ADDIS ABABA UNIVERISTY

SCHOOL OF GRADUATE STUDIES

COLLEGE OF NATURAL SCIENCES

DEPARTMENT OF INFORMATION SCIENCE

NETWORK TRAFFIC ANALYSIS FOR OPTIMIZING THE PERFORMANCE OF CRITICAL APPLICATIONS: THE CASE OF AAU NETWORK

FARIS AWOL

A THESIS SUBMITTED TO THE SCHOOL OF GRADUATE STUDIES OF ADDIS ABABA UNIVERSITY IN PARTIAL FULFILMENT FOR THE DEGREE OF MASTER OF SCIENCE IN INFORMATION SCIENCE

October 2015

ADDIS ABABA UNIVERISTY

SCHOOL OF GRADUATE STUDIES

COLLEGE OF NATURAL SCIENCES

DEPARTMENT OF INFORMATION SCIENCE

NETWORK TRAFFIC ANALYSIS FOR OPTIMIZING THE PERFORMANCE OF CRITICAL APPLICATIONS: THE CASE OF AAU NETWORK

By:-

Faris Awol

Advisor: Million Meshesha (PhD)

APPROVED BY EXAMINING BOARD:

Name	Title	Signature	Date
1. Million Meshesha (PhD)	Advisor	_____	October 23 2015
2. Workshet Lamenew (PhD)	Examiner	_____	October 23 2015
3. Gashaw Kebede (PhD)	Examiner	_____	October 23 2015

Dedication

I would like to dedicate my work in memory of my Dad (AWOL HASSEN) the most positive, lovable and cherish person I ever know. He is the greatest teacher in my life. May Allah place your soul in Jenna.

Acknowledgements

I would like to begin by thanking Dr. Million my advisor who was abundantly helpful and offered invaluable assistance, support and guidance throughout this work. His knowledge, patency and discipline inspires me more than I could ever imagine.

I would like to thank also Abiy Zemede AAU ICT Director for allowing me to access important resource and collect data without any limit and restriction. Even he help me by providing some procedures and guidelines. Generally without his continuous support this study wouldn't have been possible.

While at AAU, I get the chance to meet amazing, bright and talented staffs; classmates. I also like to thank all AAU Information Science staffs and my classmates for all your friendship, support and assistant especially Dr. Solomon for his humbleness and making tough situations comforting starting form even the day of my registration.

I would like to thank my Mam for her love and support. I also like to thank my beloved family. Finally I would like to thank everybody who has helping me to do this study. I express my sincere apology that I could not mention them personally.

Table of Contents

List of Figures

List of Tables

Abbreviation and Acronyms

AAU:	Addis Ababa University
HTTP:	Hyper Text Transfer Protocol
IP:	Internet Protocol
MAC:	Media Access Control
MPLS:	Multi-Protocol Label Switching
NAT:	Network Address Translation
OPNET:	Open Network
OSI:	Open System Interconnection
QoS:	Quality of Service
RMON:	Remote Network Monitoring
RTT:	Round Trip Time
SLA:	Service level agreement
SNMP:	Simple Network Management Protocol
TCP:	Transmission Control Protocol
ToS:	Type of service
UDP:	User Datagram Protocol
VPN:	Virtual Private Network
WAN:	Wide Area Network

Abstract

Nowadays almost all applications designed being connected to the internet in order to perform their basic tasks in addition to exchanging large volumes of information. This has resulted in an ever increasing need for effective application network optimization tools that can monitor the network. Inefficiency of the current AAU network system represents a significant financial and academic burden for the institution.

This study investigates detail application network related effects, behaviors and problems so as to propose suitable framework that will enhance the performance of applications on AAU network. The analysis phase of the study started after performing two vital tasks. The first one is selecting HTTP, Email, Video, Voice and Database as critical applications for the institution consecutively. Critical applications selected by interviewing system administrators and by identifying most used applications by users. The second one is simulating AAU network on OPNET modeler and analyze it for two month. The analysis help to know the magnitude of network effects on applications, application problems or bottlenecks and solutions or optimization techniques to the problem. Response Time, Delay, Delay Variation, TCP Retransmission count and Jitter used as a metrics.

The analysis shows that the network is experiencing application contention and application related network problems like protocol, latency and chattiness bottlenecks. In addition the analysis revealed that applications highly affected by network effects like when there is high amount of background traffic in the network, if the link is shared by applications and if the network link delay increases.

Based on the finding, this study proposes a network application optimization framework and algorithms for AAU network. The proposed framework are designed to improve the reliability, performance and delivery of critical applications across the network. The framework approaches the optimization first by analyzing the network condition to identify application network problem type after that it identifies the application running in the network then it apply optimization technique based on type of problem and application. Finally it monitors and checks the outcome of the optimization on the network.

The main challenge faced in this study is getting common metrics used to compare multimedia and data applications. Running the simulation for relatively long time needs huge computer capacity and processor as a result limited data and short simulation period of time used. Accordingly we recommended further research on this topic.

CHAPTER ONE

INTRODUCTION

1.1 Background

A computer network is a set of two or more computer systems and other computing hardware devices connected together through communication channels for the purpose of sharing resources. The most common resource shared today is connection to the internet. Other shared resources can include a file server or a printer [1]. Network broadly categorized in to wire and wireless network but it can be broadly categorized as Local Area Network (LAN), Personal Area Networks (PAN), Home Area Networks (HAN), Wide Area Networks (WAN), Campus Area Networks (CAN) and Metropolitan Area Networks (MAN) [1].

Data Networks have penetrated in our day-to-day life in a big way. Wherein we cannot think of any organization to run without or the other form of it. Networks have evolved themselves into forms never thought of before with a large demands for tools and technical staff to manage them. Companies have invested on their networks in a huge way and they are looking forward to achieve zero downtime for their network [2].

Network and Systems Management refers to the sum of all approaches, procedures and products for planning, configuring, controlling, monitoring and managing computer networks and distributed systems so as to provide user friendly, error-free and effective service using all the precautions and activities needed to ensure the effective and efficient use of the resources to be managed [3].

Modern organizations need a way to guarantee the delivery of applications that drive the growth and success of the business so network administrators face increasing challenges as they try to keep ahead of threats to the performance and productivity of critical business applications and services [4].

To overcome the complexity, growth, cost and problem of a network we need to investigate the network traffic, performance, security, operations and management on permanent base. Such analysis will enable us to design and apply optimization technique on the network.

Network analysis means the process of breaking down a complex project's data into its component parts (activities, events and durations) then plotting them to show their interdependencies and interrelationships [19]. Network analysis is a two way process. The first process will be done

before the implementation of the network. It is used to identify the specification and requirement of the company. The second analysis is done after the implementation of the network to evaluate the network performance by comparing expected and existed performance of the network. It is crucial to undertake both before and after network implementation analysis for the sustainability of the network.

Network traffic analysis is the process of recording, reviewing and analyzing network traffic for the purpose of performance, security and/or general network operations and management. It is the process of using manual and automated techniques to review granular-level detail and statistics within network traffic. Network traffic analysis is primarily done to get in-depth insight into what type of traffic/network packets or data is flowing through a network. Typically, network traffic analysis is done through a network monitoring or network bandwidth monitoring software/application. The traffic statistics from network traffic analysis helps to understand and to evaluate the network utilization, the network download/upload speeds and packet (type, size, origin, destination and content/data) [5].

Network security staff uses network traffic analysis to identify any malicious or suspicious packets within the traffic. Similarly, network administrations seek to monitor download/upload speeds, throughput, content and different statistics to understand network operations. Network traffic analysis is also used by attackers/intruders to analyze network traffic patterns and identify any vulnerabilities or means to break in or retrieve sensitive data.

As we all know, network is a very broad field and it holds various technologies. There are factors which affects the network for instance Latency, Firewall (different network security), Network accessibility, Different costs, Network architecture and Performance issues like, Bandwidth, Delay, Speed and Application Performance needs.

Network performance can be measured using metrics. Metrics are measurement unit used to measure the performance of the network. Metrics categorized broadly in to Network performance metrics, System performance metrics and Service performance metrics. System administrator can choose based on the need. for example if he/she wants metrics for diagnosing performance related problems he/she can choose metrics such as input queue drops, output queue drops, and ignored packets.

As pointed out by [1] Metrics are classified as lower is better (LB} which means the smaller value is better than the higher. For example if our network scores lower value we can say that the network is on good shape. The metrics included in lower is better are delay and error rate. Higher is better (HB) means scoring higher value is preferable than scoring lower. For example availability and

throughput, or nominal is best (NB) in this case the result is preferable if it is not either lower or higher.

Metric	Classification	Description
Availability	HB	Measure of what percentage of the time a network resource is available for use. Clearly, high availability is better because down time is not welcome.
Throughput	HB	A measure of how much data can be sent on or through a network resource in a given time period. Also referred to as available bandwidth. Higher throughput is certainly better from users' and administrators' points of view.
Utilization	NB	Measure of the usage of a link, port, or network resources. Nominal is best because high utilization is accompanied by high delays and low utilization is seen as a poor use of resources.
Delay	LB	The amount of time for a packet to traverse, either one-way or round trip, a network, network segment or network device.
Error Rate	LB	Usually refers to the percentage of packets or bits that contain errors on a network link, segment or device. High error rates can signal to an administrator that there is a problem in the network.
Congestion	LB	Excessive traffic on the network and creates a state of overcrowding this makes the traffic very slow or difficult
Data loss	LB	The amount of packet lost or failed to deliver to the destination.

Table 1.1: Performance metrics useful for network administration and engineering.

1.2 Overview of Addis Ababa University Network

Addis Ababa University (AAU), which was established in 1950 as the University College of Addis Ababa (UCAA), is the oldest and the largest higher learning and research institution in Ethiopia. Since its inception, the University has been the leading center in teaching-learning, research and community services [6].

Beginning with enrollment capacity of 33 students in 1950. AAU now has 48,673 students (33,940 undergraduate, 13,000 graduate and 1733 PhD students) and 6043 staff (2,408 academics and 3,635 support staff). In its 15 campuses, the University runs 70 undergraduate and 293 graduate programs (72 PhD and 221 Masters) and various specializations in Health Sciences [6].

As of now AAU has fifteen campuses. Thirteen of these are situated in Addis Ababa, and two of them located in Bishoftu and selalea. The campuses are Sidist Kilo Campus (Main Campus), Yekatit 12 campus (Department of Dental Medicine), CBE Campus, Yared School Campus, Amist Kilo Campus, Arat Kilo campus, Abune Petros Campus, Tikur Anbessa Campus, Commerce School Campus, Lideta Campus, Akaki Campus, Bishoftu Campus, Salale Campus, Art School Campus, Sefere Selam Campus.

Some of AAU campuses have their own data center but both the backbone connection to Ethio Telecom and the main data center located in Sidest killo. Sidest killo data center connected to Internet service provider (Ethio Telecom) to access WWW through fiber optics cable using Huawei router as a getaway. AAU ICT department controls and manages the overall network infrastructure and activities. Recently AAU campuses are having their own data center and this will decrease the traffic to the main campus. AAU uses various network equipment like routers, switches, wireless devises, core switches and servers (Email, DHCP, DNS and EGranary) and they are relatively old. In addition it uses application for network managing service, teaching learning and desktop application.

1.3 Statement of the problem

The use of computers has rapidly increased in the last few decades with this has been the exponential growth of the internet. Computers can now exchange large volumes of information. This has resulted in an ever increasing need for effective tools that can monitor the network [7].

In the context of our country bandwidth is very expensive and scarce that most organization cannot afford to have large amount of bandwidth. Even the internet service provider have shortcomings on providing internet services for customers who request big amount of bandwidth. So controlling this precious resource is vital for both parties. This can be done using network optimization techniques after making deep analysis on both the application and network performance.

Inefficiency of the current network system represents a significant financial burden for the institution. The university loses huge amount of money per month for service charges of the current bandwidth. This will force us to use the bandwidth effectively by implementing different network optimization techniques.

The campus currently experiencing various network problems major ones are described below

Computer Applications have no equal share on the network. Some application take high bandwidth and prevent other applications to use the network wisely. So other important educational connections like e-learning sites unable to use the network at the end of the day thereby the connection becomes very slow. The university network need to give priority to educational usage than entertainment.

AAU is now experiencing tough time in network infrastructure. The combination of different networking problem such as slow download and upload speed, low processing power or lack of latest devices, broadcast storm and different problems makes the network to be slow. To overcome this problem AAU officials tries to increase the bandwidth to 400Mb/S but this does not bring oversize change in the network as expected because it is impossible to solve every network problem by increasing bandwidth speed without considering other subjects.

Some protocols are not designed to be used in WAN network. Using those protocols cause latency, packet loss and bandwidth utilization problem. To control the negative effect of the protocols network optimization need to be deployed. AAU network don't have a single method implemented on the network to fill this gap.

Lack of modern networking equipment plays its own role in the networking problem. As preliminary investigation AAU uses old networking equipment both in the data center and in each campus network divisions. This causes different problems; for example the devices lack updated protocols to communicate with different WANs including ISP. This makes the network to become slow and unaware of latest updates including newest problem solution like latest packet switching, application aware network operating and packet inspections.

As we all know universities holds different kinds of stakeholders with different needs. For example, users can abuse the network by using nonacademic applications and AAU network infrastructure don't have the capability to manage applications based on their importance to the institute. So we have to consider the application that using the internet when we optimize the network.

Currently every application designed to be connected in order to update themselves without user authorization. For example antivirus software automatically connected to the internet to update

themselves, in addition to applications communicate to social networks creates a burden to the campus networks, If every application needs to connect the internet or the Local area network, possibly problem may occur unless some optimization practices executed. The user can also able to help us on the way of managing the network by installing software which control application network access but most of the users don't have the knowledge to do this or lack awareness and willingness. On AAU network situation most of the user didn't know which application uses the network even there are background applications which use the network the user don't know about. This increases the burden to the network. In addition not only computers connected to the network but also GPS, smart phones and different measurement devices all the devices are frequently connected to the network, which is affecting the performance of the network.

Bearing in mind the Bandwidth limitation and cost, as well as the low resourced network infrastructure in the campus, deep ingestion and improving the performance problem of critical applications is necessary. But, this is impossible without detail WAN studying so the current situation worth studying.

Different studies done to explore and improve the performance of AAU network. The research done by Tsegaye [83] focused on the performance of the network. Other studies such as Awet conducted to understand user navigational behavior as to the researchers knowledge there is no study attempted to explore the application side of AAU network therefore the aim of this study is to simulate and analyze the application behavior and problems on AAU network. To this end this study attempts to explore and address the following research questions.

- What are the critical and top applications currently running on AAU network?
- What are the behaviors of critical application showing when network condition change?
- What are the problems affecting applications not to use its performance on full-scale and there implication?
- What suitable optimization techniques can be designed for enhancing Application network performance?

1.4 Objective of the study

1.4.1 General Objective

The general objective of this study is to investigate the behavior of critical applications and problems on AAU network so as to propose suitable framework that ensure to enhance the performance of applications in AAU network.

1.4.2 Specific Objectives

To achieve the general objectives the following specific objectives are formulated.

- To review literatures so as to understand the current sciences and technologies related to this study.
- To identify critical applications for AAU.
- To inspect the behavior of critical application showed under different network circumstances.
- To identify application bottlenecks in AAU network.
- To classify applications by combining both application behavior and criticality.
- To propose a Framework that can enhance application performance of the network.
- To evaluate what the proposed Framework improves on the network.

1.5 Scope and limitations of the study

The study only intended to analyze critical applications identified from AAU network. Which is HTTP, Email, Database, Video and Voice. Applications other than critical applications not analyzed. In addition network factors which have no direct relation with application not covered for instance security policies, client and device operating system, infrastructure problems, user behaviors, quality of the devices, protocols and compatibility issues.

The result and solution which identified by network application analysis is implemented or applied only on the network which the analysis is done. If we analyze a network in some institution the analysis result/solution can only be applied in that institution only because the analysis result may not be identical with the other network and it is rare to find similar network architecture and behavior in two different networks so this study results and optimization applied only in AAU network.

This study only focus on wired network. Wireless network subjects not covered. The study only uses the following metrics which used to measure both the performance of the network and application. Received and Sent Bits per second, Packets per second, Errors Rates, Response Time, Dropped packets, Flows per second, Delay (RTT) and Jitter (delay variation). To make the study

complete as much as possible additional measurement used. For instance service level agreement between the campus and Internet Service Provider used. Service level agreement is a written agreement between a service provider and their customers on the expected performance level of network services. It consists of metrics agreed upon between the service provider and its customers. The values set for the metrics must be realistic, meaningful, and measurable for both parties. This study also use measurement on System performance metrics or device level performance metrics like CPU utilization, buffer allocation (big buffer, medium buffer, misses, hit ratio), and memory allocation.

Even if the simulated result extracted in every link but the online data gathered mainly from the main campus of AAU (Sidist Kilo) in addition the study considers the main data center is located in Sidist Kilo.

1.6 Methodology of the study

This section explains each step the analysis considers in order to achieve the objective of the study In addition it discusses what measures the research takes in order to test the study's hypothesis.

1.6.1 Research design

The study is performed by following a serious of steps which is described in section 1.6.5. Tools used to collect and analyze data. Most of the analysis made on controlled environment. By combining all the statistics together the study tries to find out what the problem really is and Proposed a frame work used to optimize applications in AAU network.

The study generally contains three elements Data collection plus preparation, Experimentation and Evaluation. Data collection and preparation done based on a thorough understanding of the experimentation. Data which is necessary to the experimentation collected for instance device configuration, log data, application model, network topology and device specification. The experimentation phase done repeatedly to expose and diagnose application performance issues on the network and to determine the problems related to application on AAU network. Finally the proposed framework evaluated based on predefined metrics.

1.6.2 Literature review

Literatures like (books, journal articles, research reports Conference papers and the Internet) reviewed to increase the knowledge and understanding about WAN optimization, network analysis (traffic), network analysis tools, network protocols, end user transaction tracing and monitoring, network modeling, generally about application and network performance management

1.6.3 Data collection

In this study an attempt is made to pursue the analysis of traffic statistics based on data collected over AAU to know critical applications, usage pattern, backbone links rates, configurations and necessary information's to execute the analysis of applications. Based on the dataset there is a valuable contribution aiming at the longitudinal study of the traffic. Where the characteristics are investigated at different OSI layers. The field of traffic analysis is very vast which encompasses data collection, statistical analysis, and prediction and pattern recognition analysis [10]. All the data necessary for experimentation collected from both from AAU network and simulation environment. The experimentation totally depend on the collected data.

1.6.4 Evaluation Metrics

Evaluation is the application of scientific method to the study to determine effectiveness and fairness of the result. Evaluation techniques have been developed to accurately measure the effectiveness with which network system resource are managed while striving to provide service. The study uses two separate sets of metrics used for data applications and multimedia application.

Most of application bottlenecks may occur on Data applications who uses TCP as a protocol. The current application problems existed in AAU network can be resolved using application aware network optimization and this can make applications to operate faster and more effective than the current performance.

The metrics used to evaluate both the performance of the network and the performance of application. The metrics are the following

For Data applications (Database, Email, HTTP)

- Response Time (Sec): Time elapsed between sending a request and receiving the response packet. Measured from the time when the Application sends a request to the server to the time it receives a response packet.
- Traffic Received and Sent (Packets/Sec): Average bytes per second forwarded to all specific Applications by the transport layers in the network.
- TCP Delay (Sec): Delay of packets received by the TCP layers in the complete network for all connections. It is measured from the time an application data packet is sent from the source TCP layer to the time it is completely received by the TCP layer in the destination node.
- TCP Retransmission Count: Total number of TCP retransmissions in the network. Written when data is retransmitted from the TCP unacknowledged buffer.

For Multimedia Applications (Video and Voice)

- Packet Delay Variation: Variance among end to end delays for packets. End to end delay for a packet is measured from the time it is created to the time it is received.
- Packet End-to-End Delay (Sec): The time taken to send a video or voice application packet to a destination node application layer. This statistic records data from all the nodes in the network. But for voice application it is the total voice packet delay called "analog-to-analog" or "mouth-to-ear" Delay = *Network delay + Encoding delay + Decoding delay + Compression delay + Decompression delay*. Network delay is the time at which the sender node gave the packet to RTP to the time the receiver got it from RTP.
- Traffic Received and Sent (Packets/Sec): Average bytes per second forwarded to all specific Applications by the transport layers in the network.
- Jitter (Sec): Arrival time difference between two consecutive packets this will have major impact on the quality of both video and voice applications.

