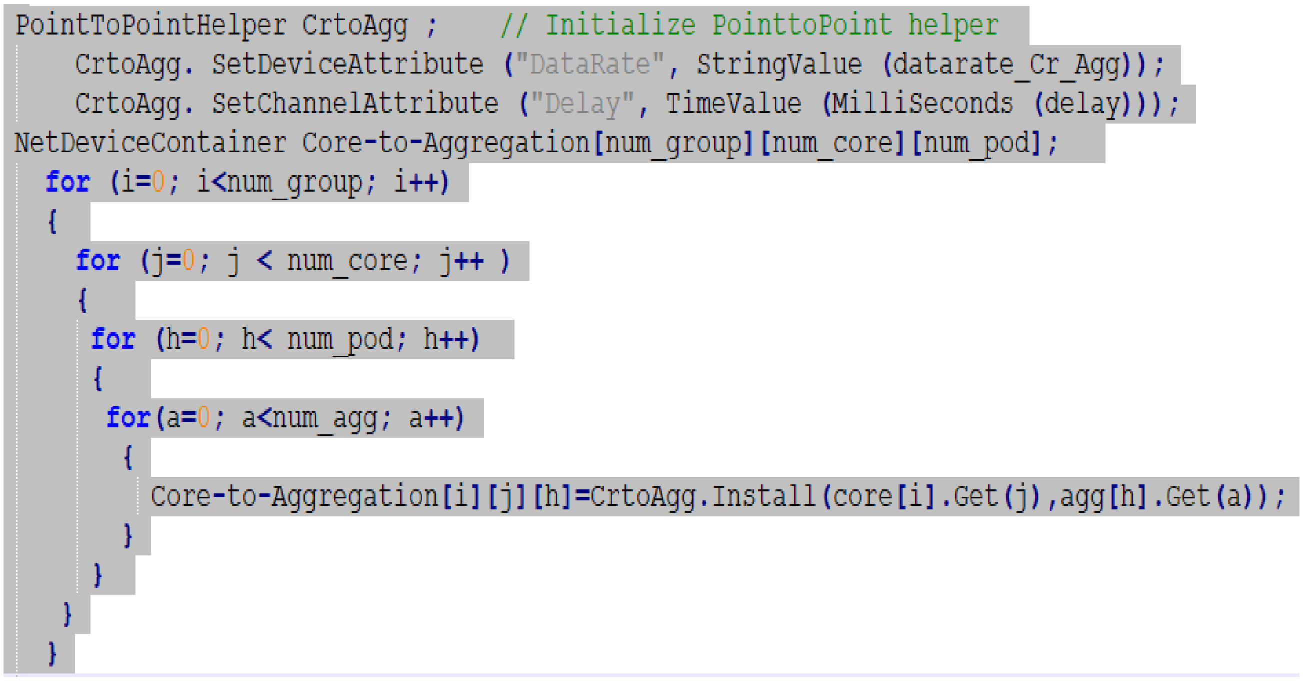
Figure 3- 5: ns-3 lines of code used to create node objects

3.3.2 Connecting Nodes

After creating node, the preceding step in constructing a topology is setting up a point-to-point link between each node to create a link between core switches witch aggregation layer switches, aggregation switches with ToR layer switches and hosts to ToR layer switches. In the real network connectivity, to connect a network router/switch/PC with another router/switch, a network cable and a NIC (network interface card) is needed and also network driver software is a must.

In ns-3, generally in Unix (or Linux) operating system, these network hardware and NICs (network interface card) are categorized as devices and these devices are controlled device drivers. The combination of these devices and device drivers are called net devices. In the ns-3 environment, the net device abstraction is represented in C++ by the class called NetDevice. The ns-3 NetDevice class provides methods for managing connections to Node and Channel objects.

In the previous step to create the nodes, the ns-3 NodeContainer topology helper object was used and likewise the ns-3 NetDeviceContainer function is used to hold the network devices that form point-to-point channel. The NetDeviceContainer topology helper is used to connect one node with another node which is found at the above layer or below layers network. After creating a NetDeviceContainer that contains a neighbor network node, the next action is connecting the nodes and configuring attributes on the channel using the ns-3 PointToPointHelper topology helper object. Some of the attributes which the ns-3 PointToPointHelper object installs are the queening delay and the data rate/bandwidth of the interfaces.

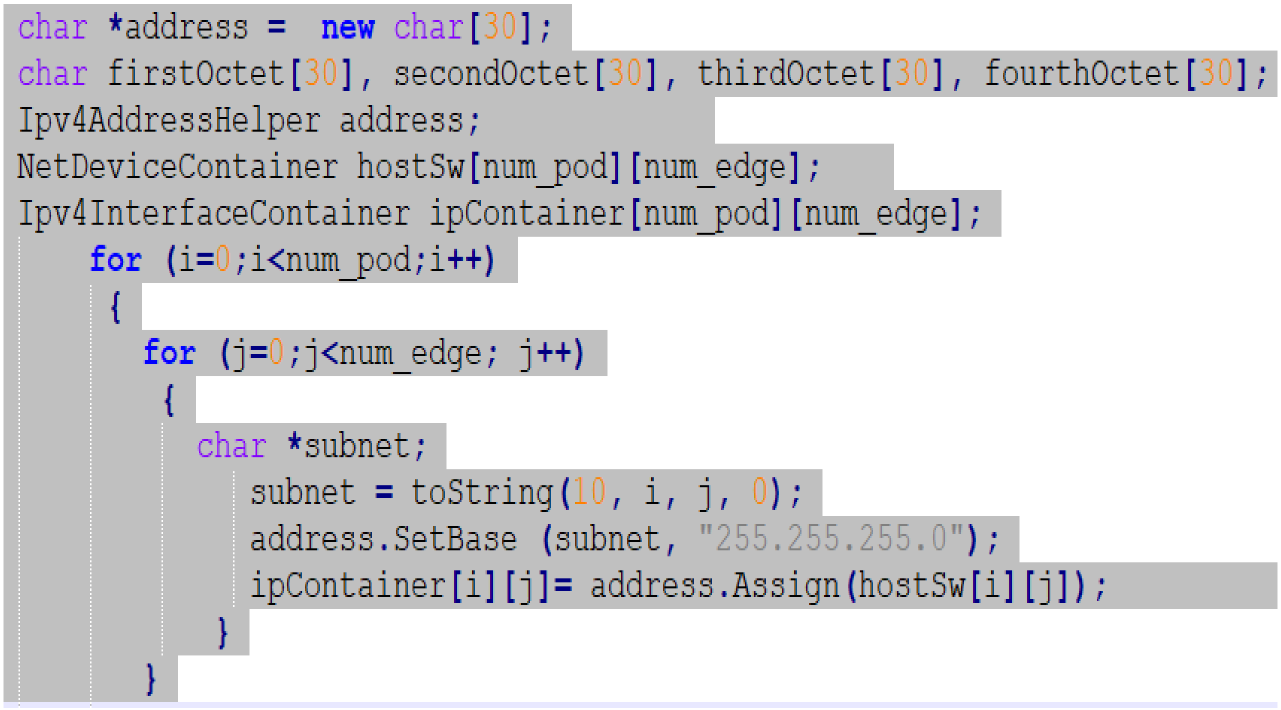


```
PointToPointHelper CrtoAgg ;    // Initialize PointtoPoint helper
    CrtoAgg. SetDeviceAttribute ("DataRate", StringValue (datarate_Cr_Agg));
    CrtoAgg. SetChannelAttribute ("Delay", TimeValue (MilliSeconds (delay)));
NetDeviceContainer Core-to-Aggregation[num_group][num_core][num_pod];
  for (i=0; i<num_group; i++)
  {
    for (j=0; j < num_core; j++ )
    {
     for (h=0; h< num_pod; h++)
     {
      for(a=0; a<num_agg; a++)
       {
        Core-to-Aggregation[i][j][h]=CrtoAgg.Install(core[i].Get(j),agg[h].Get(a));
       }
     }
   }
  }
```

Figure 3- 6: ns-3 lines of code used to connect core and aggregation layer devices

3.3.3 Assigning IP addresses

After the point-to-point link is installed between each NetDevices, the next step is assigning IP address on each node interface and to do this the ns-3 Ipv4AddressHelper topology helper is used. This topology helper not only used to set the base IP address and network mask but also manage the allocation of IP addresses on each NetDevices. The below sample codes are used to create address pools with subnet mask of 24 (255.255.255.0) for each access switch. The access switches connected to aggregation switch in one module/pod are in different network address.



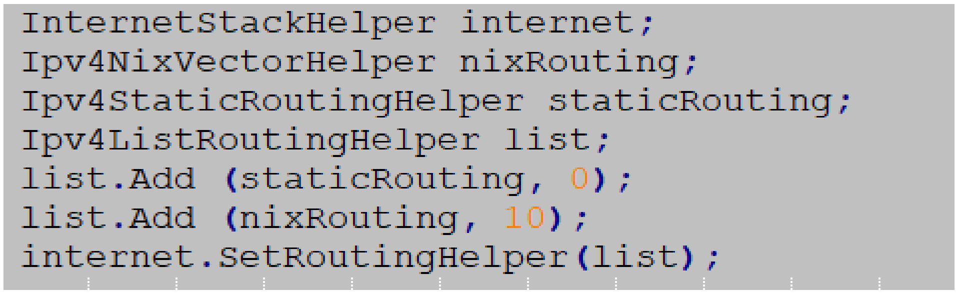
```
char *address =  new char[30];
char firstOctet[30], secondOctet[30], thirdOctet[30], fourthOctet[30];
Ipv4AddressHelper address;
NetDeviceContainer hostSw[num_pod][num_edge];
Ipv4InterfaceContainer ipContainer[num_pod][num_edge];
    for (i=0;i<num_pod;i++)
     {
       for (j=0;j<num_edge; j++)
        {
          char *subnet;
             subnet = toString(10, i, j, 0);
             address.SetBase (subnet, "255.255.255.0");
             ipContainer[i][j]= address.Assign(hostSw[i][j]);
           }
       }
```

Figure 3- 7: ns-3 lines of code used to assign IP address for access switches

The first line of code initializes an address helper object, the second line of code declares the network address and the third line is used to tell the address helper object to allocate node interface IP address from this pool. After creating the pool of addresses for the nodes, anther ns-3 topology helper called Ipv4Interface is used to associate net devices and IP addresses. Finally, The Ipv4InterfaceContainer topology helper is used to installs internet stacks on two nodes and creates interfaces between them. This topology helper finally assigns IP addresses from the network pool by checking the free IP address from the pool.

3.3.4 Configure Routing algorithm

As stated in section 4.2.3(C), the routing protocols used in this simulation is ns-3's Nix-vector routing protocol and to enable this routing protocol on the network the following lines of code were used. The InternetStackHelper is used to initialize Internet Stack and Routing Protocols.

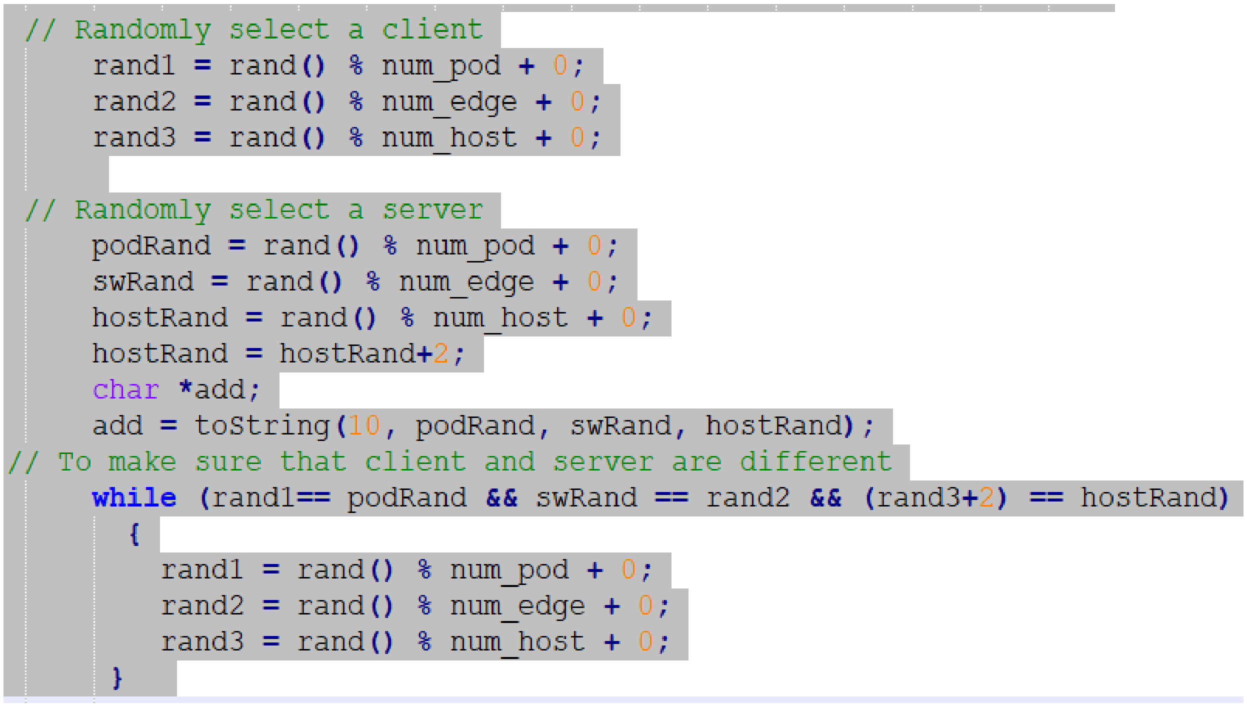


```
InternetStackHelper internet;
Ipv4NixVectorHelper nixRouting;
Ipv4StaticRoutingHelper staticRouting;
Ipv4ListRoutingHelper list;
list.Add (staticRouting, 0);
list.Add (nixRouting, 10);
internet.SetRoutingHelper(list);
```

Figure 3- 8: ns-3 lines of code used to configure routing protocol on the network

3.3.5 Generate traffic for the simulation

In the previous sub-sections constructing the DCN topology is completed, a point-to-point link is created, assigning IP address and enabling routing protocol is finished. The preceding task is generating traffic between the nodes and to do this sender (client) and receiver (server) nodes are selected randomly using a random generator function. This function randomly selects addresses for pods, access switches and hosts and add these addresses as IP address of the sender and receiver. A while loop was used to make sure the sender host is not the same as the receiver host and the following lines of code describes how to randomly select a sender and receiver host from the network.

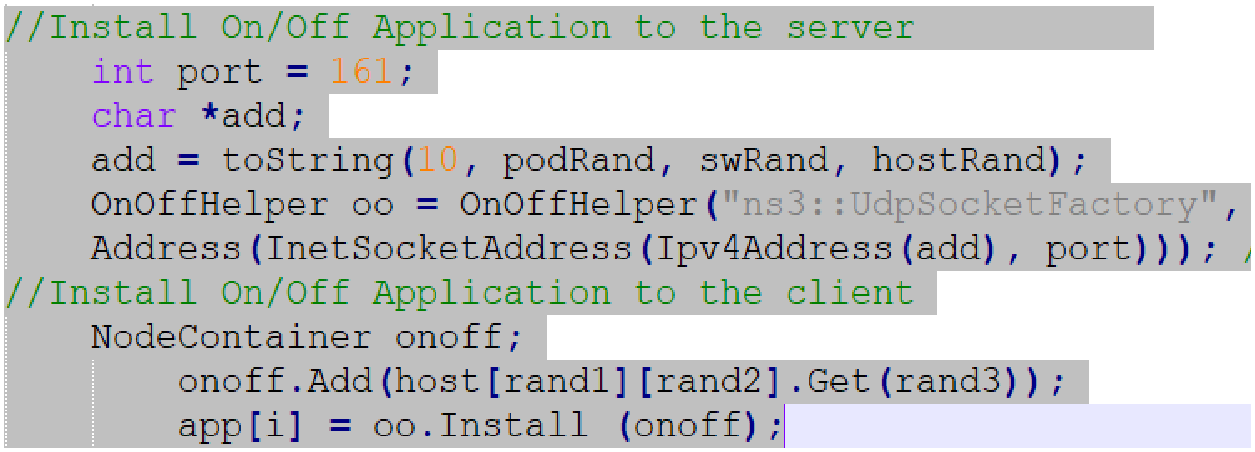


```
 // Randomly select a client
     rand1 = rand() % num_pod + 0;
     rand2 = rand() % num_edge + 0;
     rand3 = rand() % num_host + 0;

 // Randomly select a server
     podRand = rand() % num_pod + 0;
     swRand = rand() % num_edge + 0;
     hostRand = rand() % num_host + 0;
     hostRand = hostRand+2;
     char *add;
     add = toString(10, podRand, swRand, hostRand);
// To make sure that client and server are different
     while (rand1== podRand && swRand == rand2 && (rand3+2) == hostRand)
       {
         rand1 = rand() % num_pod + 0;
         rand2 = rand() % num_edge + 0;
         rand3 = rand() % num_host + 0;
      }
```

Figure 3- 9: ns-3 lines of code used to create sender and receiver hosts randomly

To generate traffic the ns-3 simulator uses Application class which is the core abstraction of the ns-3 simulator. There are different traffic patterns in the ns-3 simulation such as FTP traffic, CBR Traffic and OnOff traffic. The ns-3 OnOff application is used to generate a unicast traffic between two nodes and as other helpers install the attributes on the declared objects, this application also installs the necessary attributes on the node using an OnOffHelper.



```
//Install On/Off Application to the server
    int port = 161;
    char *add;
    add = toString(10, podRand, swRand, hostRand);
    OnOffHelper oo = OnOffHelper("ns3::UdpSocketFactory",
    Address(InetSocketAddress(Ipv4Address(add), port))); /
//Install On/Off Application to the client
    NodeContainer onoff;
        onoff.Add(host[rand1][rand2].Get(rand3));
        app[i] = oo.Install (onoff);
```

Figure 3- 10: ns-3 lines of code used to install OnOff applications

CHAPTER FOUR: RESULTS AND ANALYSIS

4.1 DCN Performance Evaluation

In this chapter simulation results obtained from the simulator are discussed and analyzed and in order to evaluate the performance of the considered data center architectural the following important network design and performance metrics or measurements are used. To evaluate the Three-tier DCN architecture performance different simulation parameters are used. The parameters used in the simulation process were described in Table 3-1 and 3-2 and simulations were performed by varying the value of those parameters and setting the oversubscription ratio of the links at different layer is also considered as key performance measurement of the DCN architecture data center networks.

In this study mainly, the end-to-end packet delay(the time required to transmit a packet along its entire path created by an application, handed over to the OS, passed to a network card (NIC), encoded, transmitted over a physical medium (copper, fiber, air), received by an intermediate device (switch, router), analyzed and re-transmitted over another medium) is measured. The capacity of the DCN architecture to transfer data from one communication end host to anther communication end host across the network i.e. average throughput is also measured and analyzed.

4.1.1 Average Packet Delay

Network delay is an important performance characteristic of data center network that is used to measures the amount of time it takes for a packet to be transmitted from one communication host in a network (sender) to another communication host (destination) over the network. Network Latency and delay are similar terms and often used interchangeably, however there is a difference between them. Latency refer to the amount of time it takes for a packet to be transmitted from source host to destination host which is the same as one-way delay and the time it takes to return back to the sender i.e. the Round-trip-time (RTT). But this Round-trip of latency or packet delivery time does not include the packet processing time at the destination host like routing computation time, rule filtering time if there are some access list configurations, policy-based routing configurations and etc.

Moreover, the end-to-end delay is a collection of four different delay components. The first component of packet delay is the processing delay which includes the packet analyzing time, routing table execution time and the packet forwarding time of the node. Queuing delay which refers to how long the packet is queued until it is being sent is another component of packet delay. The third packet delay component is transmission delay and it is the time taken for the interface to put a data packet on the transmission link. Finally, the propagation delay which indicates the time taken for one bit to travel from sender to receiver end of the link. The average packet delay in the network is calculated using the following equation [11].

$$D_{avg} = \frac{1}{n}\sum_{i=1}^{n} d_i$$

where D_{avg} refers to the average packet delay, n is the total number of packets received in the network and d_i is the delay of packet i.

4.1.2 Average Throughput

Throughput is defined as the amount of material or items passing through a system or process. Relating this to networking, the materials are referred to as “packets” while the system they are passing through is a particular “link”, physical or virtual. It is also a term used to describe the capacity of a system to transfer data. Since the demand for data exchange in DCNs is extremely large compared with other networks, the first design goal is to maximize the throughput. The amount of bandwidth allocated to different types of packets affect throughput [42]. The average throughput of the network can be calculated using the following equation [11].

$$T_{avg} = \frac{\sum_{i=1}^{n} p_i x\, \delta_i}{\sum_{i=1}^{n} d_i}$$

Where T_{avg} is the average throughput in the network , p_i is the i^{th} received packet , $\delta_i$is the size of packet i in bits and d_i as the delay of packet i and n is the total number of packets received in the network.

4.2 Simulation results for average packet delay

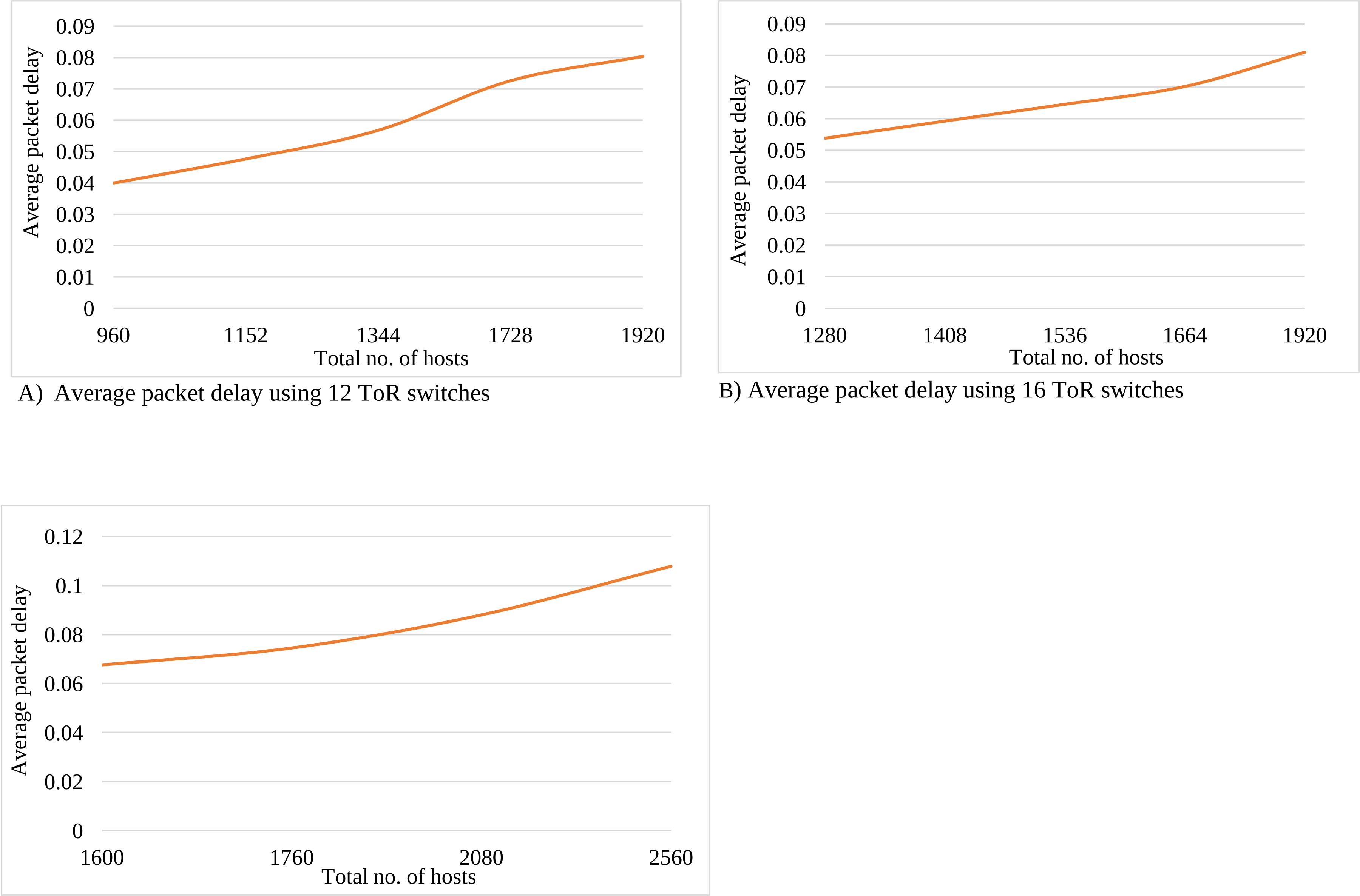
A) Case 1 : Using 4-module Three-tier DCN architecture

The network topology used in the first simulation scenario is the Three-tier architecture depicted in Figure 3-3 which has 4 core and 8 aggregation switches. The data rate used between each core layer switch and aggregation layer switch is 10Gbps, between each aggregation switches and ToR/access layer switch is also 10Gbps and to connect each host to ToR layer switches 1Gbps interface is used. In the simulation, the total number of hosts used in the entire network and the traffic flows generated by the hosts were the same and the total oversubscription ratio at the core layer was also the same. The only difference is the number of access switches used in the network and the arrangement of hosts in each access switch under the pod/modules. Table 4-1 describes sample output results for the average packet delay of Three-tier architecture build with 4-module network and incrementing the network size.