1.6.4 Optimization Approaches

The application network optimization focused on optimizing the performance of applications on the network. Network and application optimization refers to an extensive set of techniques that organizations have deployed in an attempt to optimize the performance of networks and applications as part of assuring acceptable application performance. The primary role these techniques play is to Reduce the amount of data sent over the WAN, Ensure that the WAN link is never idle if there is data to send, Reduce the number of round trips necessary for a given transaction, Mitigate the inefficiencies of older protocols and offload computationally intensive tasks from client systems and servers.

The study proposed application optimization framework by investigating the core problem then well know optimization technique tried to solve the problem for example Protocol optimization, compression, prediction and rout smoothing then better application optimization for AAU network selected. The effectiveness of the optimization also tested finally the optimization framework evaluated using simulation tool.

1.6.6 Procedures

The study follows the procedures depicted in Figure 1.1

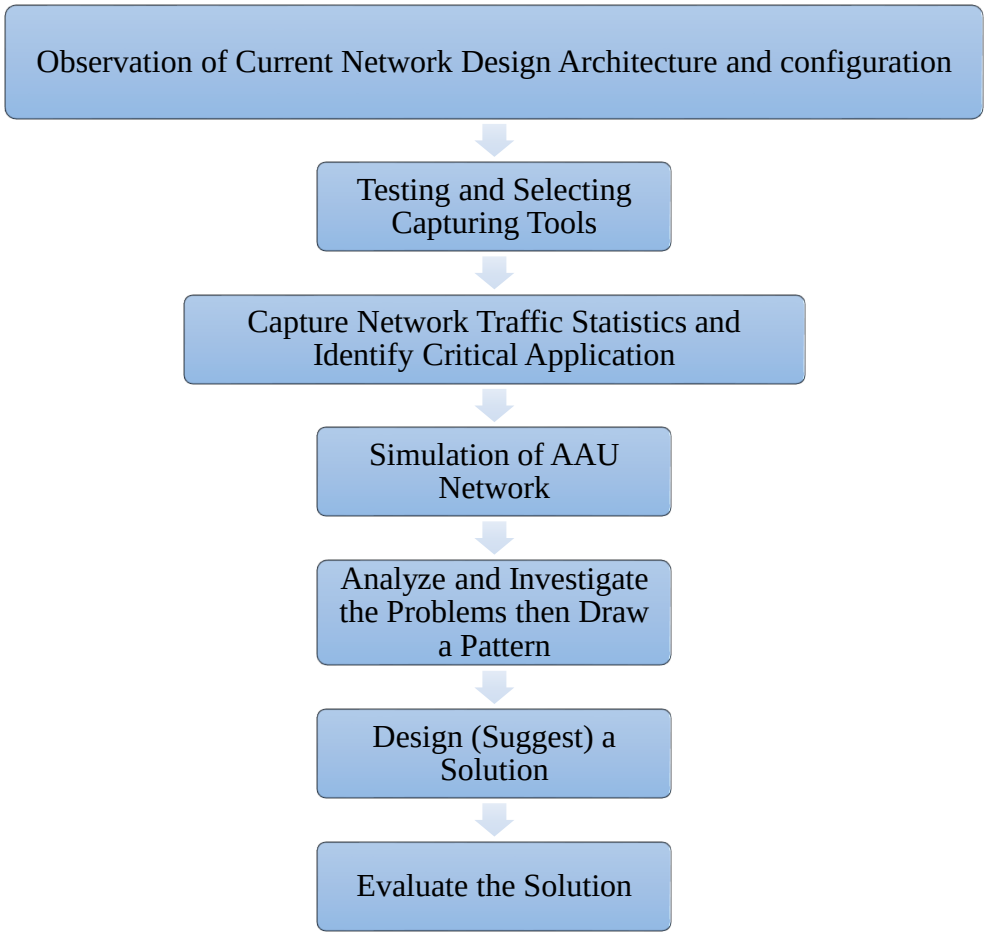


Figure 1.1: The procedure of the study

Observation of current network design architecture and configuration: Observation made on AAU network to understand the network design, architecture and configuration to the possible extent. This help us to get knowledge about AAU network.

Capturing Tools: Testing and selecting different network performance capturing tools based on the situation. In this stage we learn one important concept which is there is no perfect tool to analyze, evaluate and identify expected facts on the network so different capturing tools used.

Capture Network Traffic Statistics and identify critical applications for AAU: Real time network traffics was captured by using Sflow, NetFlow, Solarwinds and Wireshark tools for two month.

Simulation of AAU network: This is the most important stage of the study. Simulation take over by using the observation and collected data on AAU network. This stage help us to identify the problems and solutions.

Analyze and investigate the problem then draw a pattern: This state totally done on the simulation environment by applying network effects like background traffic, application when sharing WAN link and changing link delay. The nature of the problem and the solution to the problem identified on this stage.

Design (Suggest) a Solution: based on the analysis result application network optimization framework proposed which used to resolve observed application bottlenecks and problems.

Evaluate: The proposed application optimization framework evaluated by comparing the result before and after the implementation of the optimization framework. AppDoctor applicatioan used to evaluate the proposed framework by comparing the number of bottleneck before and after implementation.

1.7 Significance of the study

The study attempts to analyze the application network problems and issues related to applications which faced in AAU network then suggested an optimization framework so the institution can use the proposed framework as an input on the application network optimization process. In addition it can help the ICT office of AAU to set rules and improve its policy based on the application nature and criticality which recommended by this study so as to increase the reliability, Performance and delivery of critical applications across the network.

Generally speaking the study can be used as major input for several purposes some of are the following. It helps on planning and coordinating resources related to application on AAU network. It can be used as a tool and solution for boosting utilization of critical application accessing the network resources. The study can help the network to be managed on more accurate and scientific way since the study based on concert survey and facts.

This study may help future researchers to do researches on the topics recommended by this study and it also help them to understand the facts related to applications on AAU network.

1.8 Organization of the Thesis

The whole thesis is divided into seven chapters. The first chapter discusses the background of the study and the study area, statement of the problem, general and specific objective, scope and the limitation included also in introduction chapter then methods used on the study discussed in methodology section finally the significant of the study covered.

On the second chapter different literature related to the study reviewed. On the third chapter related works studied on network traffic analysis and measurement discussed.

The fourth chapter contains topics about experimentations which conducted on the study. The network architecture of AAU discussed on detail also discussed in this chapter in addition the measurement techniques, the metrics and the environmental setup also described.

The fifth chapter covers the result and discussions of the analysis. The behaviors and effects showed up by applications when the network condition changes on AAU network also explained in fifth chapter finally the problems and solutions also elaborated on detail.

The six chapter contains the proposed application optimization framework and clarify how the framework work on detail, pseudo code, algorithm and validation of the proposed framework also included in chapter six. The last chapter presents the conclusions and recommendations.

CHAPTER TWO

LITERATURE REVIEW

Computer network is a connection when more than one computer system connected through link to share resources with other computer system. To access the network computer system can use two type of medium wireless technology that goes by the name wireless local-area network (wireless LAN), or Wi-Fi, instead of using wired connection [11].

To understand or describe network components on more understandable way it is better to use networking models because it is easy to understand using model. Sometimes models also called either a networking architecture or networking blueprint, refers to a comprehensive set of documents. Individually, each document describes one small function required for a network; collectively, these documents define everything that should happen for a computer network to work. Some documents define a protocol, which is a set of logical rules that devices must follow to communicate. Other documents define some physical requirements for networking. For example, a document could define the voltage and current levels used on a particular cable when transmitting data [12] [13].

There are two type of models In the history of networking which is Transmission Control/Internet Protocol (TCP/IP) and Open Systems Interconnection (OSI)[12] [11]. In this study we used TCP/IP because it is the most used model currently and it support in almost all of the Operating Systems from mobile phones to mainframe computers,

To help people understand a networking model, TCP/IP breaks each model functions into a small number of categories called layers. Each layer includes protocols and standards that relate to that category of functions. TCP/IP actually has two alternative models, the first one contains Four layers (Application, Transport, Internet and Link) the second one contains seven layers (Application, Transport, Network, Data Link and Physical). Application layer provide services to the application running on the computer. Transport contains two protocol TCP and UDP it provide services to the application layer protocols that reside one layer higher. Network layer provides several features most importantly addressing and routing. Data link and physical layer defines the protocols and hardware required to deliver data across some physical network.

2.1 Network Traffic Theory

Researchers agree that any network traffic is organized into data packets, but they have different answer for the question what is a data packet. Even starting from the name packet, some of them pronounce it as a Frame others call it Segment and Bit. Cisco definition resolve the difference by

categorizing the data unit according to the OSI network model [13] [11], the model puts Segments in layer four (transport layer), Packet in layer three (network layer), Frame in layer two (data link layer) and finally Bits in layer one (physical layer).

Since this research focus on IP based network traffic analysis, we use the name Packet as a data unit. This is because IP based network traffic analysis can be done on layer three with logical addressing is used.

All network traffic is organized into data packets. Each packet contains header and payload information. A collection of packets, grouped by a set of common header fields, is called a flow. For example, a typical aggregation is the IP-flow, which refers to the aggregation of data packets sharing the same addresses and ports. Finally, a collection of flows between two end-points corresponding to the same application or user transfer is called a session [11].

The packet traffic observed on a single network element arises from the superposition of flows, which in turn arise from the superposition of user sessions. The arrival of flows in a session reflects user activity and demands and therefore is relatively independent of network conditions. Within a session, flow behavior is determined by the application. For example, a web transfer session generates constituting IP flows that differ substantially from a telnet session. Thus systematic dependencies arise across these three aggregations of traffic, which have been explored when characterizing and constructing models for traffic behavior.

Traffic is typically expressed as a time series, and modeled as a discrete time stochastic process, i.e., some attribute of traffic over time. For instance, one may be interested in studying the total number of packets that arrive every millisecond at a router for an hour long period. The resulting time series can be expressed as a discrete stochastic process $X = \{x_1, x_2,\ldots..,x_n\}$ where x_i denotes the number of packets(or number of bytes) at the i^{th} millisecond bin. Characterizing and modeling this stochastic process is difficult because there are a number of properties and parameters that must be estimated. The methods proposed in the literature to characterize and model X depend on whether the intended application is performance evaluation or network management [1].

2.2 Network Traffic Analysis

The field of traffic analysis is very vast which encompasses data collection, statistical analysis, and prediction and pattern recognition analysis [14]. The data to be analyzed depends on the availability of network traffic measurements. Traffic measurements in operational networks aid in understanding the traffic characteristics in deployed networks, developing traffic models and evaluate performance of protocols and applications. Computer network measurements provide network operations, its development and research with information regarding the network

behavior. The reliability and credibility of this information directly affects the quality of these activities, and thus the perception of the network and its services [14] [15].

In addition to Schormans, and Timotijevic [16]definition wrote The purpose of network traffic analysis which is used to know the network performance, different security issues, general network operations and management, network traffic analysis have various advantages but the advantage highly various based on our needs for example some companies do analysis to find out their security issues some others to know the network usage, in our context network traffic analysis used to understand the general network problem and application network usage.

Schormans, and Timotijevic [16] noted that study network analysis holds data collection, statistical analysis, and prediction and pattern recognition analysis but they didn't put the time frame which used to do the analysis. jie sun [17] recommended that it is necessary to collect network traffic on a continuous basis rather than as a onetime event which only captures transient behaviors that provides insight into network problems, longer term traffic measurement need to be carried out, because of Internet-based technologies, applications and protocols are constantly being developed and applications are updated quite frequently this reason will have an impact on the quality of network analysis. Analysis can be considered as at the time of analysis it is also important to the future researchers to use the previous analysis to say something about the old network.

Some network professionals spent huge amount of time trying to answer the question of who is generating the most traffic, what protocols are used, where is enormous amount of traffic originate and where it lands. Network traffic analysis helps on the way of answering the above questions. journal[5] inscribed network analysis can be used us cheaper solution to network problem rather than implementing expensive solution; for instance, upgrading network infrastructure, i.e. replace servers with high end servers and increase the bandwidth [18].

Based on Crovell Network Traffic Modeling, our discussion of traffic is organized along the two principal reasons for studying and modeling network traffic [20]: which is for performance evaluation and for network management.

Performance analysis seeks to understand traffic so as to address questions related to equipment design, such as studying packet throughput, delay and loss characteristics. As such, performance evaluation typically deals with network traffic at very small timescales (less than one hour), works with traffic at the granularity of individual links, and focuses on volume features, such as the number of bytes and the number of packets.

Network management on the other hand, seeks to analyze traffic so as to address issues that concern network operators, such as capacity planning, and traffic engineering. Many of these

problems require studying traffic at longer timescales (hours to weeks), and typically at a topological granularity of multiple links. Further, network management problems require the ability to extract meaning from multiple traffic features, beyond volume metrics, such as header and payload information [21].

2.2.1 Traffic Analysis for performance Evaluation

Network traffic measurement gives the designer and opportunity to adjust and control the quality of service mechanisms of modern telecommunication networks [22].The quality of service means the overall performance of a telephony or computer network, particularly the performance seen by the users of the network. Traffic analysis will help us to evaluate our network performance in accurate way. In our country organization put wrong performance evaluation to their network by lacking traffic analysis.

Traffic analysis for the purpose of performance evaluation have many dimensions. Different authors put many list of performance evaluation metrics; but the most important idea they agree on is, performance evaluation varies based on our need which means the network administrator can use specific evaluation scenario based on his specific problem.

Category	Metric	Units
Productivity	Throughput and Effective Capacity	MBPS
Responsiveness	Delay, Round Trip Time and Queue Size	Milliseconds Packets
Utilization	Channel utilization	Percentage of Time Busy
losses	packet loss rate and frame retries	loss percentage
buffer problems	AP queue overflow and play out buffer underflow	Packet drops rebuffed events

Table 2.1: Computer Network Performance Evaluations

In contrast to the above performance evaluation metric, some authors [17] [23] put Traffic load, throughput and delay as the major performance evaluation techniques. Traffic load means the number of packets put into a network during a time interval. Throughput on the other hand is the

number of bytes transmitted through a network during a time interval. Delay is the amount of time that a packet takes in transmission in a network.

Another important concept in performance evaluation is that the time scale traffic analysis for the purpose of performance evaluation is very short starting from Nano seconds up to maximum one hour.

There are literatures use the properties of traffic at such short timescales [24]. In fact analyzing traffic on a single link over short timescales, and in isolation of other links, has been the dominant paradigm of traffic analysis studies to date. Many important results have emerged from such investigations, such as the celebrated findings of self-similarity and long range dependence [25], their impact on provisioning network resources [26], and, finally traffic models that capture these features [27].

2.2.2 Traffic Analysis for Network operation

In contrast to the extensive research done by the performance evaluation community, network operations has received little attention, and demands a different style of traffic analysis, network operations encompasses a large set of tasks from attack detection to capacity planning to traffic engineering. Broadly speaking, these tasks require studying traffic at longer timescales hours to weeks, using more sophisticated views than simple byte or packet counts and finally often demand looking across more than single link traffic.

The central problem when studying traffic on longer timescales is that traffic is no longer stationary. As such, the models developed for performance evaluation cannot be applied, when working with backbone link traffic; a number of methods are available to remove the non-stationary trends. For instance, [28] [27] proposed using wavelets to extract the dominant frequencies, which corresponds to the diurnal cycles. Nucci et al [1] extracted the diurnal non-stationary trends by using the dominant Fourier frequencies further Roughan [32] et al proposed using moving filters to remove the non-stationary trends in backbone link traffic. Seen in a general light, all these methods are essentially unilabiate time series analysis methods that extract the diurnal cycles as dominant frequencies.

Crovella and Kolaczyk [30] were the first to step beyond unilabiate analysis of traffic time series and studied traffic across links. Their approach analyzes link traffic at a specific point in time by comparing it to traffic on adjacent links, using an extension to the Haar Wavelet, which they call graph wavelets. They show that this multivariate approach is particularly useful for a number of problems in network operations.

A central granularity of traffic that has considerable importance to network operations is the set of Origin-Destination (OD) flows that traverse a network. The traffic in an OD flow is the collection of traffic that enters the network at a particular location; for example, routers and leaves the network at a different location. Together the set of OD flows provide a view of how traffic moves across the network, and are therefore of great interest to network operation, however, OD flows have not received much attention in the research literature in part because they are difficult to measure, and in part because they are complex structures to analyze.

Two measurement studies that depart from the link-level characterization examine OD flows in a commercial Tier-1 backbone instead are [1] [14]. The authors of [26] shows how to measure such traffic flows, by relying on sampled data; the authors of [1] observed many different types of OD flows, which behaved differently depending on link speed, type of relationship(peer or customer) and popularity. The implication is that it is difficult to devise a single model (or even a family of models) that characterizes a general OD flow.

Other studies have focused on analyzing simple statistics of OD flows, For Instance, studies have observed that most of the traffic in a network is exchanged by handful of OD flows, whereas the majority of OD flows carry little traffic [41]. This phenomenon has been labeled the “elephants and mice” effect. Understanding this effect is of interest especially for traffic engineering tasks, where identifying the predictable and heaviest flows to be routed preferentially is a key problem [15].

Broadly speaking, the body previous work in traffic analysis for network operations is small with most studies largely focus in on analyzing traffic volumes at longer, non-stationary timescales. There is comparatively little work analyzing traffic across links, and analyzing multi-feature traffic, both of which hare needed by network operators today.

2.3 Network Traffic Modeling

The design of robust and reliable networks and network services is becoming increasingly difficult in today's world. The only path to achieve this goal is to develop a detailed understanding of the traffic characteristics of the network [39].

Computer networks such as local area and wide area networks possess complex characteristics due to the heterogeneous nature of the supported traffic. The network traffic exhibits highly irregular fractal-like structure and long term correlations. Various stochastic processes such as fractional Gaussian noise, multiplicative cascades and linear fractional stable motion have been proposed to model network traffic. These stochastic processes are relatively unheard of in the networking

community, until recently. This paper provides a thorough review of these stochastic processes and their application to wireless traffic modeling [83].

From the viewpoint of a service provider, demands on the network are not entirely predictable. Traffic modeling is the problem of representing our understanding of dynamic demands by stochastic processes. Accurate traffic models are necessary for service providers to properly maintain quality of service. Many traffic models have been developed based on traffic measurement data [83].

Analysis of the traffic provides information like the average load, the bandwidth requirements for different applications, and numerous other details. Traffic models enables network designers to make assumptions about the networks being designed based on past experience and also enable prediction of performance for future requirements. Traffic models are used in two fundamental ways: (1) as part of an analytical model or (2) to drive a Discrete Event Simulation (DES) [25].

The different traffic models each have its own pros and cons. The type of network under study and the traffic characteristics strictly influence the choice of the traffic model used for analysis. Traffic models that cannot capture or describe the statistical characteristics of the actual traffic on the network are to be avoided, since the choice of such models will result in under-estimation or over-estimation of network performance.

There is no one single model that can be used effectively for modeling traffic in all kinds of networks. For heavy-tailed traffic, it can be shown that Poisson model under-estimates the traffic [18]. In case of high speed networks with unexpected demand on packet transfers, Pareto based traffic models are excellent candidates since the model takes into the consideration the long-term correlation in packet arrival times [20]. Similarly, with Markov models, though they are mathematically tractable, they fail to fit actual traffic of high-speed networks.

Other than the traffic models discussed in this report there are numerous other traffic models that are used widely for traffic modeling. There are different categories of traffic models like stationary and non-stationary types. Stationary models can further be subdivided into models that are referred to as Short-range dependent and Long-range dependent types. Each model varies significantly from the other and is suitable for modeling different traffic characteristics.

A number of factors come into play while evaluating the efficiency of a traffic model. In general, the factor that differentiates one model from the other is the ability to model various correlation patterns and marginal distributions. Traffic models should have a manageable number of parameters, and parameter estimation should be simple; and, models that are not analytically tractable are preferred only for generating traffic traces.