Table 4- 1: Average packet delay of Three-tier DCN 4-module

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of hosts	Average packet delay(sec)
12	20	3:1	1:1	960	0.03994
	24	3:1	1.2:1	1152	0.04765
	28	3:1	1.4:1	1344	0.05677
	36	3:1	1.8:1	1728	0.07258
	40	3:1	2:1	1920	0.08035
16	20	4:1	1:1	1280	0.05379
	22	4:1	1.1:1	1408	0.05923
	24	4:1	1.2:1	1536	0.06455
	26	4:1	1.3:1	1664	0.07017
	30	4:1	1.5:1	1920	0.081
20	20	5:1	1:1	1600	0.06759
	22	5:1	1.1:1	1760	0.07451
	26	5:1	1.3:1	2080	0.08799
	32	5:1	1.6:1	2560	0.10784

Figure 4-1 represents the average packet delay results obtained after simulating Three-tier DCN architecture with varying the number of access/ToR layer switches and the total number of hosts in the entire network.



A) Average packet delay using 12 ToR switches

B) Average packet delay using 16 ToR switches

C) Average packet delay using 20 ToR switches

Figure 4- 1: Average packet delay of 4-module Three-tier architecture

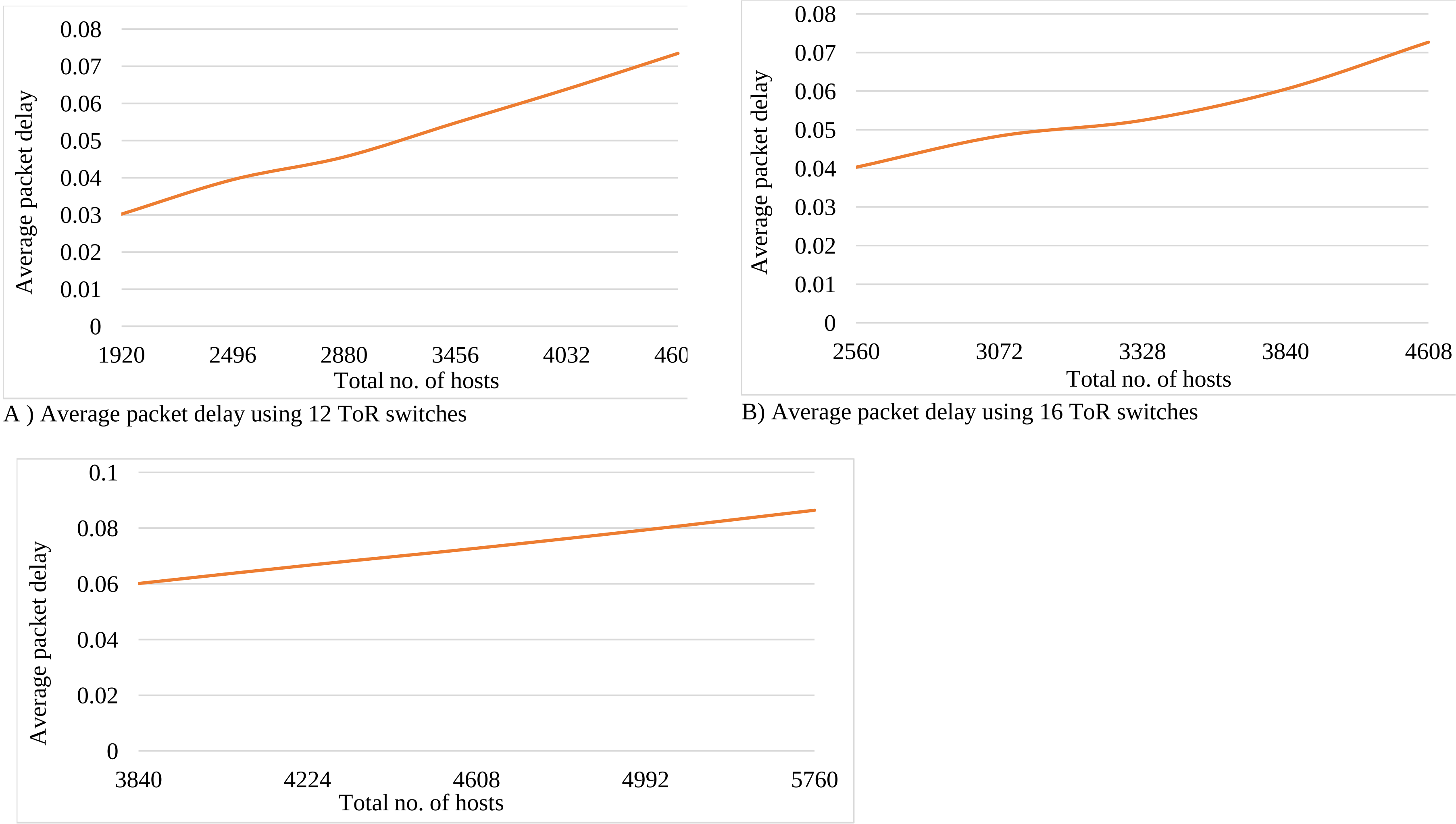
As the simulation results shows the average packet delay of the Three-tier data center network architecture is increased linearly in all types of architecture as the total number of end hosts installed in the entire network are increased. When large number of end hosts attached on the ToR/access switches, the oversubscription ratio of the layer is increased, and this high degree of ratio creates congestions which changes the state of the link to blocking as the traffic flow increases.

B) Case 2 : Using 8-module Three-tier DCN architecture

The network topology used in the first simulation scenario is the conventional DCN architecture depicted in Figure 3-4 which has 8 core and 16 aggregation switches. The data rate used between each core layer switch and aggregation layer switch is 10Gbps, between each aggregation switches and ToR/access layer switch is also 10Gbps and to connect each host to ToR layer switches 1Gbps interface is used. Table 4-2 shows the output data for average packet delay in 8-module Three-tier architecture.

Table 4- 2: Average packet delay of Three-tier DCN 8-module

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of hosts	Average packet delay(sec)
12	20	1.5:1	1:01	1920	0.0301977
	26	1.5:1	1.3:1	2496	0.0394719
	30	1.5:1	1.5:1	2880	0.0455692
	36	1.5:1	1.8:1	3456	0.0547425
	42	1.5:1	2.1:1	4032	0.0638208
	48	1.5:1	2.4:1	4608	0.073461
16	20	2:01	1:01	2560	0.0402914
	24	2:01	1.2:1	3072	0.048372
	26	2:01	1.3:1	3328	0.052434
	30	2:01	1.5:1	3840	0.0604933
	36	2:01	1.8:1	4608	0.0726806
24	20	3:01	1:01	3840	0.0600963
	22	3:01	1.1:1	4224	0.0666251
	24	3:01	1.2:1	4608	0.0727566
	26	3:01	1.3:1	4992	0.0793473
	30	3:01	1.5:1	5760	0.0863972



A) Average packet delay using 12 ToR switches

B) Average packet delay using 16 ToR switches

C) Average packet delay using 16 ToR switches

Figure 4- 2: Average packet delay of 8-module Three-tier architecture

Figure 4-2 represent the average packet delay results obtained after simulating Three-tier DCN architecture with varying the number of access/ToR layer switches and the total number of hosts in the entire network. As observed in the above figures the average packet delay is increased linearly in all types of the topologies (topologies build using different number of ToR/access layer switches) as the total number of hosts in the entire network increased. The main reason of this incremental growth is network congestion due to small interface data rate/bandwidth, packet processing time on each layer switches as the number of packets arriving on each switches increase, drop packet retransmission time due to congested link or interfaces, routing table computation on core and aggregation layer switches, etc.

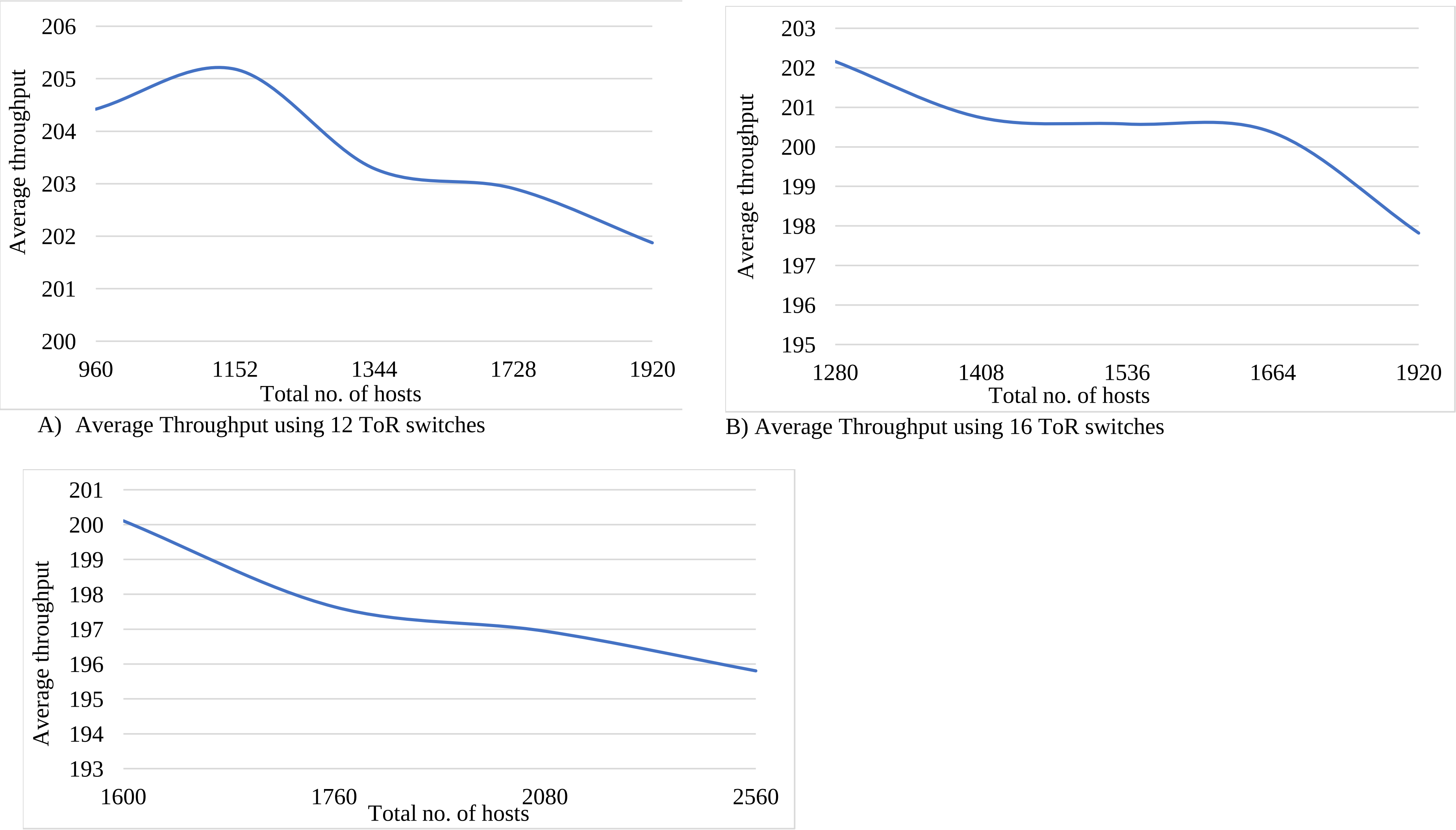
4.3 Simulation results for average throughput

A) Case 1 : Using 4-module Three-tier DCN architecture

To evaluate the average network throughput of the Three-tier DCN architecture depicted in figure 3-3 network architecture was used in the simulation. The link rate used between each core layer and aggregation layer switch is 10Gbps, between each aggregation layer switches and access layer switch is also 10Gbps and to connect each host to ToR layer switches 1Gbps interface is used. The below Table 4-3 shows sample output results for the average network throughout of Three-tier architecture build with 4-module network.