Balakrishnan [25] categorized traffic model in to five, Poisson Distribution Model, Pareto Distribution Process, Weibull Distribution Process, Markov and Embedded Markov Models(ON-OFF and IPP Models, Alternating State Renewal Process, Markov Modulated Poisson Process, Markov Modulated Fluid Models) Autoregressive Models.

One of the most widely used and oldest traffic models is the Poisson Model. The memory less Poisson distribution is the predominant model used for analyzing traffic in traditional telephony networks [22]. The Poisson process is characterized as a renewal process. Ina Poisson process the inter-arrival times are exponentially distributed with a rate parameter λ: $P\{An \leq t\} = 1 - \exp(-\lambda t)$. The Poisson distribution is appropriate if the arrivals are from a large number of independent sources, referred to as Poisson sources. The distribution has a mean and variance equal to the parameter λ.

Markov models attempt to model the activities of a traffic source on a network, by a finite number of states. The accuracy of the model increases linearly with the number of states used in the model. However, the complexity of the model also increases proportionally with increasing number of states. An important aspect of the Markov model - the Markov Property, states that the next (future) state depends only on the current state. In other words the probability of the next state, denoted by some random variable Xn+1, depends only on the current state, indicated by Xn, and not on any other state Xi, where i<n.

2.4 Application performance analyses

There are many approaches to analyze application performance; this section describes different approaches used to analyze application performance in addition different researches related to application analysis.

Ejaz Ahmeda and his team [43] analyze the effect of network-centric parameters on the application migration process. The performance of the migration process is analyzed by simulating the migration process in OMNeT++. The effects of various parameters, such as number of users in a WLAN, size of a file containing the application and its running states, traffic load on the wireless access point, message length, number of hops to the cloud, and mobility speed, are studied on the application performance metrics such as application migration time and packet drop ratio. The analysis shows that the application and its running states migration time is affected by the changes in the network conditions. Based on the research findings, they recommend application execution framework designers to incorporate the network-centric parameters along with other parameters in the decision process of the application migration.

On another paper Ouyang and Hosein[28] introduces the architecture of Universal Mobile Telecommunications System (UMTS) packet switched (PS) network and then they applies multivariate statistical analysis to Key Performance Indicators (KPI) monitored from network entities in UMTS PS network to guide the long term capacity planning for the network. The approach proposed in there paper is helpful to mobile operators in operating and maintaining their 3G packet switched networks for the long run.

Krishnan and his team [27] present a methodology for performance analysis of the interconnection network, with focus on video and multimedia benchmarking. They describe a typical video decoder based SoC system, and describe the traffic profiles for each of the processing engines. They also provide performance analysis measures of interest in a video decoder based SoC.

The team addressed the performance benchmarking problem of Network-on-Chip architectures. They motivated the need for a performance benchmarking infrastructure, which interconnection network designers can utilize to measure the goodness of their architecture. Their focus in the research has been on performance benchmarking for SoCs targeted at video applications such as HDTV and set-top box. They also presented methods to analyze the performance of an interconnection architecture, by taking the performance goals and gate count into account. They concluded that future work need to focus on other verticals in the scope of interconnection architecture design, such as wireless, automotive and multi-processor architectures [27].

According to David and Koziniec [49] an approach to Computing Response Times Computing application-level response times is usually a complex task, especially if it involves background load. The task is especially complex in the case of two-tier client/server applications. However, it is useful and relatively easy to calculate baseline end user response times under the assumption of unloaded network resources. These baseline response times are useful in troubleshooting WAN performance problems for client/server applications. As per David and Koziniec[49] The parameters used for computing response times are total number of bytes transferred in any one direction, segment size, receiver advertised window size, optimal window size for the connection, preferred end user response time (in seconds), one-way propagation delay and protocol overhead factor((S + Overhead bytes) /S).

On Pavel Masek research they elaborate different concepts on multimedia network [1], Multimedia networks is an emerging subject that currently attracts the attention of research and industrial communities. This environment provides new entertainment services and business opportunities merged with all well-known network services like VoIP (Voice Over Internet Protocol) calls or file transfers. Such a heterogeneous system has to be able to satisfy all network and end-user requirements which are increasing constantly. Therefore, there is a need for simulation tools

enabling deep analysis to find key performance indicators and factors which influence the overall quality for specific network service, the paper provides a study on the network parameters like communication technology, routing protocol, QoS (Quality of Service) mechanism, etc. and their effect on the performance of hybrid multimedia network. The analysis was performed in OPNET Modeler environment and the paper shows that proper choice of the routing subsystem for multimedia networks is influenced by many factors such as particular network topology, level of network load and composition of operated application have to be taken into account.

The network utility ping is commonly used to investigate connectivity issues. Troubleshooting a database distributed application, for example, ping can tell if the server database is running and accessible to an application and data base Client. Ping uses Internet Control Message Protocol echo packets to echo a network location and provides round-trip times and details about packages lost, if any.

For a more comprehensive analysis Cristian [48] suggested that it is better to use database own ping command to test and to measure the response time of the network infrastructure between a database Client and a database server. Database ping uses an already established connection with database to simulate a Client request and to instruct the server to send an answer through the network. It also provides a quick view of the end-to-end network performance between the Client and the database server.

Cristian [48] concluded that database ping is a handy and easy to use db2 command that can be exploited to validate a connection to a database server. It provides a consistent way of measuring the end-to-end network performance used by database applications. During troubleshooting its results can be used as a key performance indicator to detect, or to eliminate from analysis, a network performance issue.

2.5 Network Traffic Analysis Tools

Network performance could be measured using either active or passive techniques. Active techniques (e.g. Iperf) are more intrusive but are arguably more accurate. Passive techniques are of less network overhead and hence can run in the background to be used to trigger network management actions [49].

From hundreds to thousands of computers, hubs to switched networks, and Ethernet to either ATM or 10Gbps Ethernet, administrators need more sophisticated network traffic monitoring and analysis tools in order to deal with the increase. These tools are needed, not only to fix network problems on time, but also to prevent network failure, to detect inside and outside threats, and make good decisions for network planning.

Network monitoring and measurement have become more and more important in a modern complicated network. In the past, administrators might only monitor a few network devices or less than a hundred computers. The network bandwidth may be just 10 or 100 Mbps (Megabit per second); however, now administrators have to deal with not only higher speed wired network (more than 10 Gbps (Gigabit per sec) and ATM (Asynchronous Transfer Mode) but also wireless networks. They need more sophisticated network traffic monitoring and analysis tools in order to maintain the network system stability and availability such as to fix network problems on time or to avoid network failure, to ensure the network security strength, and to make good decisions for network planning.

2.5.1 NetFlow

"NetFlow" is developed by Cisco Systems [12]. Cisco routers with NetFlow switching feature can generate network flow records and be exported in either UDP (User Datagram Protocol) or SCTP (Stream Control Transmission Protocol) packets to NetFlow collectors. NetFlow record is defined as version number (version 5 is commonly used and version 9 is an IETF (Internet Engineering Task Force standard for IPFIX (Internet Protocol Flow Information export)), sequence number, input and output interface SNMP indices, timestamps for the flow start and finish time, number of bytes and packets observed in the flow, IP (Internet Protocol) headers (Source and destination IP addresses, Source and destination port numbers, IP protocol, Type of Service value), the union of all TCP flags observed over the life of the flow [50] [26].

The network flow information is very useful not only to understand network behavior, to detect security holes, but also to make good decisions on network planning. For example, source and destination addresses can be used to determine who is originating or receiving the traffic. The application utilizing or distributing can be made from port information. Class of service examines the traffic priority. The packets and byte count show the amount of traffic. Flow timestamps are used to calculate packets and bytes per second. Next hop IP address with BGP (Border Gateway Protocol) shows routing information. Network prefixes can be calculated from subnet mask of source and destination address. The union of TCP flags can implicitly explain a TCP handshake process [26] [14].

Some routers also support more flow information such as source and destination Autonomous System (AS) number. NetFlow version 9 includes all of these fields and optionally includes extra information, such as Multiprotocol Label Switching (MPLS) labels and IP version 6 addresses and port numbers. NetFlow version 9 was also chosen to be a common, universal standard of export for IP flow information from network devices by IPFIX (IP Flow Information Export) IETF working group [26].

2.5.2 SNMP

Simple Network Management Protocol (SNMP) is defined by IETF. SNMP is an application layer protocol used to monitor network-attached devices. SNMP works as the manager/agent model. The manager and agent use a Management Information Base (MIB) and a relatively small set of commands to exchange information. The MIB is organized in a tree structure with individual variables, represented as leaves on the branches. A long numeric tag or object identifier (OID) is used to distinguish each variable uniquely in the MIB and in SNMP messages.

SNMP uses five basic messages (GET, GET-NEXT, GET-RESPONSE, SET, and TRAP) to communicate between the manager and the agent. The GET and GET-NEXT messages allow the manager to request information for a specific variable. The agent, upon receiving a GET or GET-NEXT message, will issue a GET-RESPONSE message to the manager with either the information requested or an error indication as to why the request cannot be processed. A SET message allows the manager to reconfigure to the value of a specific variable. The agent will acknowledge with a GET-RESPONSE message to indicate the change or provide an error message to why the change cannot be made. The TRAP message allows the agent to inform the manager of an important event [15].

Each SNMP element manages specific objects with each object. Each object / characteristic has a unique object identifier (OID). The OIDs are the combination of numbers separated by decimal points such as "1.3.6.1.4.1.2682.1". The OIDs form a tree structure. The MIB associates each OID with a readable label such as "dpsRTUAState" and various other parameters related to the object. The MIB then serves as a data dictionary used to assemble and interpret SNMP messages [15].

2.5.3 Software sniffer (snoop, tcpdump, Wireshark)

In most operating system, the bundled packet sniffer is provided; however, either software (e.g. Microsoft Windows) or hard ware (HP-UX and Solaris) has to be purchased. "snoop"is a simple packet capture tool which is bundled on Solaris operating system. "snoop" is a command line interface and display the packet in text (a summary and multi-line format). The drawback of "snoop" is that it does not reassemble IP fragments. "nettl/ netfmt"[nettl/netfmt00]is the packet sniffer provided by HP-UX but still in command line. "Microsoft Network Monitor" [MNN06] is the packet sniffer which is bundled with Microsoft Windows. This "sniffer" must be run on Windows NT Server 4.0 or Windows Server 2003, or have Microsoft Systems Management Server installed. This "sniffer" provides the simple graphic user interface. All "sniffer" provided for each operating system can run either in real-time and in batch modes (Logging is saved to a file for further analysis). A simple analyzer is also included for filtering and protocol searching.

"tcpdump" [tcpdump06]is a packet sniffer mainly bundled in Linux operating systems, but also has a lot of distribution with other operating systems, such as Solaris, BSD, Mac OS X, HP-UX and AIX. "WinDump" can be used in Windows. Like "snoop" and "netttl/netfmt", "tcpdump" runs on standard command line and output to common text file for further analysis. "tcpdump" uses a standard libpcap library as an application programming interface to capture the packets in user level WinPcap for Win32 platform).Although all packet sniffer can examine the traffic in real-time, the processing overhead is also higher, so it might cause the packet drop. As a result, it is recommended to output raw packets and do some analysis later. However, the problem is the incompatible of the trace format such as "Microsoft Network Monitor" cannot read the trace file from "tcpdump".

"Wireshark"(formerly Ethereal), this free packet sniffer is much like "tcpdump"; however, it provides a user-friendly interface with sorting and filtering features (a command line version of the utility is "Tshark"). "Wireshark" supports capturing packets in both from live network and from a saved capture file. The capture file format is libpcap format like that in "tcpdump". It supports a various kinds of operating systems such as Linux, Solaris, FreeBSD, NetBSD, OpenBSD, Mac OS X, other Unix-like systems, and Windows. It can also assemble all the packets in a TCP conversation and show you the ASCII (or EBCDIC, or hex) data in that conversation. Packet capturing is performed with the pcap library. The need of promiscuous mode (root permission) still remains.

2.6 Related Works

In this section, previously conducted researches in areas related to the concept of network traffic analysis are presented and deeper discussion on various proposed traffic analysis techniques is made. Also, specific works which is related to our proposed techniques and have relevant in terms of their objective reviewed.

2.6.9 Network Analysis to optimize WAN performance

The research done by Tsegaye [83] focused on the performance of AAU WAN network. The study investigates the effects of different WAN factors using performance analysis tools. Based on the analysis the study develop a WAN optimization framework that can improve the performance of AAU network traffic flows over the WAN link. By using real-time cases the study collect, analyze and evaluate the data. The network performance investigated using metrics such as network availability, response time and packet loss. Based on the analysis result a WAN optimization framework is developed.

The study find out that AAU network is experiencing some problems like high response time rate, high packet loss and fluctuating network availability. The WAN optimization framework is developed to solve the real time bottleneck status of the WAN link and apply optimization techniques. Even if the study tries to develop WAN optimization framework but the framework only evaluated using interview. It is hard to evaluate a framework using interview because we can't able to tell the magnitude of effectiveness.

2.6.8 Network Performance and Application Analysis

Related work done specifically on application is by Yakob Gobena[82] for his masters thesis in 2013 by the title "Developing WAN Optimization Model to improve the Performance of Business Critical Application, on UNECA". Yakob hypothesized application behave differently in LAN and in WAN environment and he tries to explain this by simulating the UNECA (United Nation Economic Commission) Network. The WAN network connects five branch offices located in the head quarter in Addis Ababa then the rest located in five sub regions of Africa: North Africa, West Africa, Central Africa, East Africa and Southern Africa. Then he analyze deeply the problems of applications in simulated environment. He discovers three application bottlenecks which is protocol Overhead, Delay and Connection Reset. Then he diagnose the bottlenecks using AppDoctor finally he develops Application WAN optimization Model which envisioned to apply a performance improvement technique based on the applications real time behavior revealed on the enterprise WAN environments.

Even if the paper tries to resolve application network related to WAN but it is unrealistic means on the simulation phase he use real network documentation for the network only located in Addis Ababa but the rest of branches located in four other countries don't follow real network documentation.

The study didn't analyze the bottlenecks specific to individual applications. He just identify the bottlenecks to all application together. This will lead as to false conclusion about the problem because application follow different client server architecture. In addition it doesn't show as the bottlenecks specific to the application.

Finally the paper didn't put anything which used to validate the proposed optimization model so we can't able to know whether the proposed optimization model solves the bottlenecks or not.

2.6.1 Internet traffic measurement and analysis

Measurement and analysis of Internet traffic is important to get knowledge about the characteristics of the traffic, thus it has drawn a lot of researchers attention. Some researchers made long-term investigation of Internet. Borgnat and his team collected internet traffic data for 7 years

in order to sketch the evolution trend of the internet traffic [13]. Fomenkov and Keys have investigated more than 4000 traces from 1998 to 2003, to find relations between bit rates and traffic statistics [64]. Traffic pattern is a very clear and typical way to display traffic variation within the recording period, like what K.Thompson and G.J.Miller did [21]. They had made experiments to reveal the traffic characteristics in terms of packet sizes, flow duration and volume over the two time scales, one day and seven days.

It is of great interest to find analytical models for describing Internet traffic. Karagiannis, Molle, and Faloutsos in their paper [65], reviewed 10 years development of Lang-Range Dependence (LPD) theory and use LPD to model the complex traffic of the Internet. However, more researchers believed that, instead of bit and packet rate, flow level traffic would be better used for explaining the intrinsic characteristics of Internet. Barakat analyzed TCP flow by means of Markovian model in a differentiated service network [67]. They also established a Poisson Shot-noise model in flow level. As a matter of fact, modeling the traffic at the packet level has proven to be very difficult [66], because traffic on a link is the result of a high level of multiplexing of numerous flows whose behavior is strongly influenced by the transport protocol and by the application. It is not easy to judge which model is more ideal for the Internet, it all depends on which application the model is used. For example, detection of anomalies (e.g. denial of service, link failure) require an accurate traffic model. While in a protocol and application agnostic environment, a more general model is needed.

More people would like to analyze the Internet from the view of application. Back in the nineties, FTP and Mail accounted for half of the traffic volume, until HTTP becomes the majority [68]. And the invention of Peer-to-Peer (P2P) nearly toppled the pattern of Internet traffic, it could be considered as “killer Internet application” [12]. The Internet service providers, on the other hand, are reluctant to see this change as P2P consumes huge amount of bandwidth resource. And they react, inclining to interfere their customers‟ file sharing [59]. But the technologies seem to keep in pace, while modern P2P application uses random port numbers, making itself hard to be detected from authorities and Internet service providers who have the illegal P2P file-sharing concern[70], which may cause inaccurate P2P traffic measurement.

2.6.2 Network Traffic Analysis

Tomas Sisohore on his paper [61], Analyze and characterize IPTV user behavior by doing traffic measurement and analysis in fixed and mobile broadband access networks on Swedish network.

The research discover that the primetime for user activity was between 17:00 and 22:00, The ratio of usage of HTTP media stream by different customers was skewed with 10% hosts responsible for about 50% of the number of connections and traffic volume. More than 40% of the active hosts

visited the YouTube website. Less than 5% of active hosts accessed the examined internet radio sites.

In other research on the same network by Phat Hoang [62]. The research discover the following fact, P2P file sharing was the majority of the traffic generated. The outbound traffic volume from P2P file sharing was about 97% of the total traffic during the week of 2007 that was analyzed. Peak traffic occurred between 19:00 and 21:00.The daily traffic pattern during 2007 was more symmetric than in 2009, which could be seen as a result of P2P becoming more popular. The amount of streaming media increased very fast from 2007 to 2009. Inbound traffic from. Streaming media was only 4.1% of the total traffic in 2007, but increased to 22% in 2009. Spotify was one of the factors in the growth in the amount of streaming media based on P2P technology, Four games were analyzed: Second Life, World of Warcraft, Counter Strike, and Warcraft III, but they generated very little outbound traffic.

2.6.3 Analysis of Network Parameters

In Network analysis it is hard to select the type of network but Pavel Masek and his team [55] evaluated two communication technologies–passive optical network and classic TCP / IP network. Even though both these technologies are quite different and cannot be directly compared,

Pavel and his team did a study on network parameters like communication technology, routing protocol, QoS mechanism, etc. and their effect on the performance of hybrid multimedia network. The emphasis was put on the identification of the key network elements or parameters that have the greatest effect on the basic QoS characteristics like E2E delay, jitter and throughput.

The work analyzed, tested and evaluated routing optimization protocol, Network traffic classification on VLAN(Virtual Local Network) and implementation of QoS(Quality of Service) by using several simulation scenarios in the OPNET Modeler environment

The research used OPNET modeler 17.5 for simulation purpose, The goal of performed simulation was to identify the network parameters that have the greatest impact on the performance of multimedia network operated over TCP/IP network Mainly to evaluate the delay and throughput , As the output, the network characteristics such as end-to-end delay, jitter and throughput were discussed, the simulation model defines the following applications: voice calls(VoIP)), videoconferencing, web browsing (HTTP), database access (SOAP), file transfer (FTP).

The obtained results clearly shows that even though OSPF generates higher traffic overhead and greater procedural delays within the network nodes, it achieves significantly better results in the case of high-loaded network. Because of its load balancing and faster convergence so it have higher throughput and lower delay.

The overall research find that Dynamic routing protocol should be enabled and Implemented QoS mechanism does not have a favorable effect only on E2E delay or jitter, but also on the overall network throughput.

However the research use very short simulation time this can lead to incorrect conclusion in addition the research only considers two routing protocols RIP and OSPF but there are other well-known routing protocols need to be considers on the research like EIGRP and BGP. Plus they give 10mb download it is hard to know whether one routing protocol is better that the other by considering only 10mb of data.

2.6.4 Measurement and Analysis of Network Traffic

In campus network applications use a large proportion of the bandwidth by establishing number of connections to the outside network Shinya Furuta in her paper [57] try to clarify whether the behavior of an application influences the number of TCP connections by measuring and analyzing the change in this number over time. When the company uses NAT (Network Address Translation) one or two public IP address will be converted to many to cover IP address requirement, when we use NAT we have to consider the characteristics of an application in order to set a large quantity of connections using the type of service suitable for a number of users accessing through a single global address.