Table 4- 3: Average throughput result of 4-module Three-tier DCN

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of hosts	Average Throughput (Mbps)
12	20	3:1	1:1	960	204.421
	24	3:1	1.2:1	1152	205.183
	28	3:1	1.4:1	1344	203.29
	36	3:1	1.8:1	1728	202.911
	40	3:1	2:1	1920	201.875
16	20	4:1	1:1	1280	202.16
	22	4:1	1.1:1	1408	200.739
	24	4:1	1.2:1	1536	200.578
	26	4:1	1.3:1	1664	200.363
	30	4:1	1.5:1	1920	197.819
20	20	5:1	1:1	1600	200.108
	22	5:1	1.1:1	1760	197.642
	26	5:1	1.3:1	2080	196.943
	32	5:1	1.6:1	2560	195.805



A) Average Throughput using 12 ToR switches

B) Average Throughput using 16 ToR switches

C) Average Throughput using 20 ToR switches

Figure 4- 3: Average throughput of 4-module Three-tier architecture

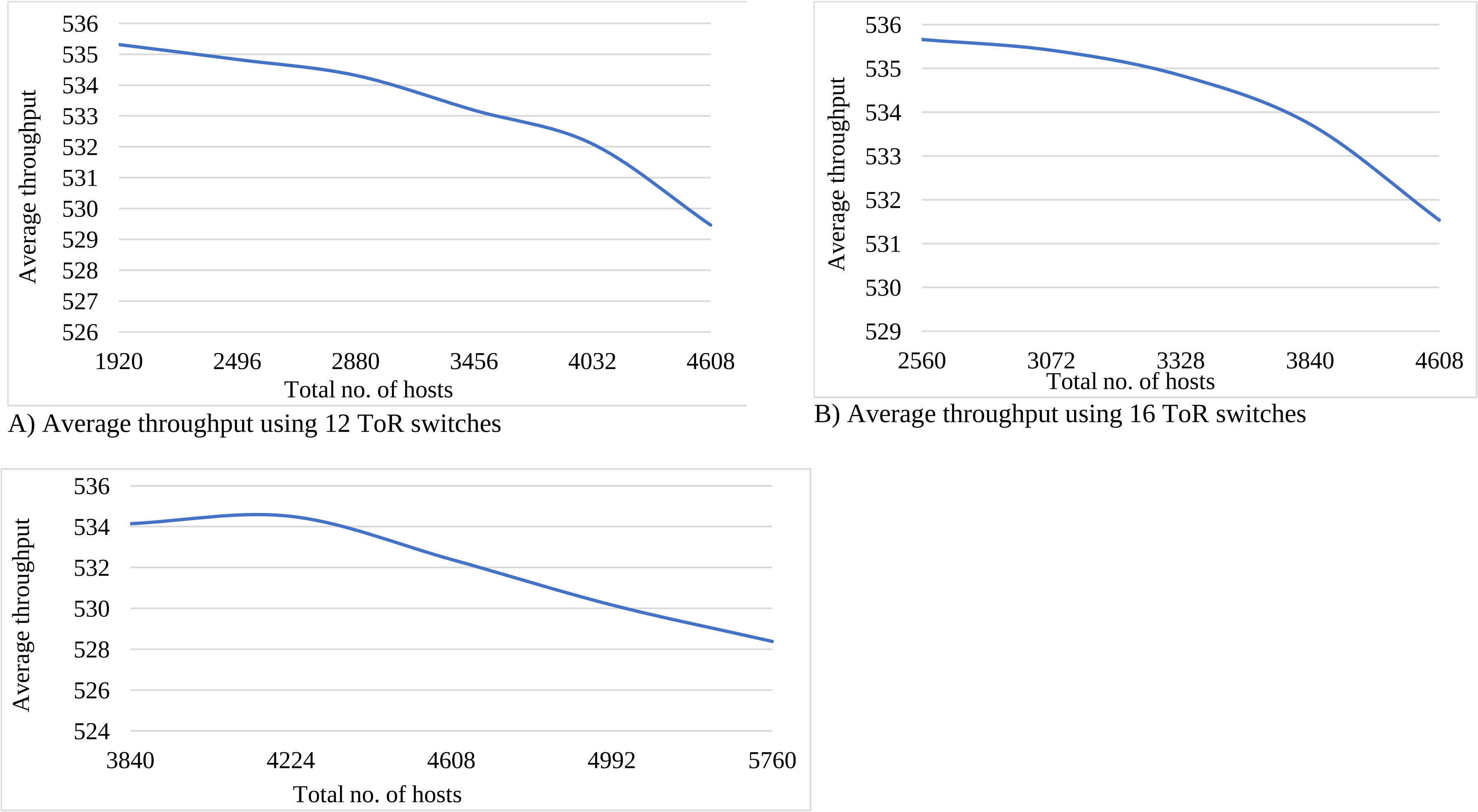
As Figure 4-3 describe the simulation results of the average packet throughput for the Three-tier data center network architecture build with 4-module network and as the graphs depicts the average network throughput is decreased when the total number of end hosts increased. In the previous section the average packet delay is increase as the total number of end hosts in the entire network increase. This increased value of average packet delay leads to decreased the average throughput of the network since packet delay and network throughput are inversely proportional and to produce high network throughput, the average packet delay of the network should have a small values and to have low average packet delay, the network oversubscription ratio of links should have a minimum degree of rate in order to decrease the network congestion and blocking points of the layer.

B) Case 2 : Using 8-module Three-tier DCN architecture

For the average network throughput evaluation, the Three-tier DCN architecture depicted in Figure 3-4 network architecture was used in the simulation. The link rate used between each core layer and aggregation layer switch is 10Gbps, between each aggregation layer switches and access layer switch is also 10Gbps and to connect each host to ToR layer switches 1Gbps interface is used.

Table 4- 4: Average throughput of Three-tier DCN 8-module

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of hosts	Average Throughput (Mbps)
12	20	1.5:1	1:1	1920	535.314
	26	1.5:1	1.3:1	2496	534.831
	30	1.5:1	1.5:1	2880	534.317
	36	1.5:1	1.8:1	3456	533.187
	42	1.5:1	2.1:1	4032	532.096
	48	1.5:1	2.4:1	4608	529.463
16	20	2:1	1:1	2560	535.66
	24	2:1	1.2:1	3072	535.415
	26	2:1	1.3:1	3328	534.841
	30	2:1	1.5:1	3840	533.729
	36	2:1	1.8:1	4608	531.536
24	20	3:1	1:1	3840	534.146
	22	3:1	1.1:1	4224	534.504
	24	3:1	1.2:1	4608	532.394
	26	3:1	1.3:1	4992	530.166
	30	3:1	1.5:1	5760	528.382



A) Average throughput using 12 ToR switches

B) Average throughput using 16 ToR switches

C) Average throughput using 24 ToR switches

Figure 4- 4: Average throughput of 8-module Three-tier architecture

As observed in Figure 4-4 , the average network throughput is decreased in all types of network topologies. This is due to the linearly increased average packet delay of the network that is caused different factors such as link congestion, error recovery and packet retransmission time and packet processing capabilities of devices involved in the transmission. As stated earlier, the main cause of packet delay is the link congestion due to high oversubscription ratio on the access layer switches as the number of hosts connected on each access switch increases. Generally, as the figures depicts the average packet delay is inversely proportional with average network throughput i.e. when the average packet delay increase, the average network throughput decreases and vice versa.

4.4 Effect of oversubscription ratio

In order to evaluate the performance of the Three-tier architecture with two major performance metrics(average packet delay and average throughput) by varying the bandwidth oversubscription ratio, a total of 2Gbps and 20Gbps uplinks from access layer to aggregation layer for 4-module and 8-module topology respectively are used in the simulation. In simulating both the 4-module and 8-module DCN topologies, different simulation scenarios are used. In the first scenario, the total number of flows or total number of applications in the entire network were equal to the total number of the hosts in the entire network and the results are depicted on the previous figures. As observed in the above figures the average packet delay is increased linearly due to the oversubscription ratio increased at the access layer as the number of hosts in the access layer increased and the average throughput is decrease.

4.4.1 Effect of oversubscription ratio on average packet delay

A) **Case 1 : Using 4-module Three-tier DCN architecture**

In the second scenario, simulation is done by increasing the total number of traffic flows and setting the same oversubscription ratio at the access layer but different oversubscription ratio at aggregation layer and keep the total number of hosts in the entire network the same. Table 4-5 and Figure 4-5 describe the average packet delay results obtained from the simulation.

Table 4- 5: Average packet delay results for 256 total number of hosts

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of flows	Average packet delay (sec)
8	8	2:1	4:1	256	0.0167237
	8	2:1	4:1	512	0.0218766
	8	2:1	4:1	768	0.0320513
	8	2:1	4:1	1024	0.0421345
16	4	4:1	2:1	256	0.0107195
	4	4:1	2:1	512	0.021444
	4	4:1	2:1	768	0.0325874
	4	4:1	2:1	1024	0.0437981
32	2	8:1	1:1	256	0.0107846
	2	8:1	1:1	512	0.0217085
	2	8:1	1:1	768	0.0330778
	2	8:1	1:1	1024	0.044588

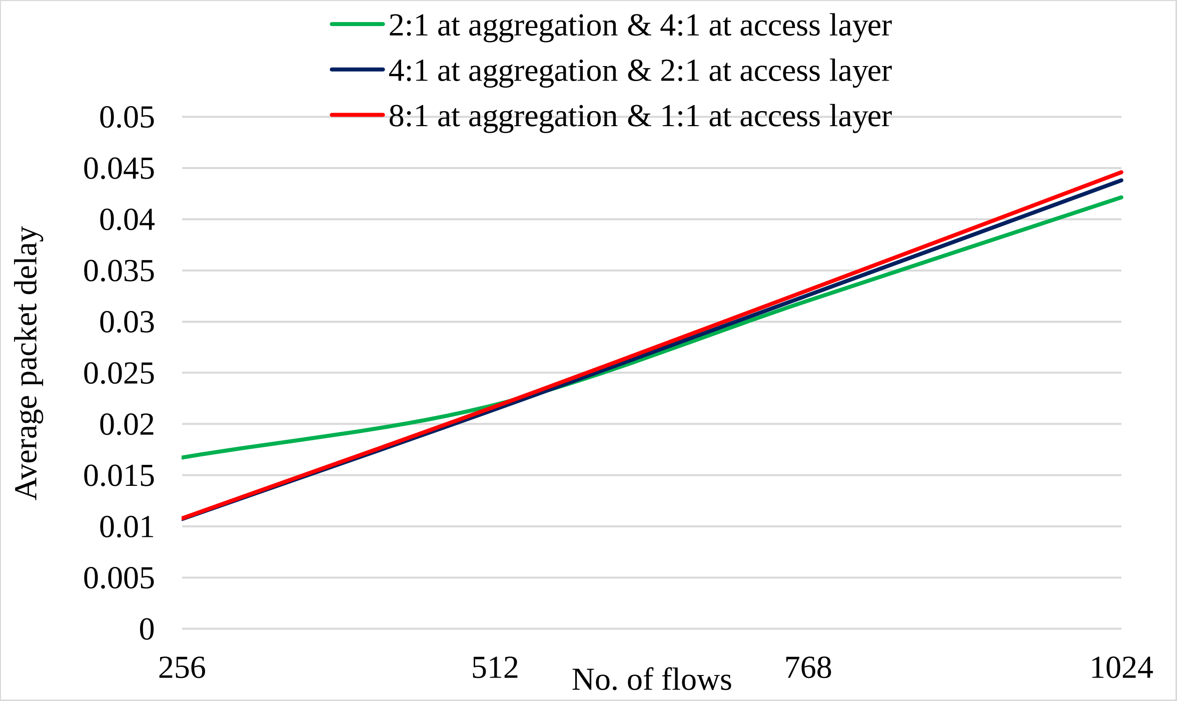


Figure 4- 5: Effect of OSR on average packet delay for 256 hosts

As observed in Figure 4-5 the average packet delay of the network which has 256 total number of hosts in the entire network has different patterns. For small number of traffic flows, the network architecture highlighted with the red color which is constructed with high oversubscription ratio at aggregation layer and low degree of oversubscription ratio at access layer produces low average packet delay, but this network topology results high average packet delay when the number of traffic flow increases.

Table 4- 6: Average packet delay results for 384 total number of hosts

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of flows	Average packet delay (sec)
12	8	3:1	4:1	384	0.0159733
	8	3:1	4:1	767	0.032263
	8	3:1	4:1	1151	0.0471781
	8	3:1	4:1	1536	0.063037
	8	3:2	4:2	1920	0.080235
16	6	4:1	3:1	384	0.0162384
	6	4:1	3:1	767	0.0326747
	6	4:1	3:1	1151	0.0495294
	6	4:1	3:1	1536	0.0651473
	6	4:2	3:2	1920	0.0822919
24	4	6:1	2:1	384	0.016306
	4	6:1	2:1	767	0.0326417
	4	6:1	2:1	1151	0.0488707
	4	6:1	2:1	1536	0.0665072
	4	6:2	2:2	1920	0.084013

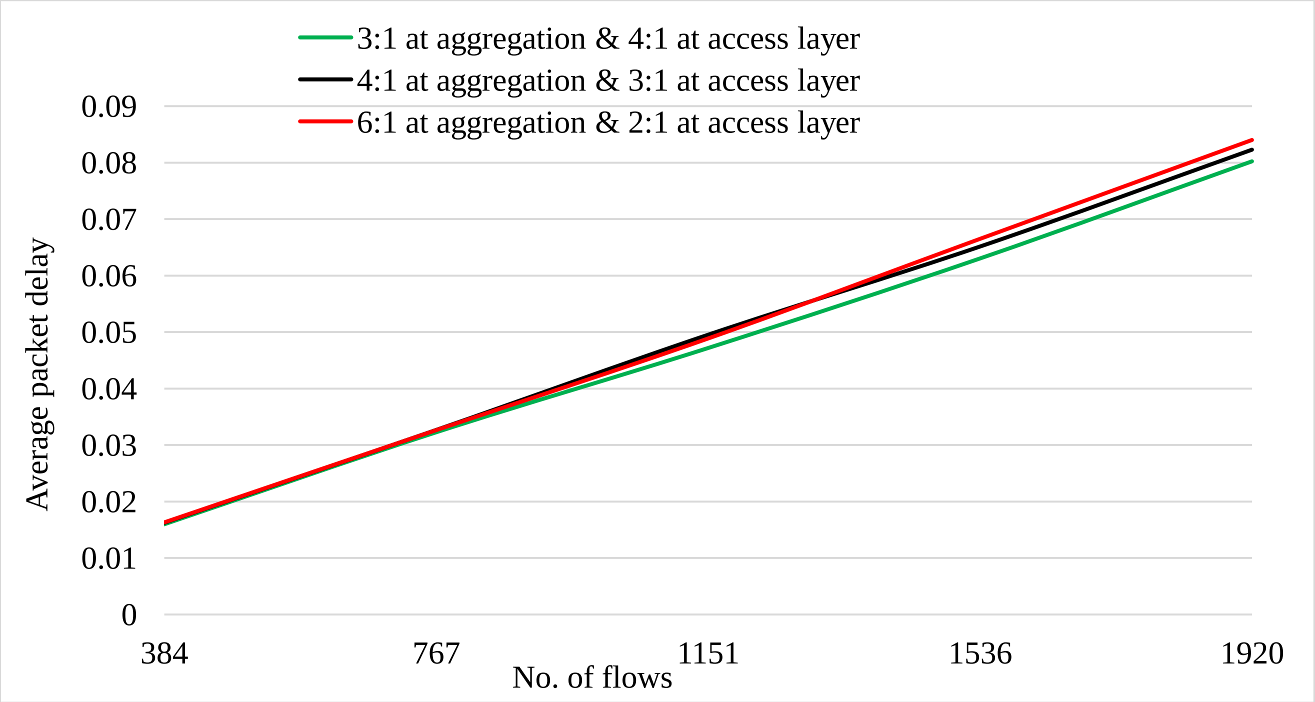


Figure 4- 6: Effect of OSR on average packet delay for 384 hosts

Figure 4-6 shows the results of the average packet delay for 384 total number of hosts in the entire data center network and as the observed in the graphs, for small number of traffic flows the network with high oversubscription ratio at the aggregation layer and small oversubscription ratio at the access layer produces low average packet delay. However, when the number of traffic flow increases, the network with low degree of oversubscription ratio at the aggregation layer and high oversubscription ratio at the access layer produces low packet delay.

The network architecture highlighted with the green color, build with low oversubscription ratio at the aggregation layer and small oversubscription ratio at the access layer produces low average packet delay when the total number of traffic flows increases. Generally, the network architecture build with low degree of oversubscription ratio at the aggregation layer and high oversubscription ratio at the access layer can produces low average packet delay when compared with the network with high oversubscription ratio at the aggregation layer and low oversubscription ratio at the access layer. Thus, to minimize the average packet delay, minimizing the oversubscription ratio at the aggregation layer is the best solution.

B) Case 2 : Using 8-module Three-tier DCN architecture

Table 4-7 shows the output result of average packet delay obtained when simulating the network using 8-module DCN by incrementing the total number of flows in the network.

Table 4- 7: Average packet delay results for 1536 total number of hosts

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of flows	Average packet delay (sec)
8	24	1:1	12:1	1635	0.00718837
	24	1:1	12:1	1736	0.00765889
	24	1:1	12:1	1836	0.00790331
	24	1:1	12:1	1936	0.00832801
	24	1:1	12:1	2036	0.00856931
16	12	2:1	6:1	1635	0.00724757
	12	2:1	6:1	1736	0.00772931
	12	2:1	6:1	1836	0.00808925
	12	2:1	6:1	1936	0.0085924
	12	2:1	6:1	2036	0.00904363
24	8	3:1	4:1	1635	0.00723371
	8	3:1	4:1	1736	0.00747198
	8	3:1	4:1	1836	0.00820551
	8	3:1	4:1	1936	0.008765
	8	3:1	4:1	2036	0.0092834
32	6	4:1	3:1	1635	0.00725611
	6	4:1	3:1	1736	0.00769444
	6	4:1	3:1	1836	0.00835236
	6	4:1	3:1	1936	0.0089399
	6	4:1	3:1	2036	0.00958171

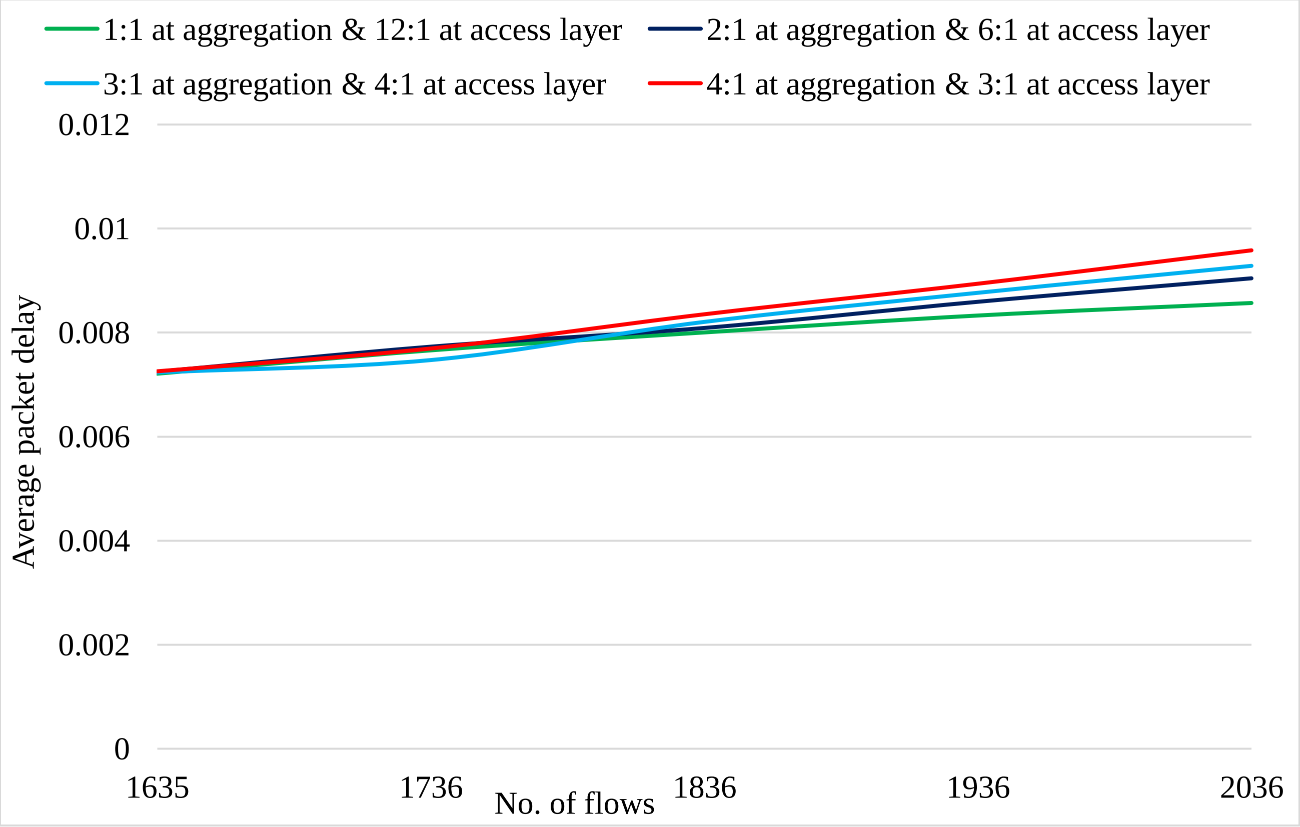


Figure 4- 7: Effect of OSR on average packet delay for 1536 hosts

Figure 4-4 to 4-7 illustrates the effect of oversubscription ratio on the average packet delay for a DCN topology build using different number of access layer switches but have the same total number of hosts in the entire network and the same number of traffic flows in each figure.

As observed in Figure 4-8, the average packet delay increases as the total number of traffic flows increased for all types of topologies as usual. In the above figures, the network highlighted with green color produces a low average packet delay than the networks highlighted with other colors. This indicates, the average packet delay of a network build with high oversubscription ratio at the access layer and low oversubscription ratio at the aggregation layer is greater than a network with small oversubscription ratio at access layer and high oversubscription ratio at the aggregation layer. Even if the difference between the average packet delay is not much large, the effect of this small fraction of packet delay makes a big difference on delay sensitive applications such as online audio and video streaming. So, even if the network is non-blocking or a partially oversubscribed with small oversubscription ratio at the access layer, the status (blocking or non-blocking) of the network at the aggregation layer is a key factor to have a low average packet delay.

4.4.2 Effect of oversubscription ratio on average throughput

Using the same parameter settings used for average packet delay analysis in the above scenario (the same number of flows, total number of hosts, the same oversubscription at aggregation and access layer) the network throughput of different network architectures was evaluated. The following table shows sample results obtained from the simulation.

A) Case 1 : Using 4-module Three-tier DCN architecture

The sample results obtained after simulating the Three-tier data center network architecture structured with 4-module/pod network topology are described in Table 4-8 and as the table shows for all types of network topologies build with different ToR layer switches, the average throughput of the network is gradually decreasing when the total number of traffic flows in the entire network increases. However, the behavior of the average network throughput has different characteristics in these topologies (topologies constructed with 8, 16 and 32 ToR layer switches) and this difference comes from the variation in oversubscription ratio both at aggregation layer and access layer.

Table 4- 8: Average throughput results for 256 total number of hosts

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of flows	Average Throughput (Mbps)
8	8	2:1	4:1	256	201.278
	8	2:1	4:1	512	200.299
	8	2:1	4:1	768	199.687
	8	2:1	4:1	1024	198.418
16	4	4:1	2:1	256	197.569
	4	4:1	2:1	512	200.938
	4	4:1	2:1	768	198.772
	4	4:1	2:1	1024	196.339
32	2	8:1	1:1	256	198.263
	2	8:1	1:1	512	195.611
	2	8:1	1:1	768	193.183
	2	8:1	1:1	1024	191.732

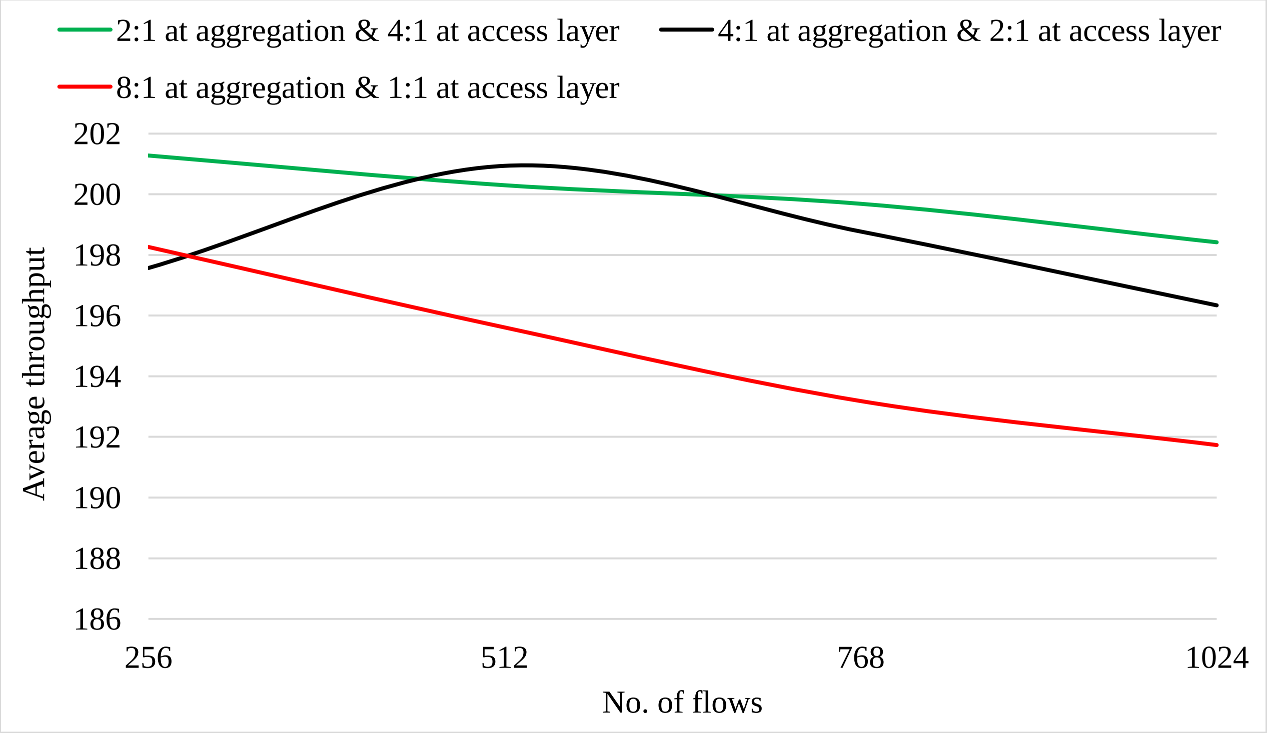


Figure 4- 8: Effect of OSR on average throughput for 256 hosts

Figure 4- 8 describes the results of average network throughput of 256 total number of hosts in the entire network and as the graphs shows the average network throughput is high in all types of architecture when the total traffic flow is low. However, as the number of traffic flow increases, the average network throughput is linearly decreased in all types of network architecture. In comparison, the network architecture highlighted with red color which has high oversubscription ratio at the aggregation layer and low oversubscription ratio at the access layer produces low network throughput but the network highlighted with green color which has low oversubscription ratio at the aggregation layer and high oversubscription ratio at the access layer produces high network throughput.