The observation of network traffic uses two kinds each of Firefox and Internet Explorer as Web browsers during use in Web Service Web Browser, on other side Thunderbird and outlook are used as e-mail client applications, Bit Torrent as P2P file sharing software, the research use Windows Live Messenger as an instant messenger (not audio communication only text chatting) finally iTunes Store as Music/Movie Player Application.

The paper analyze the following four items Transition of the number of connections by time, Transition of the number of requests for the connection by time, Distribution of the connected time of the connection, Distribution of the request intervals of the connection.

The paper recommends it is not desirable from a security viewpoint for an IP address to be fixed, and Shinya Furuta propose a solution for application connection issue which is Unified multiplex communication Architectures Its mechanism provides privacy and security by generating an address dynamically and handling it as a disposable address, the address type of Unified communication is the client identifies a session using only EA, which is dynamically generated for every session and then thrown away at the end of a session

The paper developed Unified Multiplex communication architecture to cope with such large numbers of connections, regardless of the number of address pools considered maximum. As mentioned above, we clarified that about 700 address pools are sufficient for operations.

Even if there are many application throughout the network the paper only investigate the performances of simultaneous connections with a Web browser, an e-mail client application, P2P file-sharing software, an instant messenger, and animation and jukebox software in addition it is hard to analyze the permanent behavior of an application using connection type because single application may use variable connection type for different reasons.

2.6.5 Network Traffic Data Analysis

Network traffic is a cornucopia of data and the approaches involved in measuring, analyzing, predicting and modeling solely depends on the end use of the data. Harish Kapri on his paper [58] analyzes and visualizes the network traffic data both at packet and flow connection level, the paper try to address various concerns related to the percentage of TCP/UDP traffic, protocol categorization, load vs. throughput, distribution of the flows and packets, length of the flows and TCP statistics such as Round Trip Time (RTT) and the retransmissions.

The research use the following tools to measuring, analyzing, predicting and modeling the traffic data, Tcpdump(for reading captured packet traffic), CoralReef(for pre-processing the captured traffic traces), Tcptrace(for analysis of TCP dump files), Tcpstat(for packet statistics), Perl(for scripting to normalize the data), and Matlab(for statistical analysis and visualization of the results)

Harish Kapri analyzed the data from three different perspectives: on the per-packet, per-flow and per-connection levels. Further, he have confirmed several well-known facts about the general nature of the traffic such as, Zipf-type nature of distribution of number of flows and heavy-tallness of flow sizes. Further, we tried to derive and evaluate analytical models that would fit the data best.

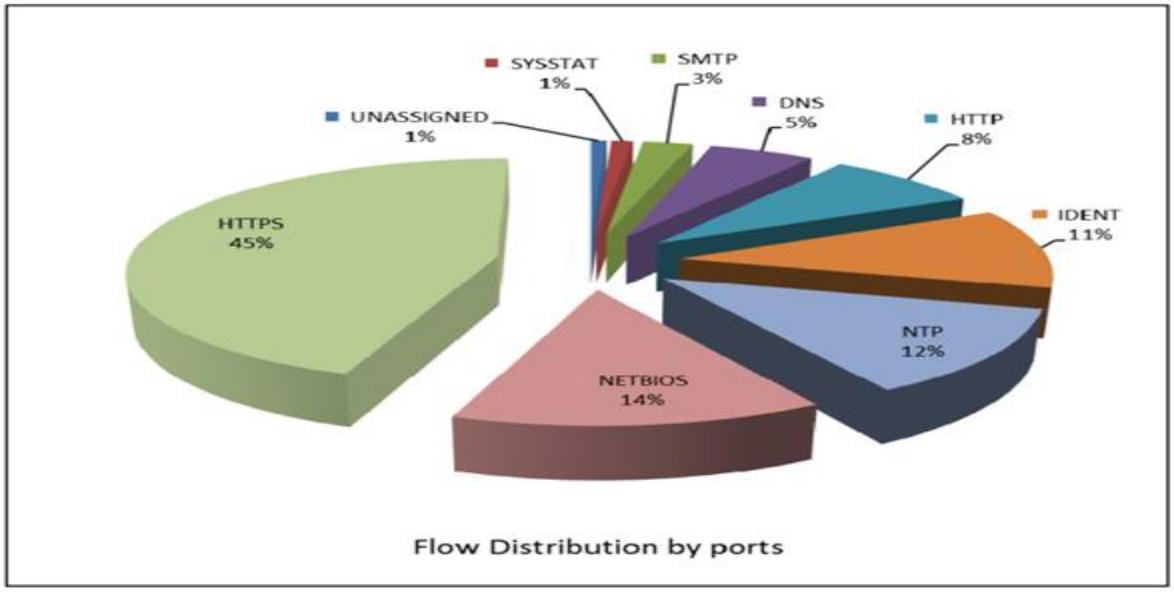


Figure 2.1: Flow Distribution by Ports

However the paper used IP based analysis since IP address changes dynamically if the network used DHCP so it is difficult to identify and trace a specific user by using IP address, in addition the researchers process packet distribution by port manually even if there are tools to do that job, finally the research analyze packet length distribution but it is pointless to analyze packet length distribution because network devise have a unit called MTU (maximum transmission unit) which used to govern the unit of packet a device allowed to send at a time if it varies the devises will not communicate each other so it is unnecessary to analyze uniform packet length.

2.6.6 Internet Traffic Analysis Based upon Traffic in an IP access network

Jie sun [56] did another research based on Phat Hoang and Tomas Sisohore recommendation he analyze and characterize traffic in a local multi-service residential IP network in Sweden, he user parameters such as Parameters such as traffic patterns (e.g., traffic volume distribution, application usage, and application popularity) and user behavior (e.g., usage habits, user interests, etc.) at different geographic localities were studied in this project.

The research categorized data processing in two parts data connection and analysis. On data collection Packet logic used to collect data from network devices collection location information, application and service protocol .On processing first rule of data collection created then objects created finally packet logic Statistics processed.

By analyzing 11.3 TB in downlink and roughly 17.1 TB in uplink directions, The majority of the traffic volume was P2P file sharing, which comprised 40% of the downlink traffic and 76% of the uplink traffic the research identifies three services P2P File sharing(Bit Torrent) which is

responsible for roughly 50% of both the downlink and uplink traffic, HTTP media streaming was the top most media streaming protocol and generated approximately 45% of the downlink traffic of different media streaming protocols. Flash video over HTTP was the second highest volume media streaming protocol and it accounted for 40% of the downlink media streaming traffic, other network services, on other network services SSL v3 and SSL v2 were the two protocols accounting for roughly 50% of the total remaining traffic in both downlink and uplink directions.

In addition the paper discover application popularity which means the list of applications based on network usage, the following chart describes the applications based on the network use.

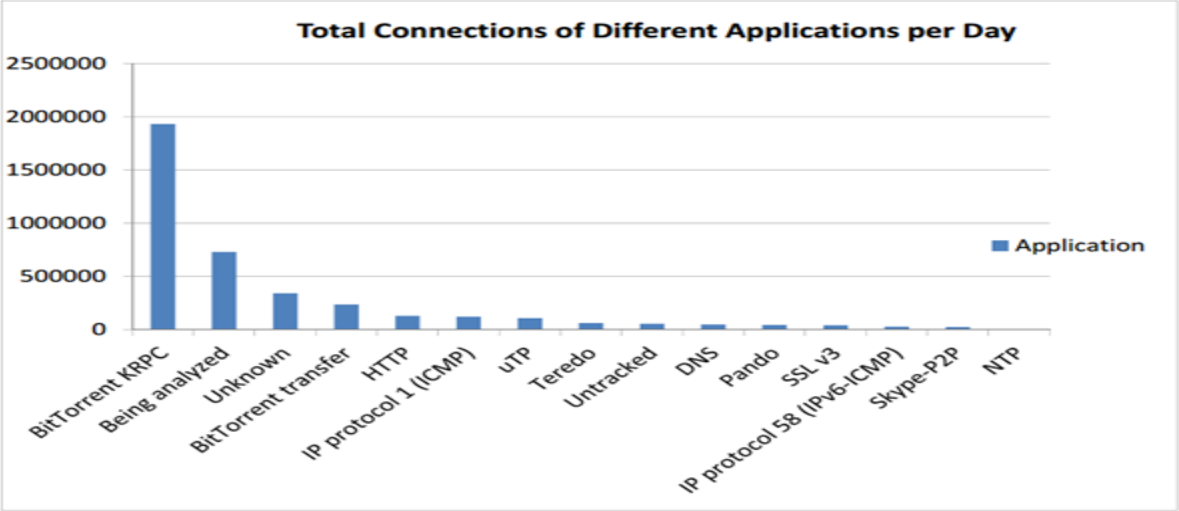


Figure 2.2: Total Number of Connections Observed of Different Applications per Day

However the research only consider packet with limited delay this will decrease the accuracy of jitter and delay measurement , in addition the research put a reason the data is too large to be fully analyzed two of the datasets were randomly selected and analyzed on a half-hour basis, on the first place the research didn't put with random selection method did used on second place it is hard to believe the result obtained by analyzing randomly selected data plus Most of the data is collected in common hours this will reduce the number of unique users which have network access on other than pick hours finally The paper didn't put satisfactory reason how uplink greater than downlink.

2.6.7 Residential Network Traffic and User Behavior Analysis

YICHI ZHANG [23] believes it is crucial to understand today's network traffic and the usage patterns of the end users, which will lead to more efficient network design, energy and costs savings and improvement of the service offered to end users so he did a research on network traffic analysis on residential network to find hidden patterns of traffic and user behavior, the research develop, a systematic framework of traffic measurement and analysis to address the problem. The

research involves PacketLogic traffic data collecting, MySQL database storing and traffic and user behavior analysis by using Python script, the research approach provides new insights on residential network traffic properties and internet user habits of households and it covers topics of aggregated traffic pattern, household traffic modeling, traffic and user penetration for applications.

The research procedure starts by collecting data from PaketLogic, and stores the collected data a data warehouse then Python script combined with MySQL queries run to extract pattern.

The analysis found out that outgoing traffic (uplink for households) always exceeds the amount of incoming traffic (downlink for households). Weekly traffic patterns were examined, but it turns out there is not a big difference between traffic on weekdays and weekends. Weibull distribution appears to be the better model for incoming traffic.

By utilizing PacketLogic deep packet inspection function, the analysis identified application level protocols, and thus could identify specific application traffic. Six categories of applications were defined: web browsing, Instant Messaging, media streaming, online game, P2P file sharing, and other applications. Bit Torrent is the dominant application in the P2P file sharing category, accounting for 90% of all application traffic classified in this category. Flash over HTTP accounts for roughly 50% of incoming media streaming traffic. But for outgoing traffic, P2P based media streaming such as Spotify, PPStream and SopCast account for the most network traffic.

However the research considers only one Internet Service Provider even if the residential network in the research area use more than one Internet Service Provider, this will decrease the result accuracy in addition the research use IP DHCP(Dynamic Host Configuration Protocol) which by its nature it is dynamic and it can’t represent one household for a period of time.

CHAPTER THREE

EXPERIMENTATION

A thorough description of research area network is given in this chapter, besides explanation where and how the data is collected and simulated. In addition how we measured and compared the performance and efficiency difference is also discussed in this chapter.

4.1 Overview of the Network Architecture

The network studied on this study is an enterprise network in AAU. It is university-wide network designed to interconnect all campuses of AAU such as Sidist Kilo, Amest Kilo, Arat Kilo, Faculty of Business, Medical Faculty, Collage of Commerce, Building College, Media institute, School of Fine Arts, Medical Lab School Nursing Paster, School of Nursing Paulos, School of Nursing Zewditu, Veterinary Faculty and Dental Training Center. It offers broadband internet access as well as some other services such as Email, Web Hosting, digital library and institutional applications to its users who are students, Teachers and academic staffs. The broadband speed is asymmetric that means the uplink speed is lower than the downlink. In addition AAU network didn't use QOS which used to separate users and give priority for selected users.

AAU Network is designed based on central network architecture which is the gateway, Security and remote so as to facilitate other network services to be distributed among the campuses. According to the network architecture there are fifteen campuses and from this Sidist Kilo (Main Campus and Faculty of Business and Economics - FBE), Amist Kilo (Faculty of Technology) and Arat Kilo (Faculty of Science) are interconnected together. The networks which interconnect the four campuses use Sidist Kilo as a gateway to the internet.

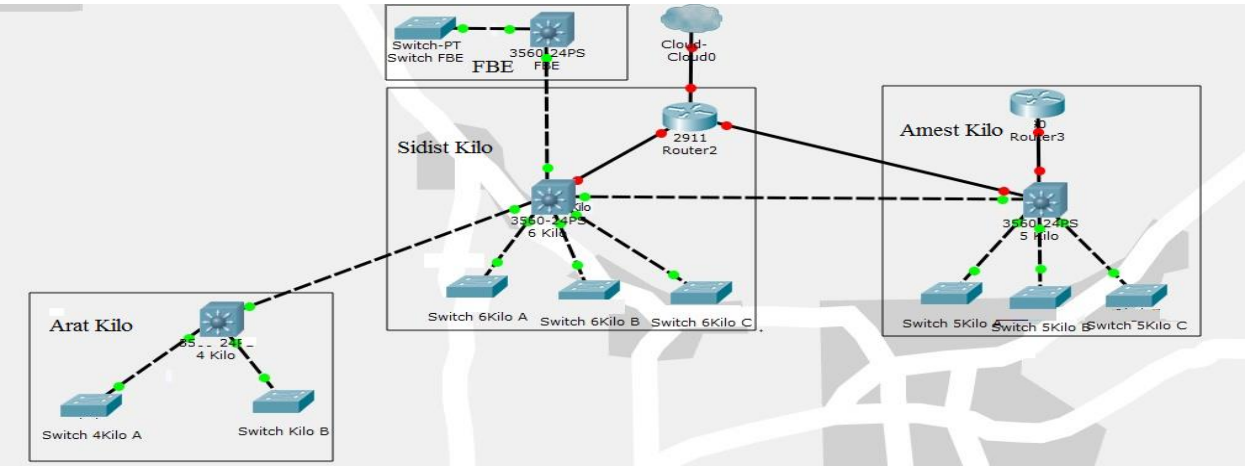


Figure 3.1: Actual Network Design of AAU (Sidist Kilo, Amest Kilo, FBE and Arat Kilo)

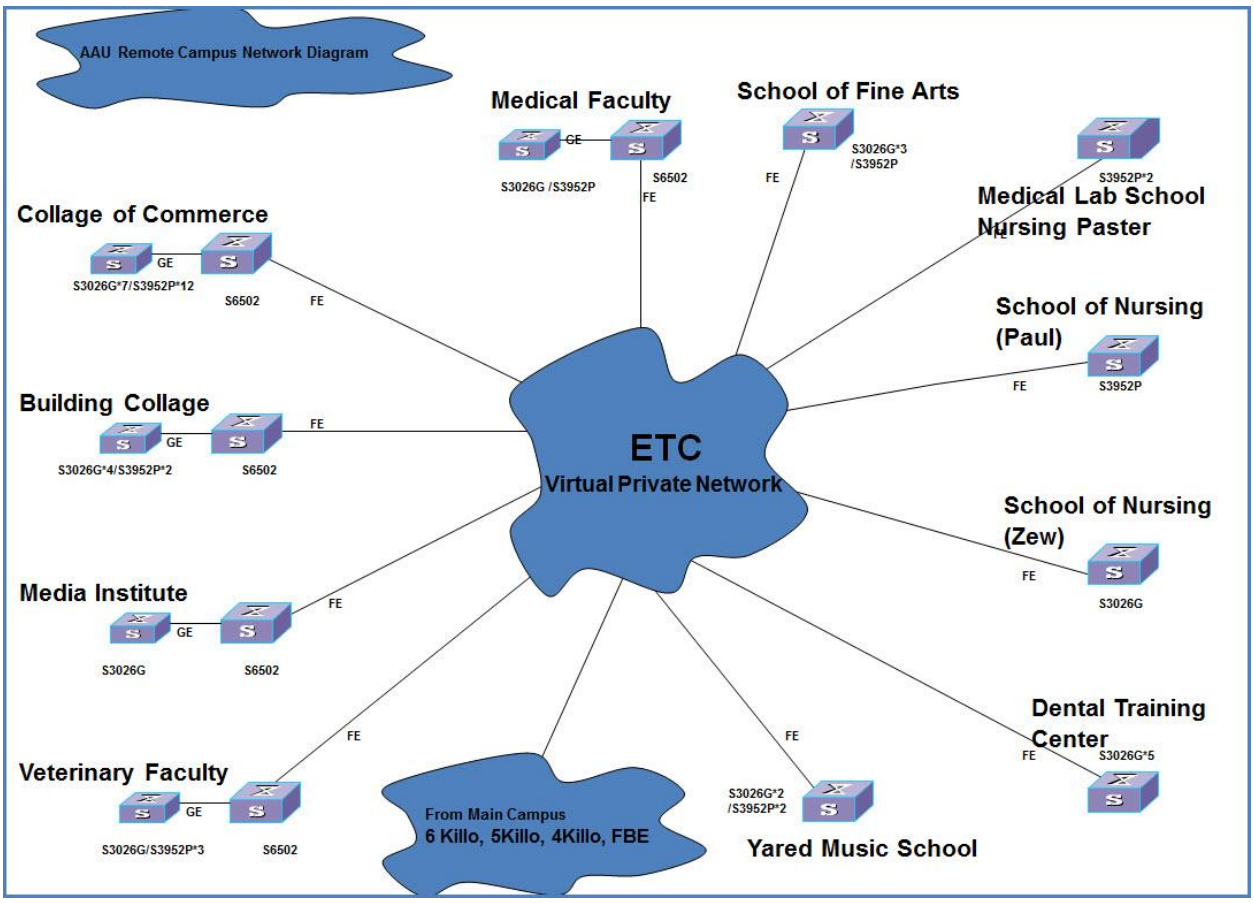


Figure 3.2: AAU Network design interconnecting remote campus

The diagram shows the architecture of AAU network used to connect remote campuses together. AAU Network have two getaway one used to connect to the internet located in Sidist Kilo and the other one is used to connect campuses through VPN which is located in Amest Kilo.

AAU network uses various devises to fit the business requirement of the institution. From those devises server play a vital role for providing services to the end-user. Most of the servers placed at the gateway to make them accessible from LAN and WAN. The campuses have a mail server which used to give mail service in addition DNS and DHCP servers used to resolve and to give IP address to the client.

The network totally controlled and managed by AAU ICT office. The office is in charge of the network security, services, expansion and additional authorities related to the network infrastructure. To keep the network from inside and outside attack the office uses proxy server. The office have enforces different rules and laws by using proxy server.

Other crucial network devices such as DNS and DHCP that are not visible to the end user but that are critical to the operation of the network infrastructure. Since the campus have massive amount network user it use class A private IP address and the IP address distributed by subnet based on the requirement.

To enhance network security for the network environment and control various possible network attacks it uses firewall server used to blocks unauthorized access to an organization's LAN and also Antivirus server that provides premium protection against viruses, spyware and Internet crime.

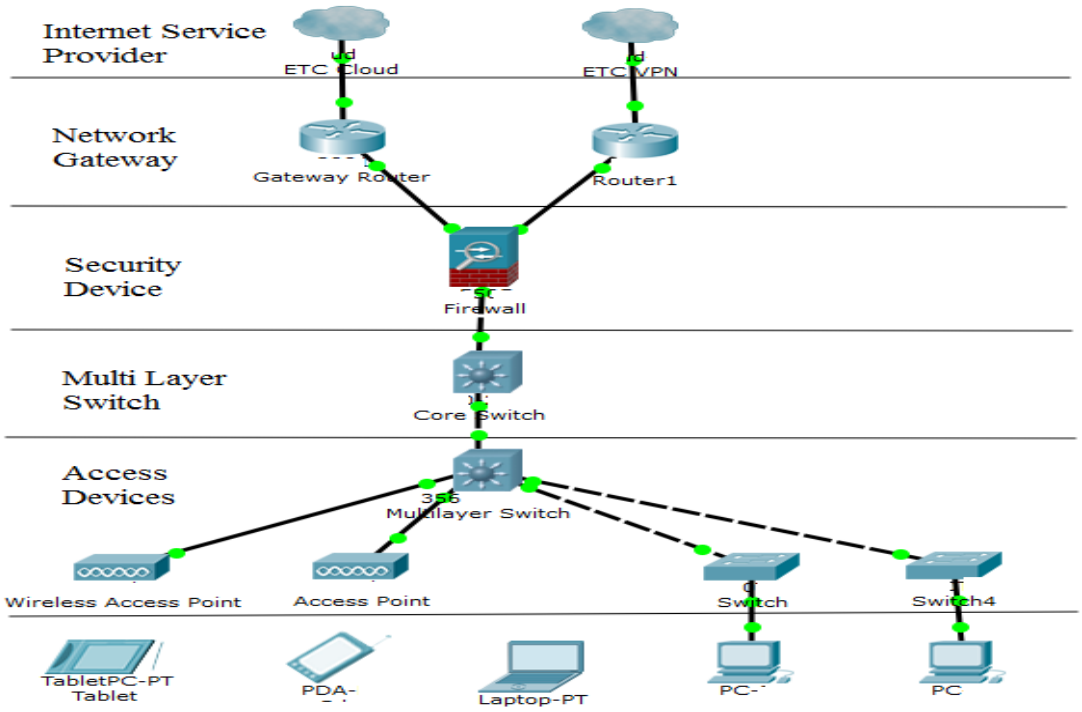


Figure 3.3: Network Architecture of AAU

Figure 3.3 shows the hierarchical architecture of AAU Network. To access the internet clients use devices like Desktop computer, Laptop Computer, Personal Digital Assistant, Tablet PC and Phone. Those devices found in the lower layer of Figure 3.3. This devices directly connected to access switches. Access switches take the traffic to core switch without any modification. Devices including routers and layer three switch have the capability to route the packet by looking IP address. This capability allow them to select a path and forward the packet to the correct destination. Finally the firewall takes the network security issue. Generally all devices can be categorized in to Access devices, Multilayer switches, Firewalls and routers. The measurement equipment for this study deployed between layer four switches and gateway router this is because

all the traffic which need routing will pass through router and most of the traffic is in need of accessing the internet.

4.2 Experimentation Design

4.2.1 Measurement techniques and Equipment

Measurement techniques can be classified based on the performance they measure. Metrics can be monitored at both the link-level and the end-to-end level. The corresponding techniques can be further labeled as link-level or end-to-end level monitoring techniques. At link level network connectivity can be monitored using both physical-layer signals and IP-layer routing protocol messages; while at the end-to-end level it is monitored using ping or *traceroute* [72].

Most of the measurements and results assimilated from online traffic environment in addition to that some results taken from preinstalled tools like proxy server and monitoring tools like solar winds which collects and analyze results on daily bases for the past couple of years.

The delay metric is measured using ping at both the link and end-to-end levels. However link-level ping can only be done through the router command-line interface which means manually. Latency calculated by using this formula *((Ping Response Time – (Hop Count * (Ping Packet Size/Data Rate)))/2)*. The formula will help us to get more accurate result than calculating latency by dividing RTT in to two. RTT formula is not accurate most the time when the network uses symmetric connection to service provider.

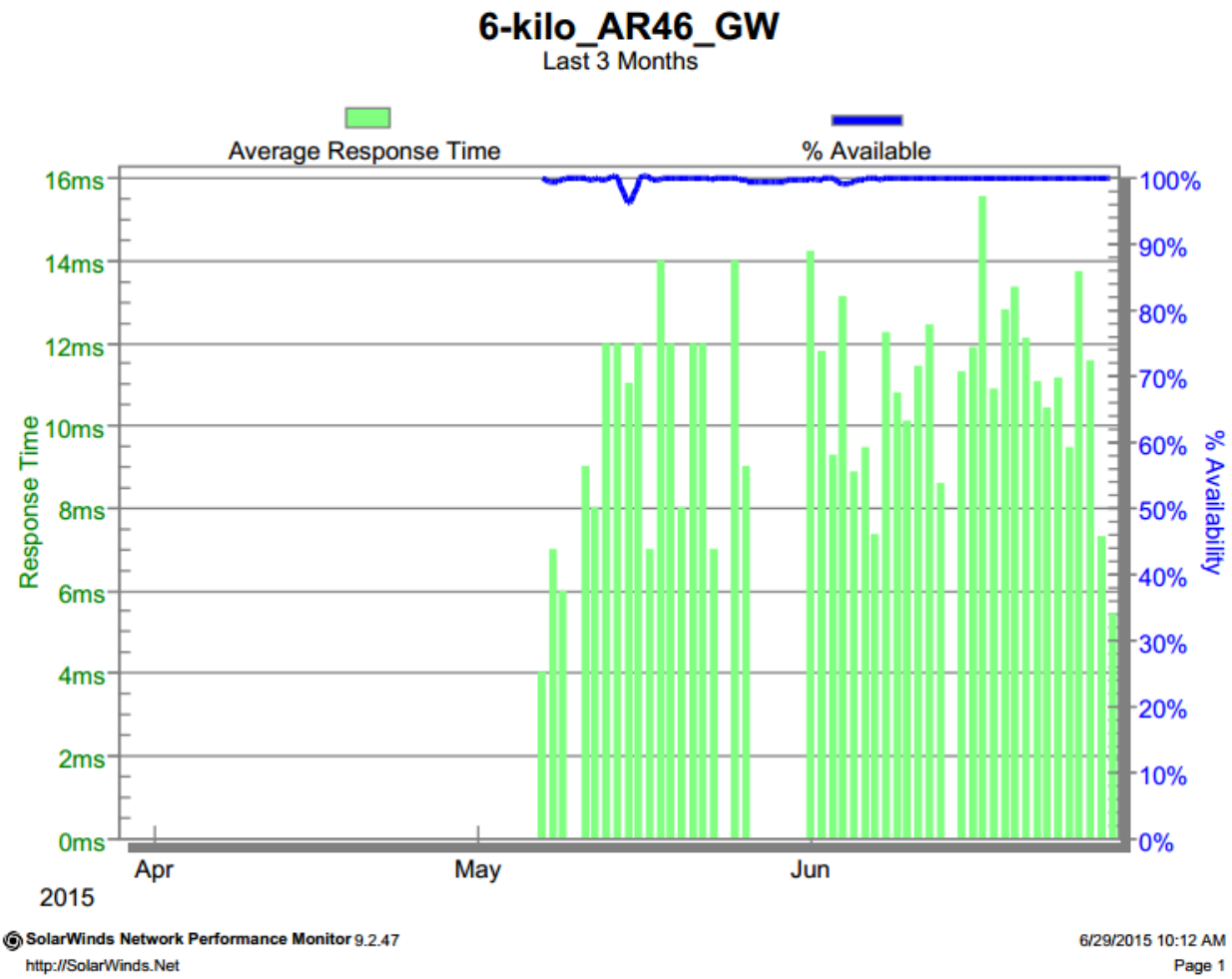


Figure 3.4: Average response time and availability of AAU Network main getaway router

The packet loss-rate metric at the link-level is measured using SNMP. It uses a counter to keep track of the total number of lost packets. End-to-end packet loss rate is hard to measure because Internet packet loss is very bursty. A reasonable estimation of path loss often requires a large number of sampling packets but the general concept of packet loss-rate metric is measuring the number of packet that didn't deliver properly by configuring the destination to send acknowledgment message to measurement node [73] [74].

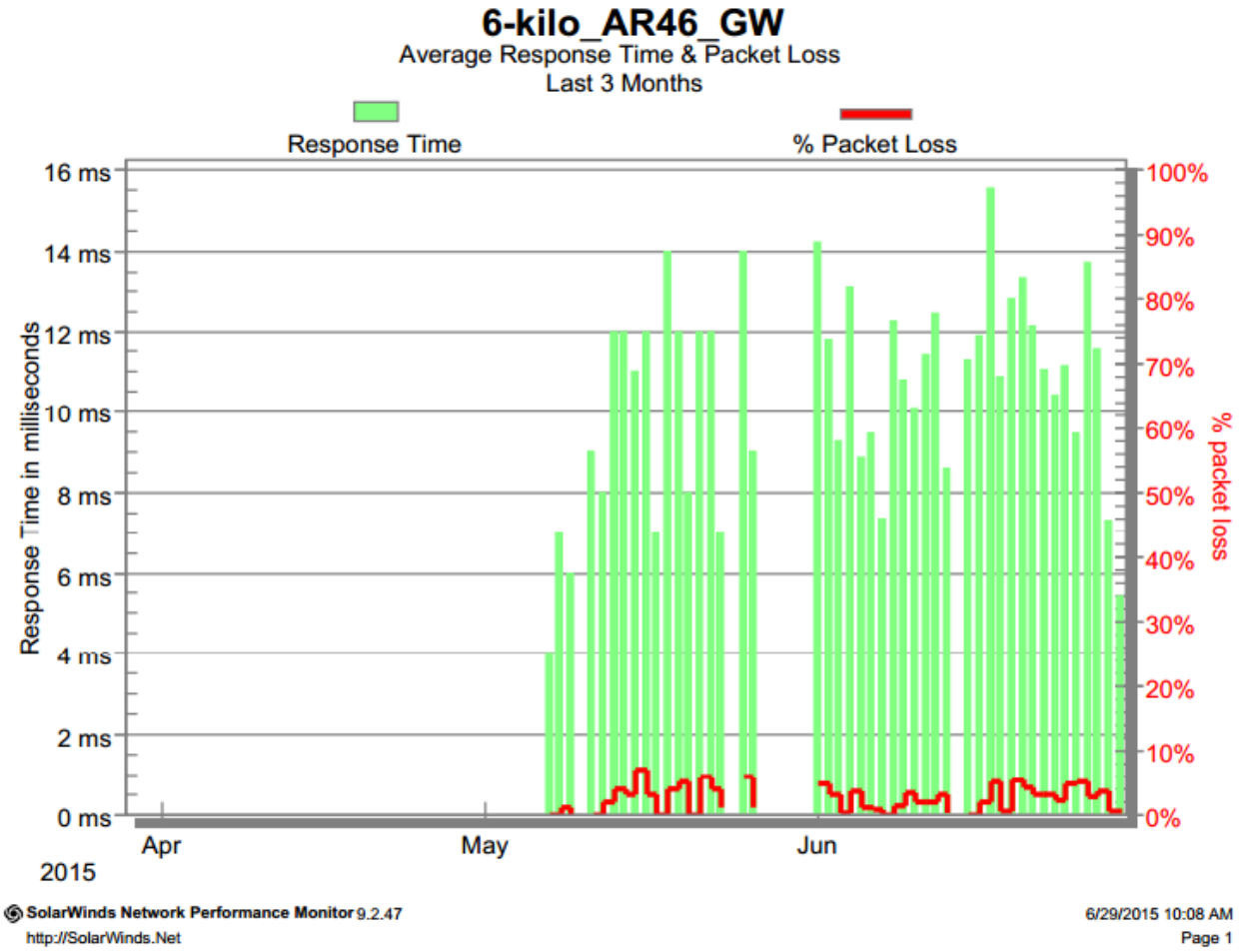


Figure 3.5: Average response time and Packet loss of AAU Network main getaway router

Available bandwidth at the link-level is measured using link capacity and traffic load information [75]. Internet link capacity is generally known a priori. The traffic load can be calculated using the statistics collected by SNMP. At the end-to-end level system administrators generally use tools which use TCP flows transmission performance in order to quantify path available bandwidth. Generally tools measure TCP achievable throughput not the available bandwidth. This is because TCP achievable throughput is not only determined by the available bandwidth of the path it is also affected by the level of multiplexing of background traffic flows and system configuration parameters such as TCP buffer sizes [76]. This study focuses on end-to-end available bandwidth measurement.

The network analysis is performed by using different software's and hardware's. The hardware include both the devices currently used by AAU network and outside the network such as router, multilayer switch, firewalls, servers and computers. All packet switching devices on the network analyzed to make the research accurate as much as possible.

Solar winds is used both for collecting and analysis of network traffic. Solar winds is a commercial real-time software solution which is used mainly for traffic surveillance, traffic shaping or as a firewall. Solar winds can also be used for traffic analysis and for gathering Statistics about the traffic.

Proxy server used to filter user access based on the ICT policy. The server currently installed at Sidist kilo data center and connected to the gateway router on AAU network. In this study proxy server used to collect log data and configuration of the current proxy server to deploy proxy server on simulation environment.

Critical application selected using two methods the first one is interviewing the system administrator and the second method is by referring most used applications by the users. Accordingly HTTP, Database, Email, Video and Voice application selected as a critical application consecutively.

To analyze application Statistics sflow/NetFlow data collected from the network by configuring the router to export sflow/NetFlow data to sflow/NetFlow collector device which is a computer sflow/NetFlow collector software where installed and static IP address configured. The sflow/NetFlow collector used both the collection and analysis of sflow/NetFlow data.

Mime types

1 - 10 of 301 Rows

	Mime type	↓ Hits		Page views	Visitors	Size	Sessions	Session duration
1	-	3,314,369	48.9 %	3,212,072	3,572	411.04 G	7,900	245d 02:05:40
2	text/html	1,496,503	22.1 %	1,466,649	3,327	5.89 G	5,991	44d 23:32:59
3	image/jpeg	338,063	5.0 %	42,102	2,195	7.10 G	1,544	19:03:47
4	image/gif	306,275	4.5 %	180,029	2,201	1.11 G	2,751	6d 09:37:03
5	text/javascript	180,340	2.7 %	90,158	2,134	2.04 G	2,514	2d 22:37:18
6	image/png	169,941	2.5 %	23,247	2,238	2.07 G	1,566	10:22:31
7	application/octet-stream	149,431	2.2 %	147,901	2,333	42.03 G	3,311	19d 01:42:43
8	application/javascript	142,330	2.1 %	64,889	1,986	1.50 G	2,027	2d 00:27:31
9	application/x-javascript	135,637	2.0 %	37,335	1,950	1.55 G	1,402	13:47:37
10	application/json	90,664	1.3 %	90,578	2,236	181.71 M	3,392	16d 23:52:47
	291 other items	460,825	6.8 %	376,753	–	104.68 G	–	38d 19:20:07
	Total	6,784,378	100.0 %	5,731,713	–	579.17 G	–	1y 13d 02:30:03

Table 3.1: AAU network application usage

Table 4.1 list top application based on Hits, page views, visitors, size, session and session duration. From the table unknown application holds 48.9% share, text/html, image/gif, text/javascript and image/png takes 22.1%, 5%, 4.5%, 2.7% and 2.5% consecutively and finally application data takes 7.6% share. We can generalize that users use mostly HTTP application. The session cannot be correlated with size because each session may carry an arbitrary number of packets which have different size.

After collecting the necessary data AAU network simulated using OPNET. This study uses simulation because of two reasons. The first one is AAU network is very sensitive network so it is very difficult to attempt any changes on the real environment of the network. The second reason is this study is low budget research so to do experimentation in a low-cost simulation is ideal. Generally based on the reasons we have to analyze the network using low-cost and low risk environment. To do that simulating the network is number one choice for this study.

The simulation tool used for simulating the network is OPNET modeler. In comparison with other simulations tools like OMNET and NS2, OPNET modeler is considered as the best simulation tool as it provides several advantages that others cannot provide. The significant advantages of OPNET modeler are it supports level of models and provides a user interface to establish several networks. Not only OPNET modeler NS2 is also considered as a best simulation tool but the single disadvantage in using NS2 is its behavior which is very complex as it includes loads of coding part and this problem is completely avoided using the OPNET modeler. The significant factor that encourages the usage of OPNET modeler on this study is its suitability for simulation of AAU network because of its support several model families over both on LAN and wireless network environment to enable the communication and its drag drop approach. Using this approach the simulation procedure is done simply by selecting the required objects. OPNET also advantages on simulating very large enterprise networks.

For the simulation purpose configuration file taken from routers, multilayer switches, wireless devises and DHCP server by using "*show commands*" and "*configuration export*" command. For security reason sensitive data for the institution not gathered. For example passwords, firewall configurations, static IP address of central devises, user IP address and other configuration file which have a security risk.

This study also uses AppDoctor for analysis, diagnosis and experimenting the result. AppDoctor is a part of OPNET Modeler tool used to determine application bottlenecks and diagnose the bottleneck by using different technique based on the network. AppDoctor also have another feature which used to predict what the network will look like if some modification done on the network. Generally we can say most of the analysis done on OPNET modeler.

4.2.2 Metrics

As discussed in section 1.6.9 the metrics used to evaluate the performance are Traffic Received and Sent (Packets/Sec) used for both data and multimedia application. Response Time (Sec), TCP Delay (Sec), TCP Retransmission Count for Data applications (Database, Email, HTTP), Packet Delay Variation, Packet End-to-End Delay (Sec), Traffic Received and Sent (Packets/Sec), Jitter (Sec), MOS Value for Multimedia Applications (Video and Voice).

4.2.3 Environmental setup

4.2.3.1 Simulation tools

The simulation phase of this study is entirely done by the network simulation software known as OPNET. OPNET modeler has been chosen for simulation environment because of the following features [9].

- Has the great ability to access with wide range of available standard and vendor specific communication networks that help in greatly reducing the time involved in developing simulation environments from scratch, allowing modelers to directly include developed models in their simulations.
- Provides a rich variety of development environment that support the modeling of communication networks and distributed systems.
- Provide huge number of documentation for the user to develop the network models.
- Offer flexible and easy graphical interface for viewing the results.
- Provides more feature than any other simulator in this market which attracts by the network operators.
- Results from OPNET are easily interpreted with comprehensive tools to display, plot and analyze time series, histograms, probability functions, parametric curves, and confidence intervals, which can be exported to a spreadsheet [78].

It can be run on both windows and Linux platform. This study uses the windows version running on Windows 8. The installation file of OPNET Modeler is available on the OPNET software module or on official website of OPNET. There are three installation files to be downloaded. OPNET Modeler installation file (modeler_145A_PL8_7808_win.exe), Model library installation file (modeler_docs_02-Sep-2008_win.exe) and documentation installation file (models_145A_PL8_24Sep08_win.exe).The OPNET Modeler also requires a compiler to compile the simulation process. The compiler used for this study is windows platform is Visual Studio 2012.

Before installing the OPNET Modeler, the compiler should be configured first, to create relationship between the OPNET Modeler and Visual Studio Compiler by setting environmental variables.

4.2.3.2 Network Components

This section discusses about the network components and models used on the simulation environment to represent the actual network [11] [9].

- Ethernet2_slip8_ler(Label Edge Router) and ethernet2_slip8_lsr(Label Switched Router) node models are used to represent an IP-based gateway running MPLS and supporting up to two Ethernet interfaces and up to 8 serial line interfaces at a selectable data rate. IP packets arriving on any interface are routed to the appropriate output interface based on their destination IP address.
- Ethernet16_switch node model is used to represent a switch supporting up to 16 Ethernet interfaces. The switch implements the Spanning Tree algorithm in order to ensure a loop free network topology. Switches communicate with each other by sending Bridge Protocol Data Units (BPDU's). Packets are received and processed by the switch based on the current configuration of the spanning tree.
- Ethernet_wkstn_adv node model is used to represent a workstation with client-server applications running over TCP/IP and UDP/IP.
- 10BaseTand 100BaseTduplex links are used to represent the Ethernet connections operating at 10 Mbps and 100Mbps, respectively. These links can connect any combination of the nodes such as Station, Hub, Bridge, Switch and LAN nodes (except Hub-to-Hub, which cannot be connected).
- ppp_adv, point-to-point link is used to connect two nodes with serial interfaces (e.g., routers with PPP ports)) at a selectable data rate.
- Application_Config includes a name and a description table that specifies various parameters for the different applications (i.e. video conferencing and voice applications). The specified application name is used while creating user profiles on "Profile_Config" object.
- Profile_Config is used to create user profiles. These user profiles can then be specified on different nodes in the network to generate application layer traffic. The applications defined in the Application_Config are used by this object to configure profiles. Traffic patterns can be specified followed by the configured profiles and the applications.
- QoS_Config is used to define the QoS configuration details for protocols supported at the IP layer. These specifications are referenced by the individual nodes using symbolic names as FIFO, CBWFQ and Priority Queuing.
- MPLS_Config is used to configure Forwarding Equivalence Class (FEC) and Traffic Trunk specifications. These specifications are associated with different flows at the ingress LERs, to differentiate the flows into various classes and different QoS agreements.