Table 4- 9: Average throughput results for 384 total number of hosts

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of flows	Average Throughput (Mbps)
12	8	3:1	4:1	384	203.823
	8	3:1	4:1	767	204.537
	8	3:1	4:1	1151	204.741
	8	3:1	4:1	1536	202.595
	8	3:1	4:2	1920	200.637
16	6	4:1	3:1	384	198.516
	6	4:1	3:1	767	198.872
	6	4:1	3:1	1151	195.541
	6	4:1	3:1	1536	194.61
	6	4:1	3:1	1920	191.865
24	4	6:1	2:1	384	196.858
	4	6:1	2:1	767	195.511
	4	6:1	2:1	1151	196.022
	4	6:1	2:1	1536	192.831
	4	6:1	2:1	1920	190.654

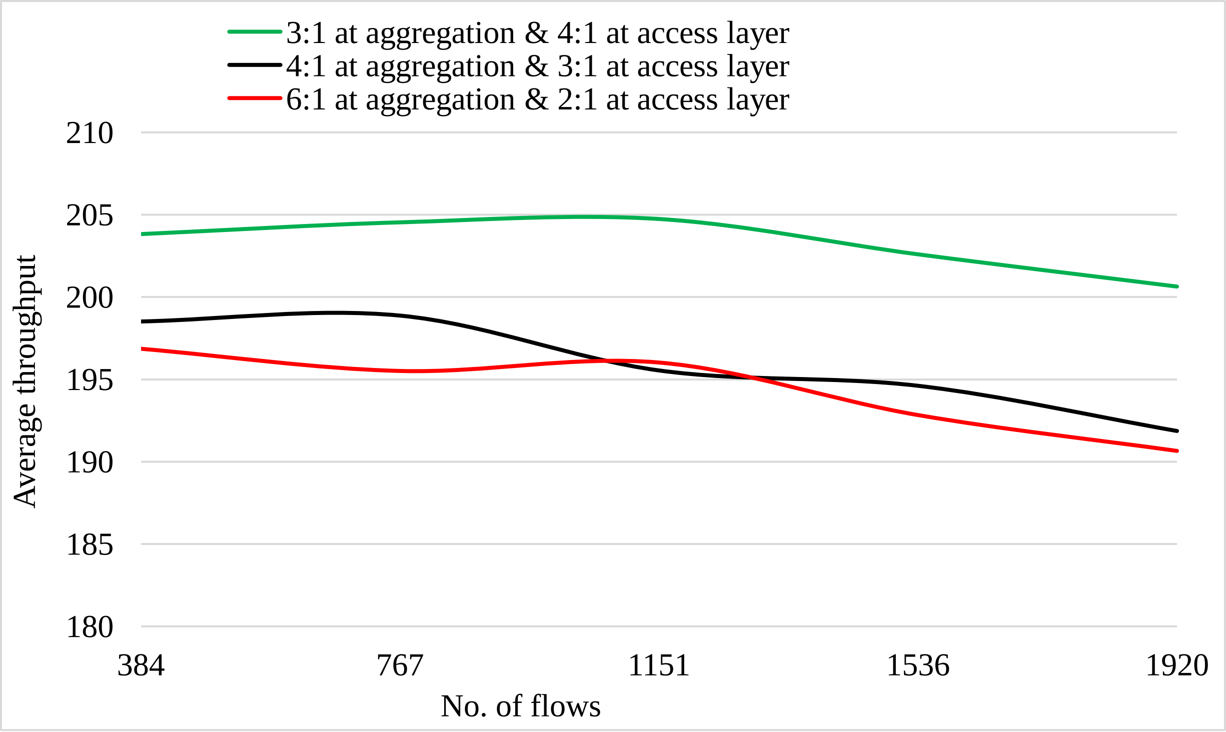


Figure 4- 9: Effect of OSR on average throughput for 384 hosts

As observed in Figure 4-8, the average network throughput of the network architecture that have 384 total number of hosts has the same characteristics the network architecture which has 256 total number of hosts in the entire network. The network build with high oversubscription ratio at the aggregation layer and small degree of oversubscription ratio at the access layer produces low average network throughput but the network build with low oversubscription ratio at the aggregation layer and high oversubscription ratio at the access layer produces high network throughput.

B) Case 2 : Using 8-module Three-tier DCN architecture

Table 4- 10: Average throughput results for 1536 total number of hosts

No. of ToR	Hosts in each ToR switch	Osr at Aggregation layer	Osr at Access layer	Total No of flows	Average Throughput (Mbps)
8	24	1:1	12:1	1635	1911.12
	24	1:1	12:1	1736	1903.13
	24	1:1	12:1	1836	1872.64
	24	1:1	12:1	1936	1867.75
	24	1:1	12:1	2036	1861.48
24	8	3:1	4:1	1635	1856.32
	8	3:1	4:1	1736	1847.08
	8	3:1	4:1	1836	1836.77
	8	3:1	4:1	1936	1831.61
	8	3:1	4:1	2036	1825.93
32	6	4:1	3:1	1635	1844.1
	6	4:1	3:1	1736	1824.36
	6	4:1	3:1	1836	1823.49
	6	4:1	3:1	1936	1814.87
	6	4:1	3:1	2036	1809.51

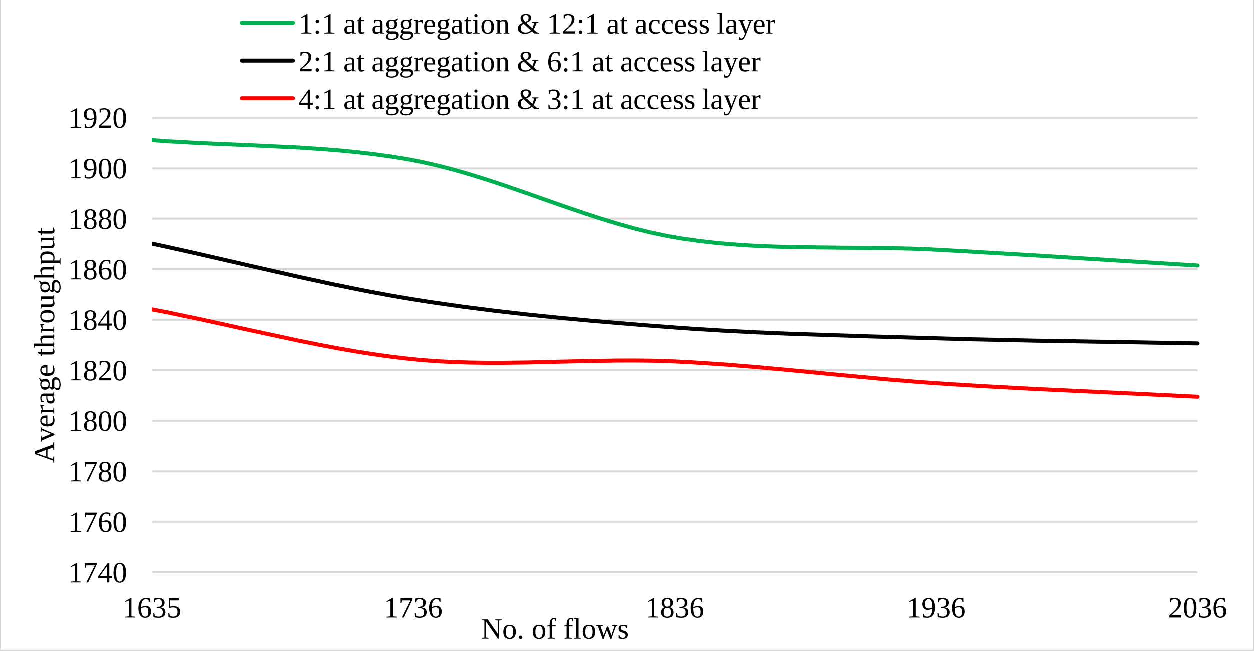


Figure 4- 10: Effect of OSR on average throughput for 1536 hosts

In the 4-module network architecture, the simulation results show that the network architecture build with low oversubscription ratio at the aggregation layer and small oversubscription ratio at the access layer produces high network throughput. Here, for 8-module network architecture the behavior of the average network throughput has the same characteristics. It is observed that the average network throughput decreases gradually when the total number of traffic flows increased in all types of the network architecture. As previously stated, network throughput is dependent of the average packet delay, if the network has low average packet delay, the network throughput will be high and if the network has high average packet delay, the network throughput is low. So, proper oversubscription ratio should be defined when designing and deploying data center networks using Three-tier architecture.

4.5 Effect of oversubscription ratio on scalability

Scalability of the data center network is the ability of the network architecture for incremental expansion to handle the growing demands of new applications without affecting the existing services and the performance of the network. As the network expands the traffic flows at the higher layers also increased and when the network expands it should accommodate these increased traffic flows without oversubscribing the network and impacting the cost of each devices. The major challenge for data center network scalability is the oversubscription ratio of links at the higher layers. In section 4.4 the effect of oversubscription ratio on the average packet delay and average throughput of the network was demonstrated by varying the oversubscription ratio both at the aggregation layer and access layer and setting the total number of hosts in the entire network constant and increasing the traffic flows.

As observed in the simulation results oversubscription ratio is the main cause of poor network performance and to produce high network performance the oversubscription ratio at the aggregation layer should be minimum. The network build with high oversubscription ratio at the aggregation layer and low oversubscription ratio at the access layer produces low network performance but the network constructed with low oversubscription ratio at the aggregation layer and high oversubscription ratio at the access layer produces high network performance.

In order to achieve 1:1 oversubscription ratio at the aggregation layer the Three-tier DCN architecture build with C core layer switches and G aggregation layer switches, the core layer switches have a minimum of 1*G number of downlink port and each aggregation layer switches has a minimum of 1*C number of uplink ports. Let’s say the number of access layer switches in one module/pod are S and to achieve 1:1 oversubscription ratio at the access layer the minimum number of uplink ports are 1*2*S because there are two aggregation layer switches in on module/pod. In the 4-module Three-tier network architecture build with 4 number of core switches, 8 aggregation layer switches and the oversubscription ratio in a pod, at the aggregation layer is S:C=S/4=Og:1 and at the access layer is H:G=H/2=Os:1 where S is the total number of access layer switches, H is the number of hosts attached in one access switches, Og and Os are is the oversubscription ratio at the aggregation layer and access layer respectively. Then, the total number of end hosts in a pod will be S*H= 4*Og *2*Os.

To examine the effect of oversubscription ratio on the network scalability, the three network architectures that are used previously for the performance comparisons are considered and to observe the scalability results the evaluation is performed by incrementally increasing the number of end hosts in each pod by adding new ToR layer switches. In scaling or incrementally expanding the network, it is obvious that the oversubscription ratio at the aggregation layer also increases while adding the ToR layer switches in the access layer and the oversubscription ratio at the access layer also increases when new end hosts added to each ToR switches.

To demonstrate these scenarios the network build with 12, 16 and 24 ToR layer switches takes as the initial network or the existing network. Figure 4-11 describes the new network after the three networks were expanded by adding different number of ToR layer switches and end hosts. As the figure describes the network build with 16 and 24 ToR layer switches can scale linearly to high and supports large number of end hosts even if the oversubscription ratio at the access layer increase. However, achieving linear scalability without oversubscribing the network and impacting the performance of the network is the major goal of data center network architectures.

As Figure 4-4 and Figure 4-10 show the performance(average packet delay and throughput) results of the three networks and as the figures depicts the network build with 12 ToR layer switches (which has low oversubscription ratio at the aggregation layer i.e. 3:1 and high low oversubscription ratio at the access layer) produces high network performance than the network build with 16 and 24 ToR layer switches. As Figure 4-11 describes the network build with 3:1 oversubscription ratio at the aggregation layer can scale to different types of networks which has 3.25:1, 3.5:1 and 3.75:1 oversubscription ratio on the aggregation layer. The results of these expanded networks are almost similar with the network build with 16 ToR layer switches and these networks also produces similar performances.