- MPLS_E-LSP_Static is a model of static Label Switched Path (LSP). This is used to create the static forwarding tables at each node whereby the LSP can traverse.

In OPNET Modeler devises will not be functional without configuring application. Various nodes which are defined in the network for the various applications these are Application configuration or "Application Config" node is used to specify different tier names used in the network model. The tier name and the corresponding ports at which the tier listens to incoming traffic is cross-referenced by different nodes in the network. Also it specifies applications using available application types.

Application configuration is totally dependent on profile configuration. Profile configuration node is used to create user profiles. These user profiles can then be specified on different nodes in the network to generate application layer traffic. The application defined in the "*application config*" object is used by this object to configure profiles. This specify the traffic patterns followed by the applications as well as the configured profiles on this object.

After configuring application and profile configuration we have to set tasks using task configuration. Task configuration node is used to define/create tasks used to characterize custom applications. These applications are then used to create profiles, which are applied across different nodes to generate desired traffic. Finally IP attribute defines attribute configuration details for protocols supported at the IP layer.

4.2.3.3 Simulation Scenarios

In this study the network simulated according to the current AAU network topology but to analyze the network in great detail more than thirty three scenarios created. First all applications such as HTTP, Email, Database, Video and Voice applications simulated individually then for each application more than ten effects applied. Additional scenarios created for each application and effects for example when the background traffic consumes 0%, 30%, 60% and 90% of available bandwidth, when delay varies from 0s up to 444 micro second and when the application share the network with other application. This study uses individual scenarios for each application and effects because of to get comparison results of scenario and to analyze each effect on application individually. But all scenarios use the following network architecture.

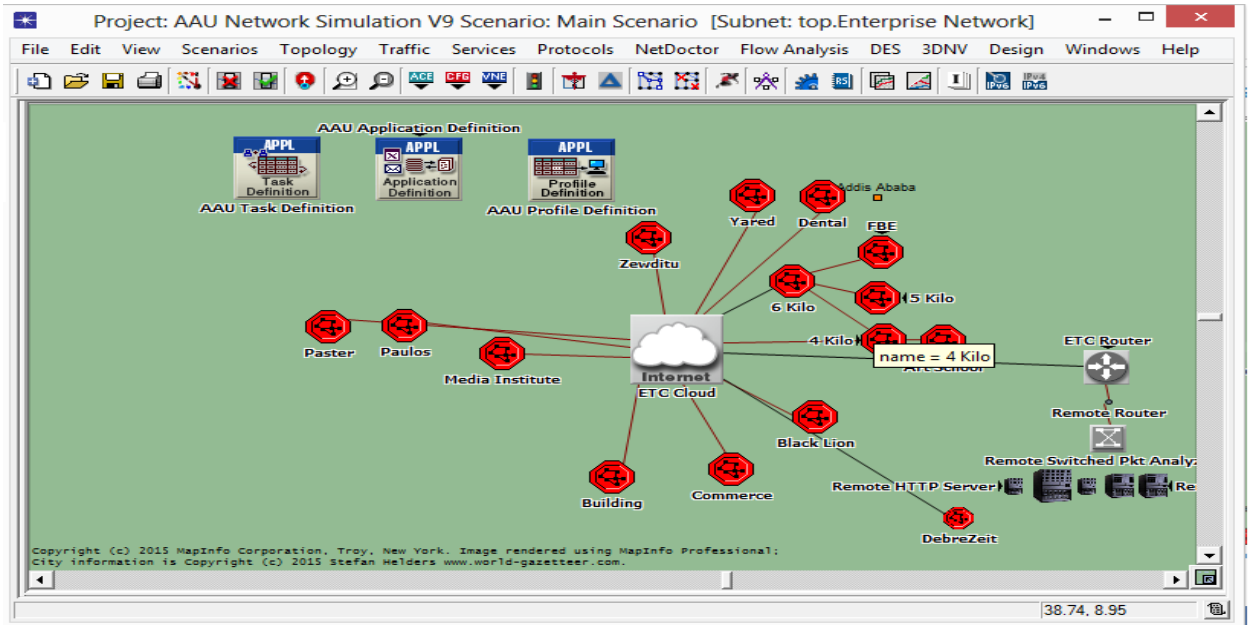


Figure 3.6: AAU network architecture on simulation environment

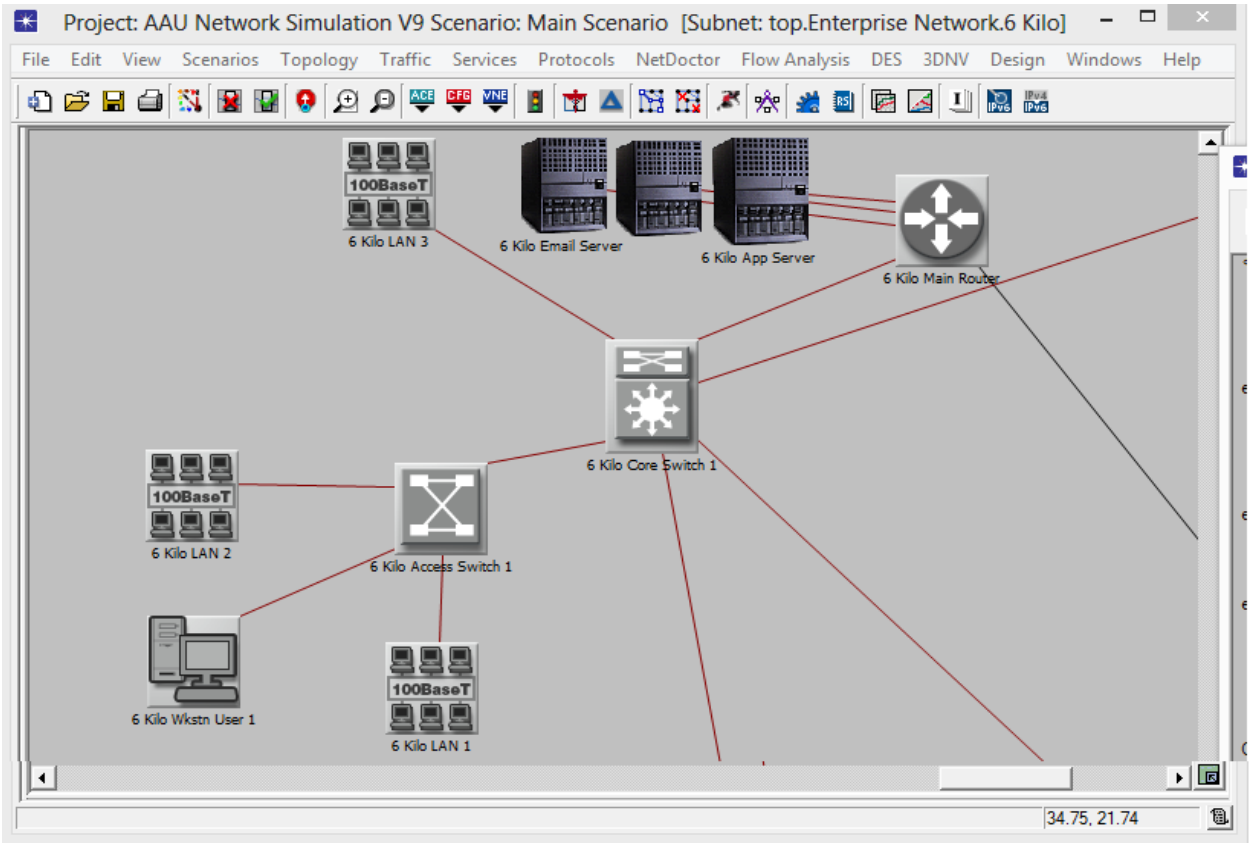


Figure 3.7: 6 Kilo Network architecture on the simulation environment

AAU network architecture on the simulation have 16 subnets for each branches of network. Subnets assigned based on ICT documentation and all the network equipment exactly matched from the real environment. Figure 3.7 shows the network equipment's used on the simulation 100BaseT LAN represents the local area network which encompasses all the detail of the LAN environment. Access switch represent a switch which directly connected to the user. Core switch and router represents layer three switch which capable of routing packets and the server represent proxy, DNS, DHCP, web and application server.

CHAPTER FOUR

RESULT AND DISCUSSION

5.1 Results

Critical applications HTTP, Email, Database, Video and Voice simulated individually. The behavior of critical applications tested by applying effects like when there is background traffic, when sharing the link with other application and when the delay varies then the result presents on this section.

5.1.1 Scenario 1: HTTP

Simulation results of HTTP application when sharing the link with other applications, when there is background traffic and when delay varies. TCP delay, TCP Retransmission Count, TCP Segment Delay, HTTP Page Response Time, HTTP Object Page Response Time, HTTP Traffic Received and HTTP Traffic Sent used as metrics.

5.1.1.1 HTTP When there is Background Traffic

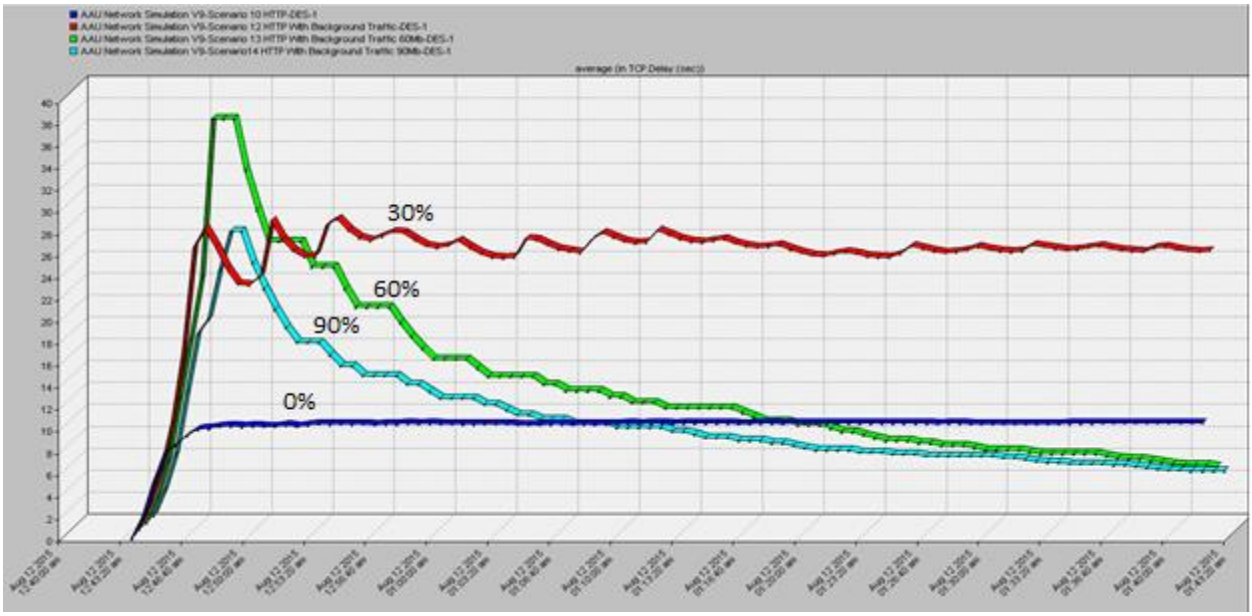


Figure 4.1: Average Comparison result of TCP Delay (sec) for HTTP application When Background traffic takes 0%, 30%, 60% and 90% of the total bandwidth respectively.

TCP	0% Up To 90% Of The Bandwidth Taken By Background Traffic			
	0%	30%	60%	90%
Statistic	**Average**	**Average**	**Average**	**Average**
TCP Delay (sec)	10.819	25.893	5.56	4.496
TCP Retransmission Count	600.3	666.1	553.4	527.0
TCP Segment Delay (sec)	7.865	9.953	69.169	92.20

Table 4.1: TCP result of HTTP applications when there is 0%, 30%, 60% and 90% background traffic usage of the bandwidth of the links

According to Figure 4.2 and Table 4.1, the amount of background traffic have an impact on HTTP applications.

Table 4.1 shows that the delay decreases while the background traffic increases. This is because the HTTP application or web browser uses TCP protocol, one of the distinguishing characteristics of TCP is that it checks the deliverability of every single packet by using acknowledgment message. This have two disadvantages the first one is the acknowledgment message consumes a bandwidth the second one is it stops sending a packet unless delivery message comes it is also acknowledgment message. Therefor HTTP application highly affected when the background traffic takes most of the bandwidth.

In Table 4.3 TCP Delay shows 25.893 second maximum when 30% and 4.496 second minimum when 90% of the bandwidth consumed by background traffic. TCP Delay stop rising after 30% but TCP Segment Delay keep on rising from 7.865 second at 0% and 92.20 second at 90%. This shows that HTTP application stops sending segments because the segments sent before didn't sent delivery acknowledgment and the application become busy checking whether the segment delivered or not. Doing this takes more time when there is small amount of bandwidth available to use.

Generally speaking when we compare http applications based on TCP delay (sec), TCP retransmission counts (sec) and TCP segment Delay (sec) the background traffic load have a considerable impact on HTTP application.

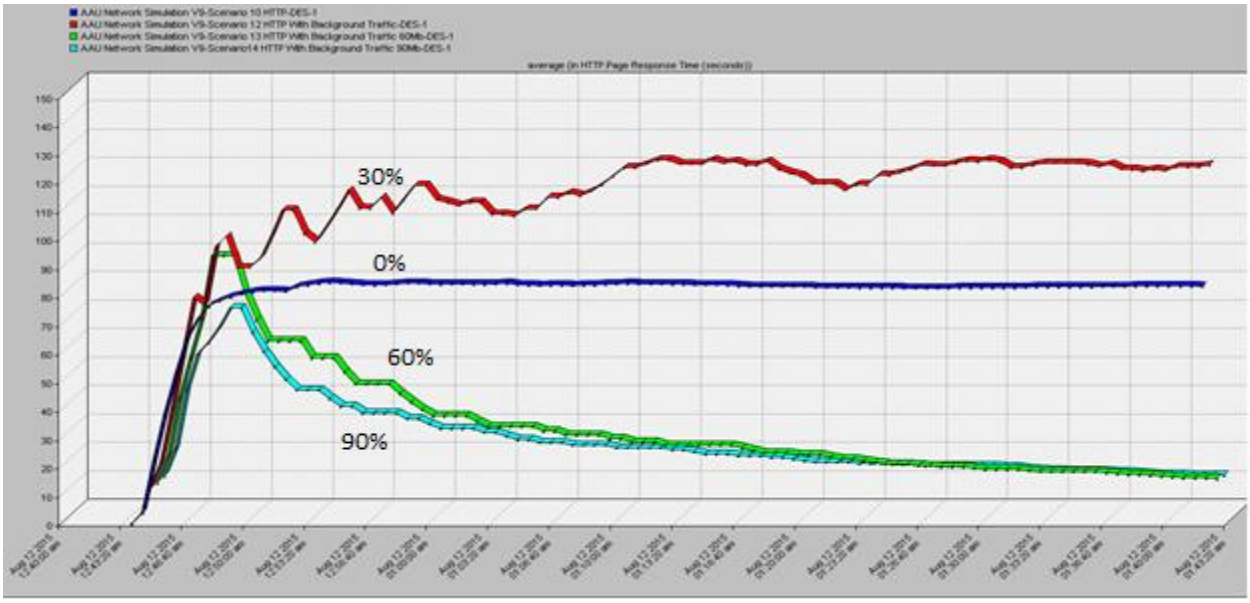


Figure 4.2: Comparison result of HTTP Page Response Time (Sec) when the background traffic consumes 0%, 30%, 60% and 90% of the bandwidth.

Page Response Time and TCP Delay shows considerably similar result according to Figure 4.2 and Table 4.2 but Page Response Time keep on rising at 30%.

HTTP				
	0%	30%	60%	90%
Statistic	**Average**	**Average**	**Average**	**Average**
HTTP Object Response Time (seconds)	27.951	55.97	7.98	7.756
HTTP Page Response Time (seconds)	84.23	124.44	11.77	10.57
HTTP Traffic Received (bytes/sec)	9,965	2,333.0	325.0	343.7
HTTP Traffic Received (packets/sec)	15.222	4.048	0.499	0.519
HTTP Traffic Sent (bytes/sec)	14,326	4,196	477	494
HTTP Traffic Sent (packets/sec)	18.763	5.622	0.626	0.645

Table 4.2: HTTP Statistics of HTTP application when 0 up to 90% of the bandwidth consumed by background traffic.

The maximum response time scored when the background traffic takes 30% of the total bandwidth. HTTP Response time rises when the background traffic consumes more and more bandwidth but it drops when it riches 60 % and stop showing variation when the background traffic load increases this is because devices stop sending and receiving packets while the background traffic increases. We can understand from Table 4.2 HTTP Traffic Received Packets is 15.222 at 0% and 0.499 at 60%. HTTP Traffic Sent is 18.763 at 0% and 0.626 at 60%. As the background traffic increases and consumes more bandwidth the HTTP application decreases the number of packet to be sent

and receive per second. The background traffic affect both the bandwidth and the process capacity of mediatory devices.

5.1.1.2 HTTP When Sharing WAN Link

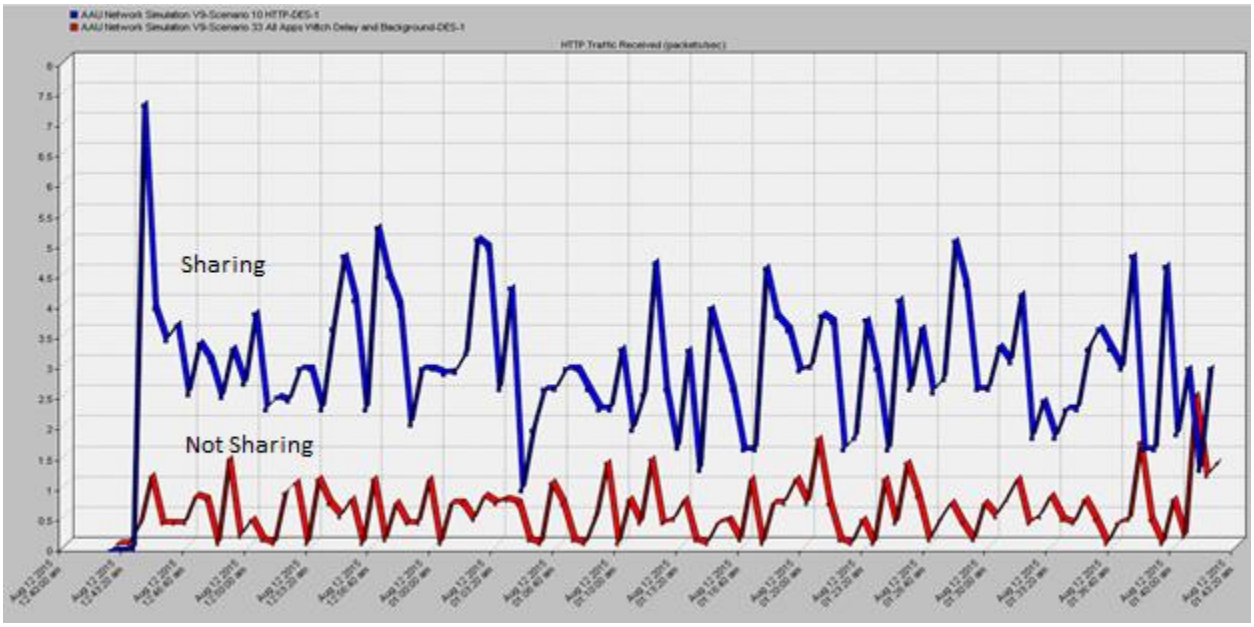


Figure 4.3: Comparison result of Average Page Response Time (In Second) for HTTP application when sharing network link.

HTTP	WAN Link	
	Not Shared	Shared
Statistic	**Average**	**Average**
HTTP Object Response Time (seconds)	0.4908	0.00764
HTTP Page Response Time (seconds)	1.0645	0.0186
HTTP Traffic Received (bytes/sec)	2,327.7	411.5
HTTP Traffic Received (packets/sec)	3.0144	0.5367
HTTP Traffic Sent (bytes/sec)	2,341.7	411.5
HTTP Traffic Sent (packets/sec)	3.0250	0.5367

Table 4.3: HTTP Statistics of HTTP application when HTTP application use the WAN link alone and when sharing the WAN link critical applications

The average response time decreased significantly from 0.4908 to 0.00764 second when the application share the WAN link. The number of packet sent and received per second pointedly decreased from 3.0144 to 0.5367 (Packet/Sec) and from 3.0250 to 0.5367 (Packet/Sec). So we can understand that sharing the WAN link with other critical application affects HTTP application.

5.1.1.3 Http When Delay Changes

HTTP			
	Distance Based	Delay 0.222 Sec	0.444 Sec
Statistic	**Average**	**Average**	**Average**
HTTP Object Response Time (seconds)	27.951	1.9520	2.1171
HTTP Page Response Time (seconds)	84.23	5.514	6.2450
HTTP Traffic Received (bytes/sec)	9,965	6,822	347.4
HTTP Traffic Received (packets/sec)	15.222	9.057	1.084
HTTP Traffic Sent (bytes/sec)	14,326	7,017	348.2
HTTP Traffic Sent (packets/sec)	18.763	9.219	1.086

Table 4.4: Average HTTP Statistics of HTTP application under heavy browsing when the link delay changed

Even if the delay changes from 0.222 to 0.444 second on the AAU simulated environment but HTTP Page Response Time changed from 5.514 to 6.2450 almost no change.

On Table 4.4 when the delay is 0.222 and 0.444 second the number of packet received decreased from 9.057 to 1.084 and the number of packet sent decreased from 9.219 to 1.086. This is caused by two reasons the first reason is HTTP application takes much time to communicate with HTTP application on the other end since the delay rises. So the number of packet to be sent or receive on constant amount of time is decrease. The second reason is HTTP application uses TCP protocol and TCP checks the deliverability of packet by acknowledgment number. If the acknowledgment takes more time than expected the application considers the packet is lost on the way so it will retransmit the packet.

Delay change affects HTTP Response Time in small scale but HTTP Traffic Received sent affected more. So Delay change affects HTTP application on two ways. The first one is it affect the quality of service. This is because when the delay increases HTTP packet takes considerably high amount of time to reach to the other end. The second one is it make HTTP servers and browsers stop sending and receiving.

5.1.2 Scenario 2: Email

Simulation results of Email application when sharing link with other applications, when there is background traffic and when delay varies. The Metrics used for evaluation are TCP delay, TCP Retransmission Count, TCP Segment Delay, Email Page Response Time, Email Object Page Response Time, Email Traffic Received and Email Traffic Sent.

5.1.2.1 Email when there is Background Traffic

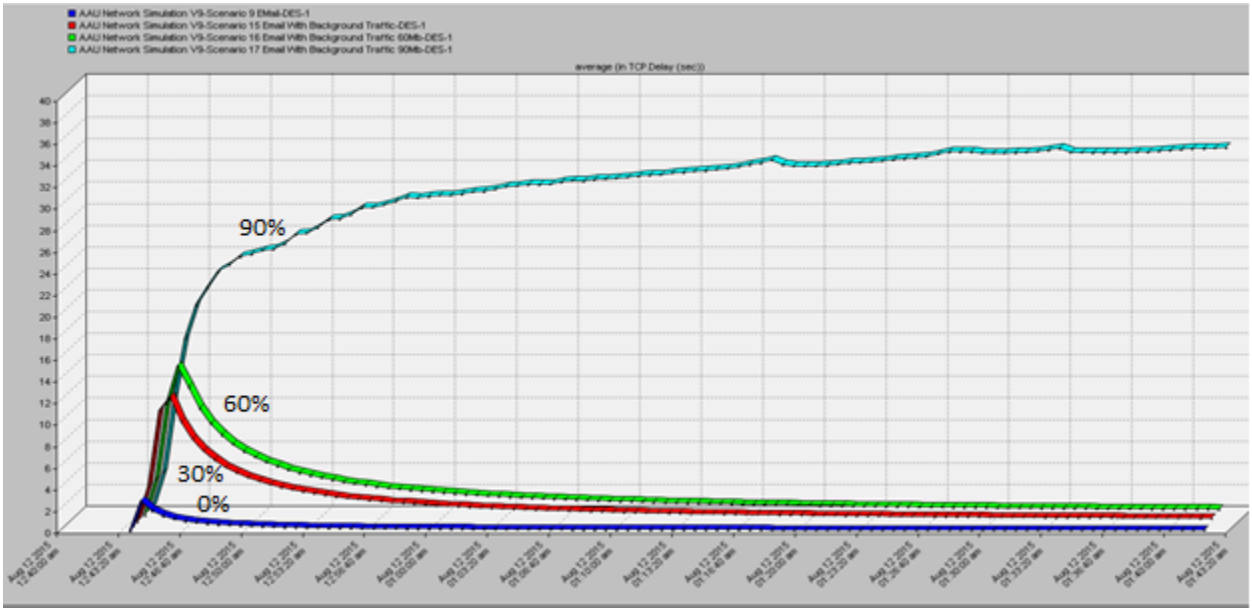


Figure 4.4: Average comparison result of Email application when the background traffic uses 0% to 90% of the bandwidth.

TCP				
	0%	30%	60%	90%
Statistic	**Average**	**Average**	**Average**	**Average**
TCP Delay (sec)	0.2808	0.748	0.955	33.801
TCP Retransmission Count	35.7	722.3	553.8	162.1
TCP Segment Delay (sec)	0.0938	0.366	0.464	19.692

Table 4.5: Email TCP comparison result when the background traffic consumes 0%, 30%, 60% and 90% of the bandwidth simultaneously.

Figure 4.4 shows that when the background traffic rises TCP Delay also rises on. On Figure 4.5 TCP Delay is smooth for 0%, 30% and 60% but for 90% it dramatically rises up to 33.801 second. In addition on Table 4.5 all the metrics on Email application affected on similar manner. For instance Average TCP Delay (sec) rises from 0.2808 second to 33.801 and Segment Delay (Sec) also rises from 0.0938 to 19.692 but Retransmission Count rises on large scale from 35.7 to 722.3.

Background traffic on Email application have direct impact. When the background traffic consumes more than 85% of the total bandwidth it makes the Email application almost to stop accessing the network.

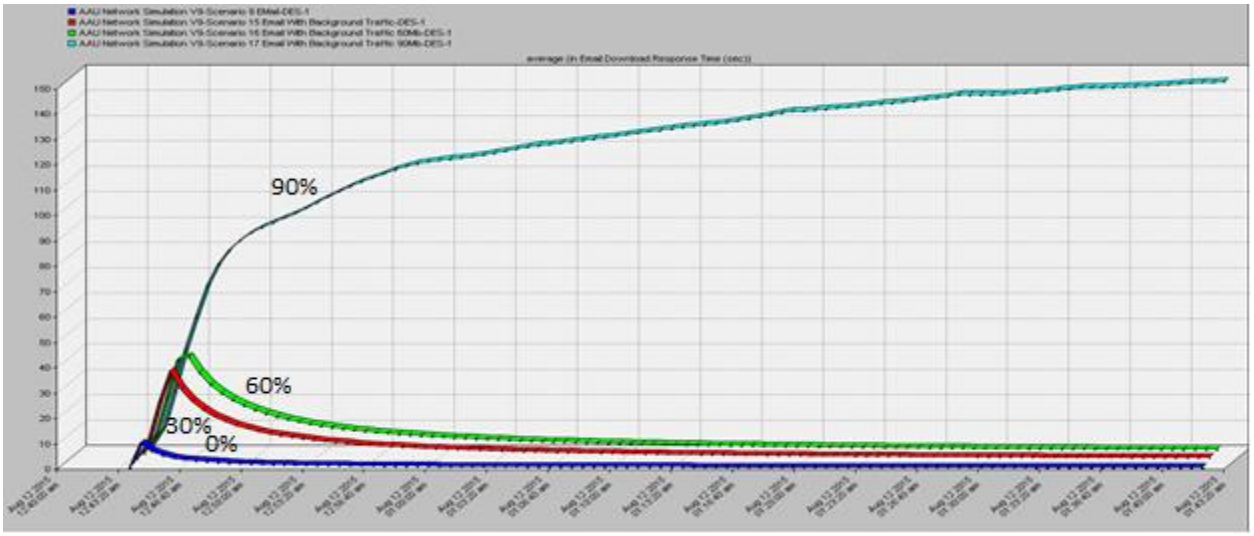


Figure 4.5: Comparison result of average Email Download Response time (Sec) when the background traffic consumes 0%, 30%, 60% and 90% of the bandwidth.

Email				
	0%	30%	60%	90%
Statistic	**Average**	**Average**	**Average**	**Average**
Email Download Response Time (sec)	0.707	2.118	3.013	145.76
Email Traffic Received (bytes/sec)	849	1,511	1,555	1,335
Email Traffic Received (packets/sec)	1.126	2.003	2.062	1.772
Email Traffic Sent (bytes/sec)	849	1,511	1,555	1,388
Email Traffic Sent (packets/sec)	1.126	2.004	2.062	1.838
Email Upload Response Time (sec)	0.728	2.076	2.993	145.80

Table 4.6: comparison result of Email application when the bandwidth taken from 0% up to 90% by background traffic.

Table 4.6 help us to understand the effect level of background traffic on Email application. Email Response Time increases from 0.707 to 145.76 second. Email Traffic Received (packets/sec) and Email traffic Sent (packets/sec) rises almost on equal rate. From 1.126 to 1.772 when the background traffic rises from 0% to 90%.

5.1.2.2 **Email When Sharing WAN Link**

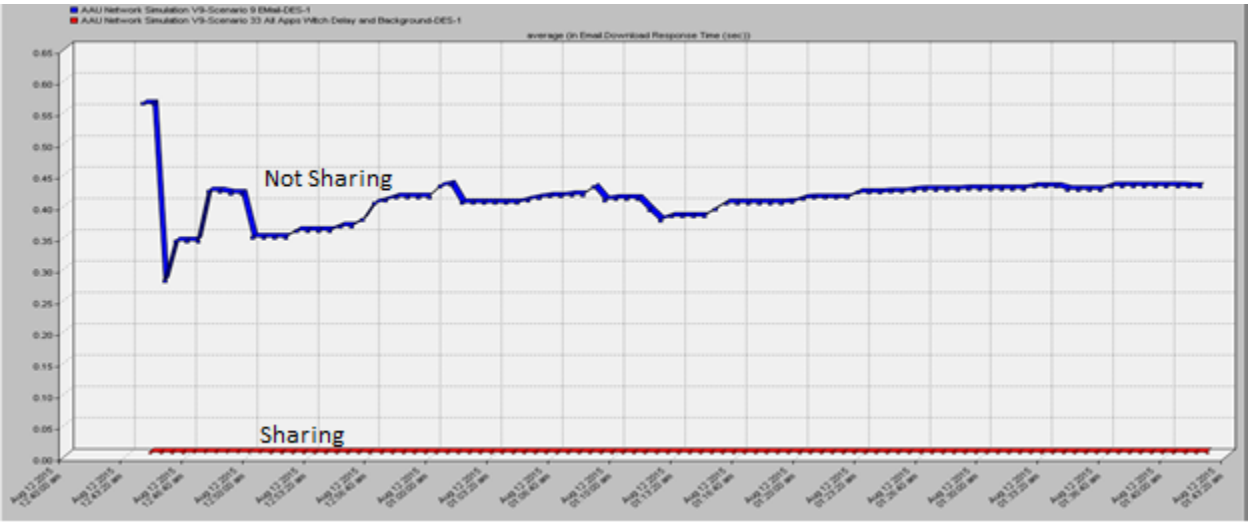


Figure 4.6: Average Email Download Response Time (Sec) result of Email application when it use the WAN link alone and when sharing the WAN link.

Email	WAN Link	
	Not Shared	Shared
Statistic	**Average**	**Average**
Email Download Response Time (sec)	0.43772	0.0040428
Email Traffic Received (bytes/sec)	84.6	47.8
Email Traffic Received (packets/sec)	0.1122	0.0633
Email Traffic Sent (bytes/sec)	84.6	47.8
Email Traffic Sent (packets/sec)	0.1122	0.0633
Email Upload Response Time (sec)	0.45448	0.03324

Table 4.7: Comparison result of Email application when sharing and not sharing the WAN link with other application.

Sharing the WAN link have more or less similar effect on Data applications. Table 4.7 shows that the response time decreased from 0.43772 to 0.0040423 seconds and the number of packets received per second also decreased from 0.1122 to 0.0633 packets/sec. The number of packets sent per second changed from 0.45448 to 0.03324 packets/sec.

The result on Figure 4.6 and Table 4.7 specifies that sharing WAN links with other application affects Email application on both quantity and quality of the service given by email application. Even if the response time improved the number of packet both sent and received per second decreased this have an impact on large network like AAU. Because AAU network send and receive gigabyte of traffic in a single second. If the capacity of sending and receiving traffic less than the

demand then the network will be congested and it need to be considered when we optimize the network. Previously on section 5.1.1.2 stated that sharing the WAN link have more or less similar impact on data application. When we compare Table 4.3 and Table 4.7 HTTP application and Email application affected more or less on similar scale when WAN link shared.

5.1.2.3 **Email When Delay Change**

Email	Delay		
	Distance Based	0.222 Sec	0.444 Sec
Statistic	**Average**	**Average**	**Average**
Email Download Response Time (sec)	0.707	4.722	8.606
Email Traffic Received (bytes/sec)	849	844	773
Email Traffic Received (packets/sec)	1.126	1.119	1.026
Email Traffic Sent (bytes/sec)	849	844	774
Email Traffic Sent (packets/sec)	1.126	1.119	1.027
Email Upload Response Time (sec)	0.728	4.741	8.380

Table 4.8: Email Download Response Time (Sec) result obtained by changing the delay 0, 222 and 444 second

When Delay changes from 0.222 to 0.444 second it have almost no impact on Email Traffic Received and Sent. Traffic Sent (Packet/Sec) declined from 1.119 to 1.027 and Traffic Received (Packet/Sec) also declined from 1.119 to 1.026. The Delay change make the Email Download and Upload Response Time (Sec) to be double.

When the Delay rises Email application only losses its quality and speed but the quantity is the same.

5.1.3 Scenario 4: Database

Simulation results of Database application when sharing link with other applications, when there is background traffic and when delay varies. The Metrics are TCP delay, TCP Retransmission Count, TCP Segment Delay, DB Query Page Response Time, DB Query Object Page Response Time, DB Query Traffic Received and DB Query Traffic Sent.

5.1.3.1 Database when there is Background Traffic

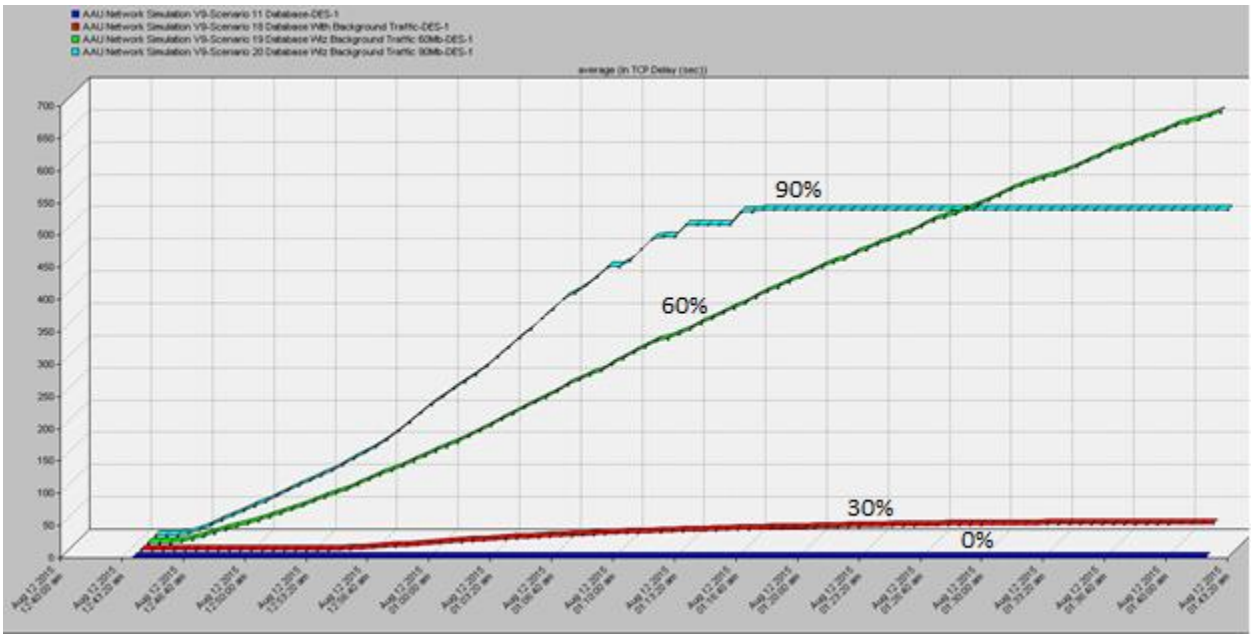


Figure 4.7: Average comparison result of Database application on the measurement of Database TCP Delay (Sec) when the background traffic consumes 0%, 30%, 60% and 90% of the bandwidth.

TCP

	0%	30%	60%	90%
Statistic	**Average**	**Average**	**Average**	**Average**
TCP Delay (sec)	0.12053	41.218	668.5	505.8
TCP Retransmission Count	1	60.45	190.57	108.89
TCP Segment Delay (sec)	0.10076	22.101	93.60	199.95

Table 4.9: Average TCP statistics of Database application when the background traffic uses from 0% up to 90% of the bandwidth.

Table 4.9 shows that TCP Delay changes from 0.12053 to 505.8 when the background traffic consumes 0% up to 90% of the bandwidth. This clearly show that database application have direct relationship with background traffic. When the background traffic increases the quality of database application decreases as well.

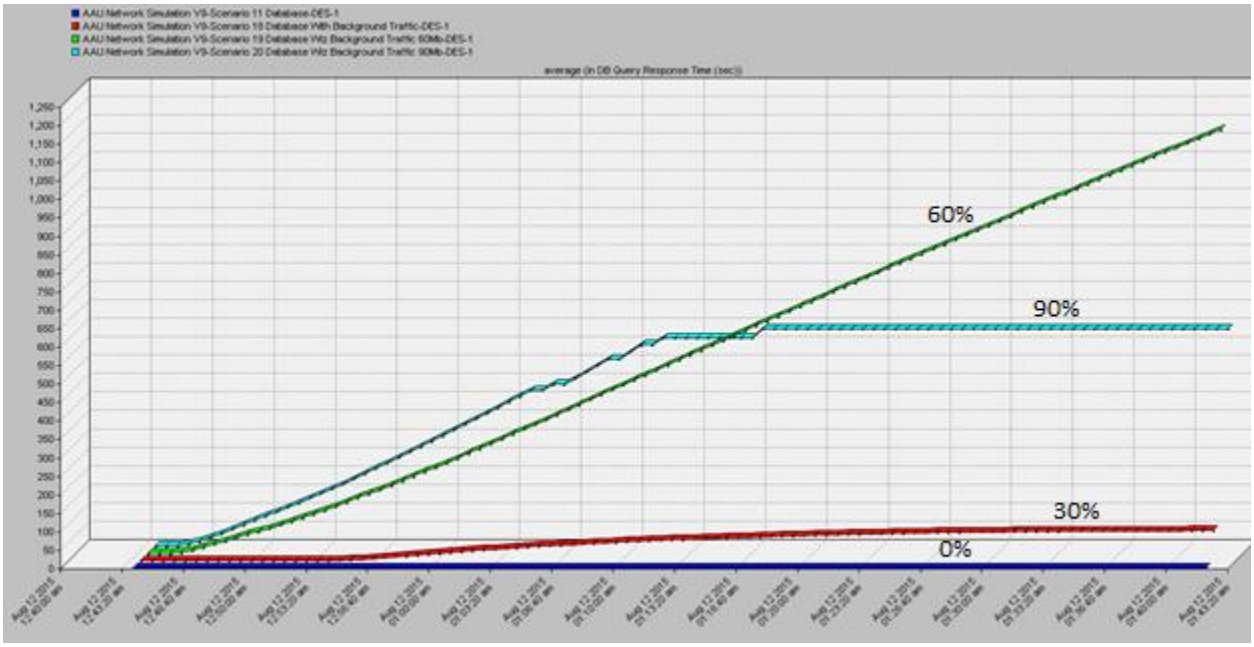


Figure 4.8: Average Database Query Response Time (Sec) when background traffic uses 0%, 30%, 60% and 90% of the total bandwidth.