So, when designing and deploying a data center network, not only the performance of the network is considered but future incremental expansions the network also needs critical consideration because if the network properly designed and deployed properly at the initial stage expanding the network and scaling the manageability and operational capabilities of the network infrastructure can be handled easily. In conclusion, when expanding the network, the performance of the network should not be affected and the cost of re-deigning and deploying should be minimum. So, scaling needs to be linear for both performance and cost i.e., as a system supports more servers, the price per host and the performance per host stays very close to the same.

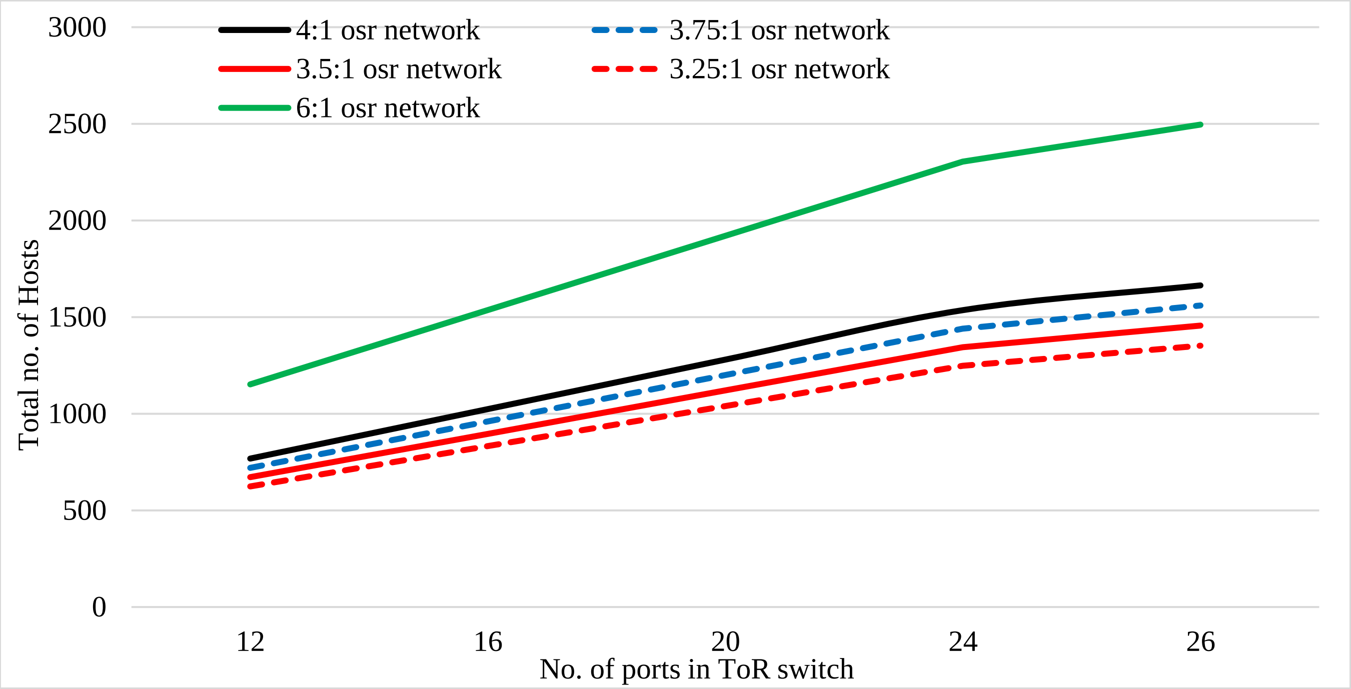


Figure 4- 11: Effect of OSR on network scalability in Three-tier architecture

CHAPTER FIVE: DISCUSSION

The main data center network performance measurements or metrics are throughput, latency and reliability. The data centers are designed to achieve high throughput, low latency and better reliability [11] on different load conditions. The oversubscription ratio is also considered as key performance measure of data center networks and 1:1 ratio indicates that communication is at full bandwidth of their network capacity [38].

5.1 Packet delay

The time required to transmit a packet of data from sender host to destination host along its entire path which is created by an application on the sender side, handed over to the Operating system, passed to a network interface card (NIC), encoded, transmitted over a physical or wireless medium such as copper, fiber and air received by an intermediate active device (firewall, router), switch load-balancer), analyzed by these network devices and retransmitted over another medium, etc. Due to budget limitations IT industries and services providers install some low-cost commodity off-the-shelf active and passive devices inside the data center network. These off-the-shelf devices has different specifications and interoperability characteristic which describes the operating properties in the computing world.

Interoperability is a characteristic of a product or system, whose interfaces are completely understood, to work with other products or systems, present or future, in either implementation or access, without any restrictions [46]. The processing capabilities, the features, the protocols they support are some of the important issues of the devices and Interoperability between device link operation standards. Thus, all these active devices and passive devices participating in the whole packet transmission have a significant impact on the packet delay due to the specifications and compatibility characteristic of those devices.

5.2 Throughput

Throughput is a fundamental performance metric of a data center networks which indicates the network's capacity i.e. how much data can be successfully transferred from source to destination at any given unit of time. The simulation results in this study reveals that the throughput and packet delay are inversely proportional each other, if the average packet delay is high, the network throughput will be low and vice versa. Since the demand for data exchange in DCNs is extremely large compared with other networks, the first design goal is to maximize the throughput. The throughput of data center network depends on the packet delay, the data rate of the channel and the rate of successful messages. Having multiple possible routing paths between nodes leads to less traffic congestion and more available bandwidth which improves the throughput. Also, minimizing packet delay is a major task to improve throughput and to lowering packet delay, monitoring the traffic flows on each switch interfaces and the bandwidth usage of each applications is important. Generally, to support the new services deployed in the data center, the performance of the network is the basic enabler that insures whether link congestion or bandwidth bottleneck will occur at any layer of the DCN architecture or not.

5.3 Oversubscription ratio

As observed in the simulation results the average packet delay is increases linearly, and the average network throughput is also decreasing as the number of hosts attached to the access switch increased. These large number of hosts in the access layer creates more traffic flows to the upper layer switch and links will be congested due to high oversubscription ratio. When the switch receives concurrent traffic flows, the number of waiting queues increase. This increases the processing delay, queuing delay, finally increase total packet delay and decreases the network throughput.

As discussed in chapter 2, the largest percentage of traffic flow is the east-west traffic pattern that flows from rack-to-rack and module-to-module. This indicates the upper/core layer couldn't be bottleneck more than the aggregation and access layer. In the data center servers install in the same access switch should have a 1:1 oversubscription ratio to other servers to communicate with full interface bandwidth. But this ratio can't achieve on the aggregation and core layers due to budget constraints.

The simulation result shows that, in a Three-tier DCN architecture, oversubscription ratio increases continuously from lower layer to the upper layer and the oversubscription ratio may be much higher and may vary from a topology to topology both at the aggregation and access layers. When the oversubscription ratio is very high, the performance of the DCN can be low and applications will suffer in losing packets and when the oversubscription ratio is too low, the total number of hosts installed, and the number of new services or applications deployed on these hosts at the data center will be decreased. So, in the designing and deploying a data center network architecture balancing the hardware cost, deployment cost and variation in the oversubscription ratio at each layer needs a critical consideration. Not only at the design and deployment stage but also needs to redesign the oversubscription ratio when there is an expansion on the data center to support the increasing demand of new applications.

In the previous section the effect of oversubscription ratio on the average packet delay and throughput was discussed and the figures illustrate that the network performance is improved when the oversubscription ratio at the aggregation layer is low and high at the access layer than a network having high oversubscription ratio at aggregation layer and low or almost non-blocking at the access layer.

As discussed in chapter 2, in the data center component costs, the largest percentage of the amortized cost i.e. ~45% from the total cost goes to the servers (CPU, memory, storage, etc.) and the ~15% cost is invested to the network equipment (switches, routers, links, etc.). Since, building data center network by attaching small number of hosts on each access switch doesn't improve the performance of the network rather this way of designing topology increases the hardware costs of switches, the wire/cable complexity between aggregation layer and access layer and management complexity . Thus, it is important and feasible to attaching more hosts on each access switches in designing a DCN architecture to improve performance of the network and reduce networking costs, deployment costs and operational complexity.

As discussed in chapter 2, traffic flow in the data center is classified as north-south and east-west traffic. The east-west traffic flow is takes place inside the data center and consists the intra-rack and inter-rack traffic. The intra-rack traffic refers the traffic generated by servers and communicate to other servers within the rack and the inter-rack traffic refers the flow of traffic from one rack to another rack. In the intra-rack traffic flow, servers can communicate each other with full bisection bandwidth depending on the data rate of ports where they are connected to the switches. However, the inter-rack traffic, the traffic that exits the rack suffers different problems due to oversubscribed network. This oversubscribed network has congested links that are fully utilized and causes high average packet delay and low average throughput as observed in the simulation results.

5.4 DCN Components

As stated earlier the main cause of a poor data center network performance is undefined oversubscription ratio on each layers of the architecture. Beside the oversubscription ratio, there are many issues that lead to have a poor network performance in the data center. The hardware components used to build the network architecture has a significant impact on the performance of the data center.

5.4.1 Switches

Switches are one of the principal components of the data center network topology specially in Three-tier DCN architecture and these switches handle the major tasks in data transmission process. the basic functionality of switches are handling layer2 and layer3 tasks such as creating media access control(MAC) table used to attaches frames on a destination ports, prevent network loops, create isolate networks or virtual LANs, implement security futures like access control lists, quality of service policies and building routing table and path determination to forward packets to the next hop or node and etc. However, all switches don't perform those layer2 and layer3 tasks the same for the services installed on the data center because different types of switches have specific characteristics.

In the aggregation layer there are many devices installed for different purposes such as firewalls for filtering trusted and untrusted traffics coming from and exiting out the top layer, load-balancers used for traffic sharing for each server, intrusion detection, intrusion prevention systems, and etc. Adding these devices on the data center network will create congestion and a high packet delay and lowers average throughput which in turn has a visible impact of network performance. The processing capabilities and congestion avoidance mechanisms in these network devices is essential and needs proper consideration. So, the switches, firewalls, load-balancers and other devices installed in the middle layer of the data center network should be new and excellent which are capable of fulfilling the requirements of the services provided by the data center networks and support the newly application demands that will be added in the data center network.

5.4.2 Servers

Network switches are not only affecting the performance of the data center network, but servers also participate in resulting poor performance network. Network servers are the main part of data center network which are used to store, process and analyze date to create and disseminate information. Servers are categorized in three types which are tower, rack and blade servers [58]. Tower servers are designed for data center networks, but they are not compatible for large scale data center networks and rack servers are fit and the best ones for the data center networks because their form factor is rack mountable which saves much space and easy of placement. Blade servers are designed and come up with both computing and switching capacity but consume much rack space.

The software installed on these servers should be compatible with the applications installed on the servers for quick response and fast data computation (retrieve, update, add, remove etc.) requested from other nodes. The processing capabilities and other hardware parts of these servers should be high-quality and comply the requirements of modern application demands. Because if the server processing capability is high there will not be more packets in queuing state, response time delay and the server NICs will not be congested. Thus, it needs proper consideration in deploying a Three-tier data center network.

5.4.3 Cables

Cables are the backbone of data center network that are used to interconnect all those active devices and makes functional, but their advantages often don’t consider specially in the beginning of network installation. Cable installations should be done with a standard and proper design rules since there will be large amount of cable in the data center and if network problems happen, it may be costly to rewire the nodes once the installation is completed and the services start working.

Network cables are categorized in two types based of the media used for packet transmission, fiber and copper. They used in separately and can be used together to create a link between two devices through media converters. These media converters are used to convert signals from fiber media type to copper and vice versa. However, such types of components add additional delay which affects the performance of the network. So, if possible removing these components in the network is advisable. Due to cost constraints data center operators use more copper cables specifically UTP (Unshielded Twisted Pair) to connect computing device because devices with fiber or optical ports are expensive. Copper cables are categorized in different types with respect to their specifications and the main metric to categorize these cables is the data rate they support. If the data rate of the server and the access switch are 1Gbps and if the cable used to connect this device 100Mbps, then it is clear there will be congestion and drop packets which increase packet delay. So even if the cable is not damage or cut, poor network cable choices can still affect the network performance highly.

5.4.4 Configuration

Even if, the physical network infrastructure and the network components used to build data center architecture are a major cause of poor performance network, but there are some issues that also affect the data center network performance. Device configuration is one of those issues because if routing protocols are not properly configured, packets will use the longest path and increase packet delay, if the load balancing protocols are not properly configured, some links will be idle and packets will be transmit only in some pair of links and if loop avoidance protocols are not properly configured, many ports will be blocked and packets will be transmitted in active ports.

Speed mismatch is another cause of poor network performance which is occur between two directly connected device. If the server port can support up to 10Gbps and the switch port is the same i.e. 10Gbps, there will be smooth packet transmission process until the traffic is more than 10Gbps. but this will be happening if the speed setting by default is auto and configured to negotiation mode.

However, if the port setting in one device is configured manually to limit the bandwidth usage of the port or other purposes and leave the port setting auto mode in the other device, speed miss-match will be occurred. All these and other configuration issues creates link congestion, increase drop packets, increase packet delay, decrease throughput and finally results poor network performance. Thus, designing and deploying data center network requires high considerations and deep understanding of the characteristic of active and passive components.

5.4.5 Cost

Cost is the major challenge of data center network and when the network is designed and deployed the amount of capital expenditure (CAPEX) will be high and to minimize these costs data center operators utilize low-cost commodity off-the-shelf components in the data center network. However, as the number of applications increase, these low-cost commodity off-the-shelf components will not operate and compatible with the new technology features and replacement of these components will require at the time of network expansion. Indeed, these low-cost components doesn't perform well in the long term since the life-time of these components may be short duration and failures will increase and finally the cost of operational expenditures (OPEX) will be increase. So, the critical consideration is needed for these network components specially cables, switches, servers, load-balancers, firewalls and etc. used in constructing the Three-tier data center network architecture.

CHAPTER SIX: CONCLUSION AND FUTURE WORK

6.1 Conclusion

In this thesis, the state-of-the-art data center network were surveyed and the characteristics, deployment strategies, the problems they address, and the limitations of those proposed data center network architectures were also discussed. The performance of Three-tier data center network architecture is evaluated using ns-3 and the simulation results shows that the performance of the Three-tier architecture is highly affected by the oversubscription ratio of links. Mainly, the architecture is evaluated by varying the number of pods/modules and the number of traffic flows in the network to observe the effect of oversubscription ratio on the average packet delay and network throughput. In this thesis, beside the oversubscription ratio, the communication components used to build the architecture such as the switches, servers, cables and media converters also directly influence the network performance also discussed and basic solution to these issues are recommended. As the data center network incrementally expands scaling the data center network affects the overall performance of the network and the cost of network components also increases. So, proper oversubscription ratio should be set on each layer of the architecture to achieve a trade-off between the network performance and the costs.

6.2 Future work

The Three-tier DCN architecture is the most promising and widely deployed architecture in the data center due to its advantages (such as security, scalability and easy management) than other. Here below are some insights and research directions to be address in the future so as to improve the performance of Three-tier DNC architecture.

- Traffic isolation and congestion avoidance mechanisms

In the future most data communication takes place in the data center which will increase the percentage of intra-data center traffic flow. There are different traffic types, management traffic, service traffic, broadcast traffics for network convergence, etc. Thus, a mechanism used to isolate and improve these traffic types is required to lower congestion. Also, in the future, congestion handling and avoidance mechanisms of the computing devices and on each layer of the network architecture are required.

- Load-balancing algorithms

Today's algorithms to load-balance traffic between different paths are performed by grouping ports with equal number of ports and the same port speed or data rate, the same duplex mode, the same interface type and etc. and the maximum number of ports supported by this algorithm are eight. Since deploying high-class switches in the core and aggregation layer is expensive, load-balancing algorithms that are more efficient, open standard and support more ports are required.

- Routing protocols and algorithms

As the number of computing devices increased in the data center, the network will take large amount of time to fully converge and to build the routing table. The routing algorithms are the major factor in the performance of the routing protocols. Thus, new routing protocols and improving the routing algorithms and protocols for the data enter network is vital to fast network convergence, quickly query of routing table and direct traffic efficiently.

- Improve Quality of service

The performance of quality of service is depend on the underlying network architecture and the modules structure in the data center. Quality of service is mainly achieved by prioritizing each type of traffic transmitted between each pod. A mechanism to isolate the services provided by each pod and load-balancing traffic across the pods if the pod is oversubscribed is required to improve application performance and quality of service.

Reference

[1] T. Wang, Z. Su, Y. Xia and M. Hamdi, "Rethinking the Data Center Networking: Architecture, Network Protocols, and Resource Sharing," in IEEE Access, vol. 2, pp. 1481-1496, 2014. doi: 10.1109/ACCESS.2014.2383439

[2] H. Emesowum, A. Paraskelidis and M. Adda, "Achieving a Fault Tolerant and Reliable Cloud Data Center Network," 2018 IEEE International Conference on Services Computing (SCC), San Francisco, CA, 2018, pp. 201-208

[3] Sergio Jim´enez Feij ´oo," Design and Evaluation of Architectures for Intra Data Center Connectivity Services", March 7, 2014

[4] Yao, Fan et al. "A comparative analysis of data center network architectures." 2014 IEEE International Conference on Communications (ICC) (2014): 3106-3111.

[5] Bilal, Kashif, et al. "A Comparative Study of Data Center Network Architectures." ECMS. 2012.

[6] Liu, Yang et al. "A Survey of Data Center Network Architectures." (2013).

[7] Sethi, Pallavi and Smruti R. Sarangi. "Internet of Things: Architectures, Protocols, and Applications." J. Electrical and Computer Engineering 2017 (2017): 9324035:1-9324035:25.

[8] Z. Zhang, Y. Deng, G. Min, J. Xie, L. T. Yang and Y. Zhou, "HSDC: A Highly Scalable Data Center Network Architecture for Greater Incremental Scalability," in IEEE Transactions on Parallel and Distributed Systems. doi: 10.1109/TPDS.2018.2874659

[9] Singla, Ankit et al. "Proteus: a topology malleable data center network." HotNets (2010).

[10] Yaqoob, Ibrar et al. "Internet of Things Architecture: Recent Advances, Taxonomy, Requirements, and Open Challenges." IEEE Wireless Communications 24 (2017): 10-16.

[11] Bilal, Kashif et al. "Quantitative comparisons of the state-of-the-art data center architectures." Concurrency and Computation: Practice and Experience 25 (2013): 1771-1783.

[12] Popa, Lucian et al. "A Cost Comparison of Data Center Network Architectures." (2010).

[13] Al-Fares, Mohammad et al. “A scalable, commodity data center network architecture.” SIGCOMM (2008).

[14] Bilal, Kashif et al. “A taxonomy and survey on Green Data Center Networks.” Future Generation Comp. Syst. 36 (2014): 189-208.

[15] Bari, Md. Faizul et al. “Data Center Network Virtualization: A Survey.” IEEE Communications Surveys & Tutorials 15 (2013): 909-928

[16] Sari, A. and Akkaya, M. (2015) Security and Optimization Challenges of Green Data Centers. Int. J.Communications, Network and System Sciences, 8, 492-500.

[17] Scarfò, Antonio. “The Evolution of Data Center Networking Technologies.” 2011 First International Conference on Data Compression, Communications and Processing (2011): 172-176.

[18] Lebiednik, Brian et al. “A Survey and Evaluation of Data Center Network Topologies.” CoRR abs/1605.01701 (2016): n. pag.

[19] Liu, Yang . “Design and evaluation of data center network typologies.” , 2013

[20] Guo, Chuanxiong et al. “DCell: a scalable and fault-tolerant network structure for data centers.” SIGCOMM (2008).

[21] Guo, Chuanxiong et al. “BCube: a high performance, server-centric network architecture for modular data centers.” SIGCOMM (2009).

[22] Alizadeh, Mohammad and Tom Edsall. “On the Data Path Performance of Leaf-Spine Datacenter Fabrics.” 2013 IEEE 21st Annual Symposium on High-Performance Interconnects (2013): 71-74.

[23] Hafeez, Taimur et al. “Detection and Mitigation of Congestion in SDN Enabled Data Center Networks: A Survey.” IEEE Access 6 (2018): 1730-1740.

[24] https://lenovopress.com/lp0573.pdf

[25] https://www.cisco.com/c/en/us/products/collateral/switches/nexus-7000-series-switches/white-paper-c11-737022.pdf

[26] Greenberg, Albert G. et al. “Towards a next generation data center architecture: scalability and commoditization.” PRESTO (2008).)

[27] Curtis, Andrew R. et al. “LEGUP: using heterogeneity to reduce the cost of data center network upgrades.” CoNEXT (2010).

[28] Curtis, Andrew R. et al. “REWIRE: An Optimization-based Framework for Data Center Network Design.” (2011).

[29] Singla, Ankit et al. “Jellyfish: Networking Data Centers Randomly.” NSDI (2011).

[30] Curtis, Andrew. “Reducing the Cost of Operating a Datacenter Network.” (2012).

[31] Cisco Nexus 2000 Series Fabric Extenders Data Sheet, 2009

[32] Patel, Rajan Kashibhai. “Investigation of Network Simulation Tools and Comparison Study: NS 3 vs NS 2.” (2016).

[33] IOSR Journal of Electronics and Communication Engineering (IOSR-JECE) e-ISSN: 2278-2834, p- ISSN: 2278-8735. PP 52-56

[34] Zola, Enrica & Martin-Escalona, Israel & Barcelo-Arroyo, Francisco. (2010). Discrete Event Simulation of Wireless Cellular Networks. 10.5772/9903

[35] Dr. L. RAJA.” STUDY OF VARIOUS NETWORK SIMULATORS”, International Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395-0056, p-ISSN: 2395-0072, Dec 12, 2018

[36] Varga, A. and Rudolf Hornig. “An overview of the OMNeT++ simulation environment.” Simutools 2008 (2008)

[37] Mrs. Saba Siraj . “Network Simulation Tools Survey.” International Journal of Advanced Research in Computer and Communication Engineering Vol. 1, Issue 4, June 2012

[38] Larocque, Guy R. and Samuel Lipoff. “Application of discrete event simulation to network protocol modeling.” Proceedings of ICUPC - 5th International Conference on Universal Personal Communications 2 (1996): 508-512 vol.2

[39] Ören, Tuncer. “The Many Facets of Simulation through a Collection of about 100 Definitions.” (2011)

[40] https://www.nsnam.org/docs/release/3.11/manual/ns-3-manual.pdf

[41] Weingärtner, Elias et al. “A Performance Comparison of Recent Network Simulators.” 2009 IEEE International Conference on Communications (2009): 1-5.

[42] Okezie, Christiana C. and Okafor Kennedy. “Performance evaluation of a Reengineered Data Center Network using a link state protocol implementation.” (2012).

[43] Greenberg, Albert G. et al. “VL2: a scalable and flexible data center network.” Commun. ACM 54 (2011): 95-104.

[44] Zhuo, Danyang et al. “Understanding and Mitigating Packet Corruption in Data Center Networks.” SIGCOMM (2017).

[45] Cisco Systems, “Cisco Global Cloud Index: Forecast and Methodology,” White Paper, 2015-2020, November 2016.

[46] https://www.nsnam.org/docs/release/3.16/doxygen/group_nixvectorrouting.html

[47] Mostafa, Ahmed M.. “IoT Architecture and Protocols in 5G Environment.” (2018).

[48] Sohini, Basu. “Evaluation of data centre networks and future directions.” (2017).

[49] Cisco Nexus 2000 Series Fabric Extenders Data Sheet, 2009

[50] Xia, Yiting et al. “A Tale of Two Topologies: Exploring Convertible Data Center Network Architectures with Flat-tree.” SIGCOMM (2017).

[51] Singla, Ankit et al. “High Throughput Data Center Topology Design” , 2014.

[52] Greenberg, Albert G. et al. “The cost of a cloud: research problems in data center networks.” Computer Communication Review 39 (2008): 68-73

[53]. Bilal, Kashif et al. “Green Data Center Networks: Challenges and Opportunities.” 2013 11th International Conference on Frontiers of Information Technology (2013): 229-234.

[54] https://perspectives.mvdirona.com/2010/09/overall-data-center-costs/

[55] Portland: a scalable fault-tolerant layer 2 data center network fabric, “in SIGCOMM, 2009.

[56] Hsieh, Ming. “Datacenter Traffic Control : Understanding Techniques and Trade-offs.” (2018).

[57] Mr. Mahendra N Suryavanshi, “ Comparative Analysis of Switch Based Data Center Network Architectures “, (JMEST) ISSN: 2458-9403 Vol. 4 Issue 9, September – 2017.

[58] T. Chen, X. Gao, and G. Chen, “The features, hardware, and architectures of data center networks: A survey,” Journal of Parallel and Distributed Computing, pp. 45 –74, 2016.

[59] Yu, Ye and Chen Qian. “Space Shuffle: A Scalable, Flexible, and High-Performance Data Center Network.” IEEE Transactions on Parallel and Distributed Systems 27 (2014): 3351-3365.

[60] Hammadi, Ali and Lotfi Mhamdi. “A survey on architectures and energy efficiency in Data Center Networks.” Computer Communications 40 (2014): 1-21.

[61] Mrs. Saba Siraj . “Network Simulation Tools Survey,“ 2017.

[62] Dr. L. RAJA, Amit J. Nayak . “ Study of various network simulators,” 2015.

[63] Ronit L. Patel, Maharshi J. Pathak. “ Survey on Network Simulators,” International Journal of Computer Applications (0975 – 8887) Volume 182 – No. 21, October 2018