DB Query

	0%	30%	60%	90%
Statistic	**Average**	**Average**	**Average**	**Average**
DB Query Response Time (sec)	0.24155	82.29	1,149.0	589.2
DB Query Traffic Received (bytes/sec)	13,252	12,987	3,661	1,005
DB Query Traffic Received (packets/sec)	25.883	25.365	7.151	1.962
DB Query Traffic Sent (bytes/sec)	13,252	13,147	7,383	2,806
DB Query Traffic Sent (packets/sec)	25.883	25.677	14.419	5.480

Table 4.10: Comparison result of Average Database Query statistics when 0%, 30%, 60% and 90% of the WAN link bandwidth consumed by background traffic.

Figure 4.8 and Table 4.10 shows the result of database application gathered when the bandwidth consumed by background traffic. DB Query Response Time (Sec) rises from 0.24153 to 589.2 second this means the application taking more time to get into the destination. This can affect both the quality and performance of the application. On the other hand the application keep on decreasing the number of packet sent and receive per second. We can comprehend this by observing Table 4.10 DB Query Traffic Received (packets/sec) is 25.883, 25.365, 4.151 and 1.962 packet per second when the background traffic consumes 0%, 30%, 60% and 90% of the bandwidth respectively.

5.1.3.2 When Sharing WAN Link

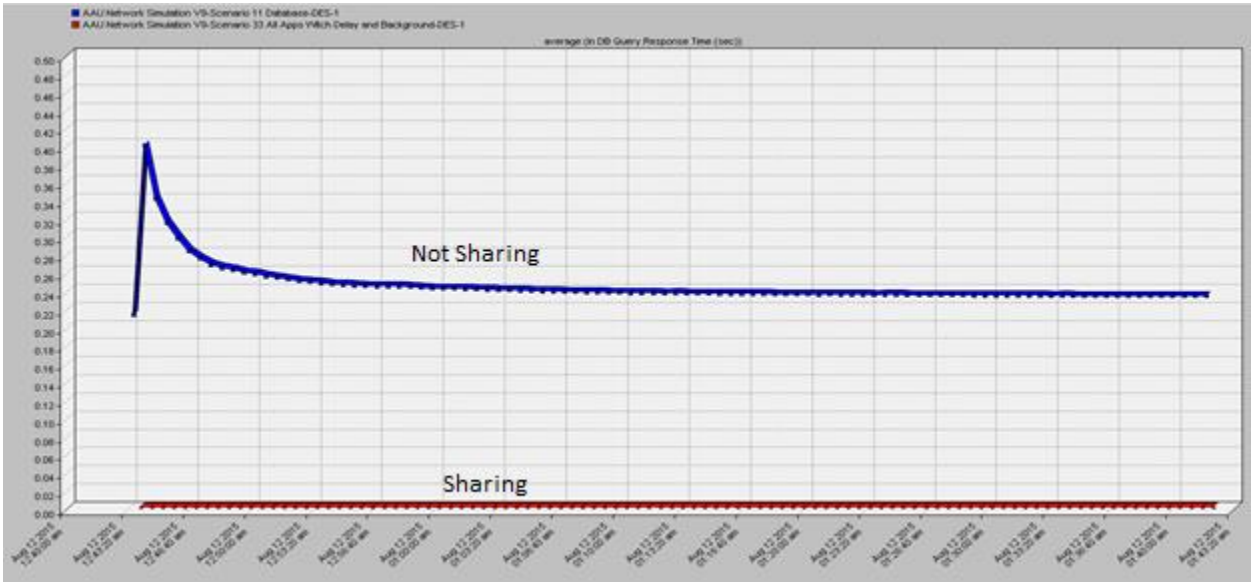


Figure 4.9: Average Database Query Time (Sec) of Database application when database application use the WAN link alone and when sharing the WAN link with other applications.

DB Query

	WAN Link	
	Not Shared	Shared
Statistic	**Average**	**Average**
DB Query Response Time (sec)	0.19922	0.0009555
DB Query Traffic Received (bytes/sec)	1,203.9	573.44
DB Query Traffic Received (packets/sec)	2.3514	1.1200
DB Query Traffic Sent (bytes/sec)	1,204.1	576.71
DB Query Traffic Sent (packets/sec)	2.3517	1.1264

Table 4.11: Database Query Statistics for Database Applications when they use the WAN link alone and with other applications.

Table 4.11 illustrate that the number of packet sent and received per second decreased by half. DB Query Traffic Received (Packets/Sec) reduced from 2.3514 to 1.1200 packets per second and DB Query Traffic Sent (Packets/Sec) also decreased from 2.3517 to 1.1264. The results on Figure 4.9 and Table 4.11 generally lead us to know the fact that when the network link shared by other applications Database application reduce it's sending and receiving capacity by half.

5.1.3.3 Database When Delay Changes

DB Query	Delay		
	Distance Based	0.222 Sec	0.444 Sec
Statistic	**Average**	**Average**	**Average**
DB Query Response Time (sec)	0.24155	1.1597	2.1702
DB Query Traffic Received (bytes/sec)	13,252	5,574.7	7,221.8
DB Query Traffic Received (packets/sec)	25.883	10.888	14.105
DB Query Traffic Sent (bytes/sec)	13,252	5,575.4	7,223.6
DB Query Traffic Sent (packets/sec)	25.883	10.889	14.109

Table 4.12: Database Query comparison result of application uses database when delay changes

The impact of Delay variation on Database application analyzed by changing the Delay of AAU WAN links on simulation environment. Table 4.12 shows the effect of Delay change. Response Time (Sec) rises from 1.1597 to 2.1702 second when the Delay changes from 0.222 to 0.444 second. The number of packet sent and received increases even if effect of delay increases on the application. Generally when the delay increases the Response Time increases but Delay change shows no impact on the number of packet sent and received.

5.1.4 Scenario 4: Video

Simulation results of Video application when sharing link with other applications, when there is background traffic and when delay varies. Packet End-to-End Delay, Packet Delay Variation, Traffic Received and Traffic Sent used as a metrics.

5.1.4.1 When there is Background Traffic

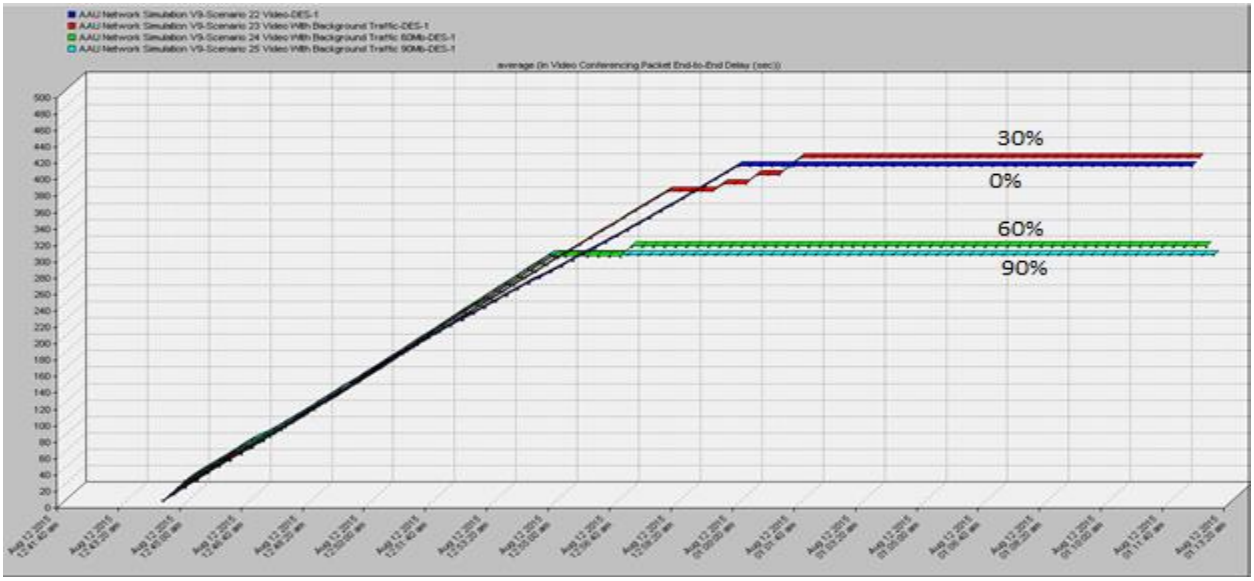


Figure 4.10: Average Packet End-to-End Delay of video application when there are background traffic takes the bandwidth from 0% up to 90%.

Video Conferencing

	0%	30%	60%	90%
Statistic	**Average**	**Average**	**Average**	**Average**
Video Conferencing Packet Delay Variation	11,286	17,839	12,289	10,783
Video Conferencing Packet End-to-End Delay (sec)	416.89	418.99	303.07	284.51
Video Conferencing Traffic Received (bytes/sec)	4,685	4,752	3,821	3,370
Video Conferencing Traffic Received (packets/sec)	0.2711	0.2750	0.2211	0.1950
Video Conferencing Traffic Sent (bytes/sec)	8,046,078	8,337,669	8,334,558	8,331,198
Video Conferencing Traffic Sent (packets/sec)	465.65	482.53	482.35	482.15

Table 4.13: Video statistics when there is background traffic

Only minor changes showed on Table 4.13 when the background traffic rises. Packet Delay Variation is around 10 to 17. Packet End-to-End Delay improved from 416 to 284 second. Packet Sent/Received per second also shows the same behavior. Generally Figure 4.10 and Table 4.13 shows that background traffic have only minor effect on video application.

5.1.4.2 Video When Sharing WAN Link

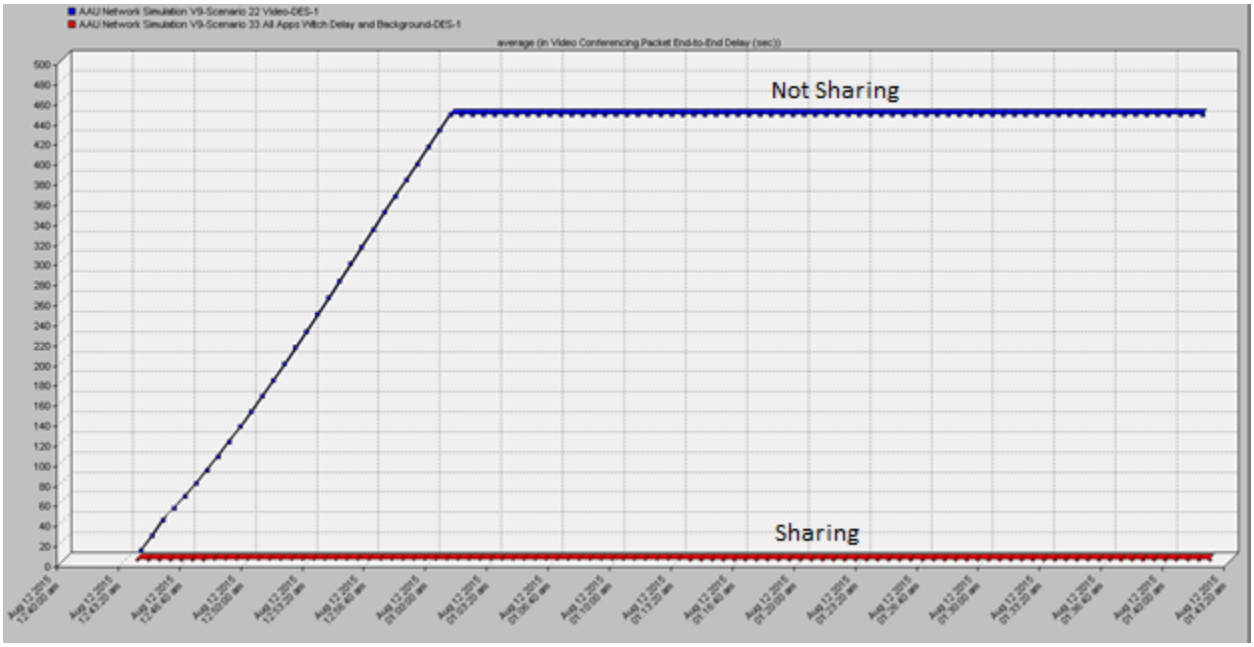


Figure 4.11: Video Conferencing Packet End-to-End Delay (Sec) comparison result of Video application when sharing and not sharing the WAN link

Video Conferencing	WAN Link	
	Not Shared	Shared
Statistic	**Average**	**Average**
Video Conferencing Packet Delay Variation	11,286	4,444
Video Conferencing Packet End-to-End Delay (sec)	416.89	0.2413
Video Conferencing Traffic Received (bytes/sec)	4,685	1,676,390
Video Conferencing Traffic Received (packets/sec)	0.2711	97.01
Video Conferencing Traffic Sent (bytes/sec)	8,046,078	4,179,034
Video Conferencing Traffic Sent (packets/sec)	465.65	241.85

Table 4.14: Video Conferencing Statistics when Video application using the WAN link with and without sharing with other application.

Table 4.14 shows that Video Conferencing Packet Delay Variation decreased from 11,286 to 4,444 this means the quality of the video conference increased. Video Conferencing Traffic Received (Packets/sec) increased from 0.2711 to 97.01 but Video conferencing Traffic Sent (packets/sec) decreased from 465.65 to 241.85. Sharing WAN link only affects the number of packet sent per second it didn't affect both Packet Delay Variation and the number of packet received per second. So we can conclude that sharing the network link have only slight impact on Video applications.

5.1.4.3 Video When Delay Changes

Video Conferencing		Delay	
	Distance Based	0.222 Sec	0.444 Sec
Statistic	**Average**	**Average**	**Average**
Video Conferencing Packet Delay Variation	12,064	5,879	2,197
Video Conferencing Packet End-to-End Delay (sec)	424.43	1.552	1.026
Video Conferencing Traffic Received (bytes/sec)	2,342	337,402	336,989
Video Conferencing Traffic Received (packets/sec)	0.1356	19.526	19.502
Video Conferencing Traffic Sent (bytes/sec)	8,343,039	4,839,461	5,334,792
Video Conferencing Traffic Sent (packets/sec)	482.83	280.07	308.73

Table 4.15: Average result of Video conferencing when the WAN link delay changed.

Delay has visible impact on multimedia applications (Video and Voice). This is because multimedia applications need real time interaction. For video conferencing packet need to be delivered to the other end without taking more delay. If there is a long delay the quality of the conference can be significantly poor.

On Table 4.15 the result of Video Conferencing Packet End-to-End Delay (sec) decreased from 1.552 to 1.026 second. 1.026 second have better quality than 1552 second. Generally when the delay greater the quality is poorer in other word the lower is the better.

The result obtained when the delay change on Table 4.15 showed that the delay have no impact on the number of packet sent and received. The number of packet received per second is constant. The number of packet sent increases from 280.07 to 308.73 per second.

5.1.5 Scenario 5: Voice

Simulation results of Voice applications when sharing link with other applications, when there is background traffic and when delay varies. Jitter, MOS Value, Packet End-to-End Delay, Packet Delay Variation, Traffic Received and Traffic Sent used as a metrics

5.1.5.1 Voice when there is Background Traffic

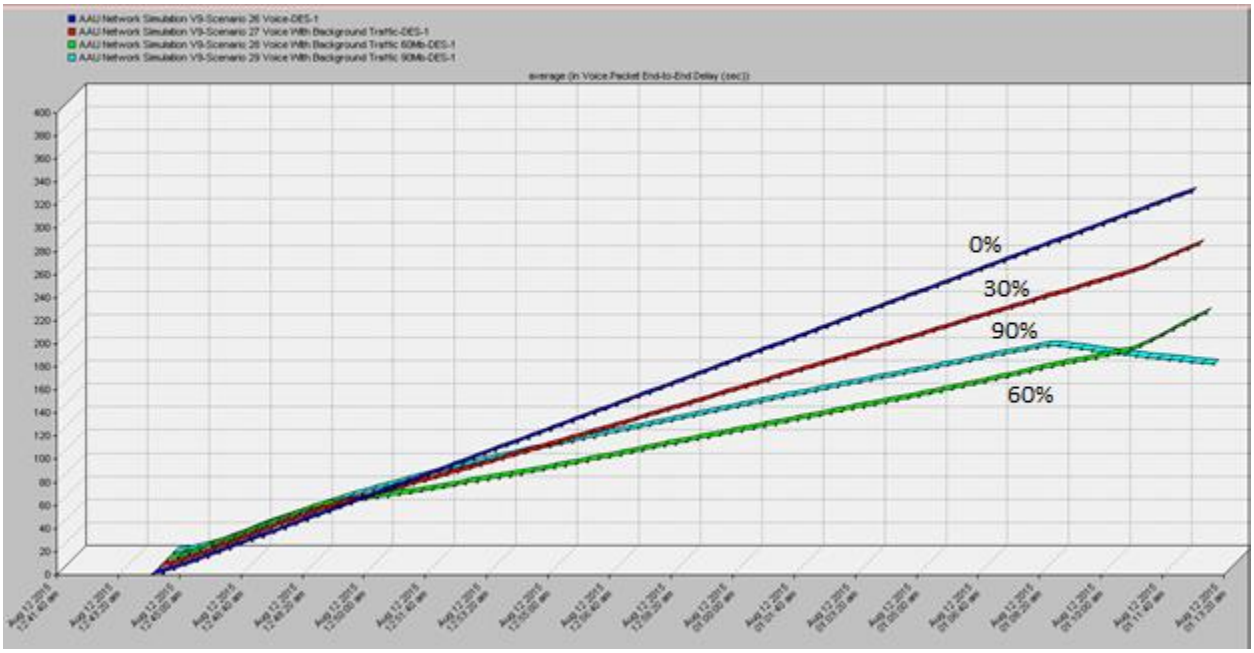


Figure 4.12: Average Packet End-to-End Delay (Sec) of Voice app simulated when the 0% to 90% of the bandwidth consumed by background traffic.

Voice				
	0%	30%	60%	90%
Statistic	**Average**	**Average**	**Average**	**Average**
Voice Jitter (sec)	0.060461	0.067902	0.080971	0.10819
Voice MOS Value	1.0314	1.0314	1.0314	1.0314
Voice Packet Delay Variation	24,428	21,459	18,365	7,720
Voice Packet End-to-End Delay (sec)	331.46	279.12	213.81	123.60
Voice Traffic Received (bytes/sec)	7,014.0	6,191.2	5,363.0	3,891.7
Voice Traffic Received (packets/sec)	350.70	309.56	268.15	194.58
Voice Traffic Sent (bytes/sec)	36,809	36,817	36,821	36,821
Voice Traffic Sent (packets/sec)	1,840.4	1,840.9	1,841.0	1,841.0

Table 4.16: Voice Statistics results taken from simulation by making the background to consume 0%, 30%, 60% and 90% of the bandwidth

Figure 4.12 illustrate that Voice application exhibited the same behavior with video application. Both voice and video application shows similar behavior when the background traffic increased but voice application perform better than video application. This fact presented on Table 4.16 which is Voice Jitter (sec), Voice Packet End-to-End Delay (Sec) and Voice Traffic Received (packet/sec) all them shows insignificant changes when the background traffic takes 0%, 30%, 60% and 90% of the link bandwidth. Finally we can conclude that voice application less affected than video application by background traffic.

5.1.5.2 Voice When Sharing WAN Link

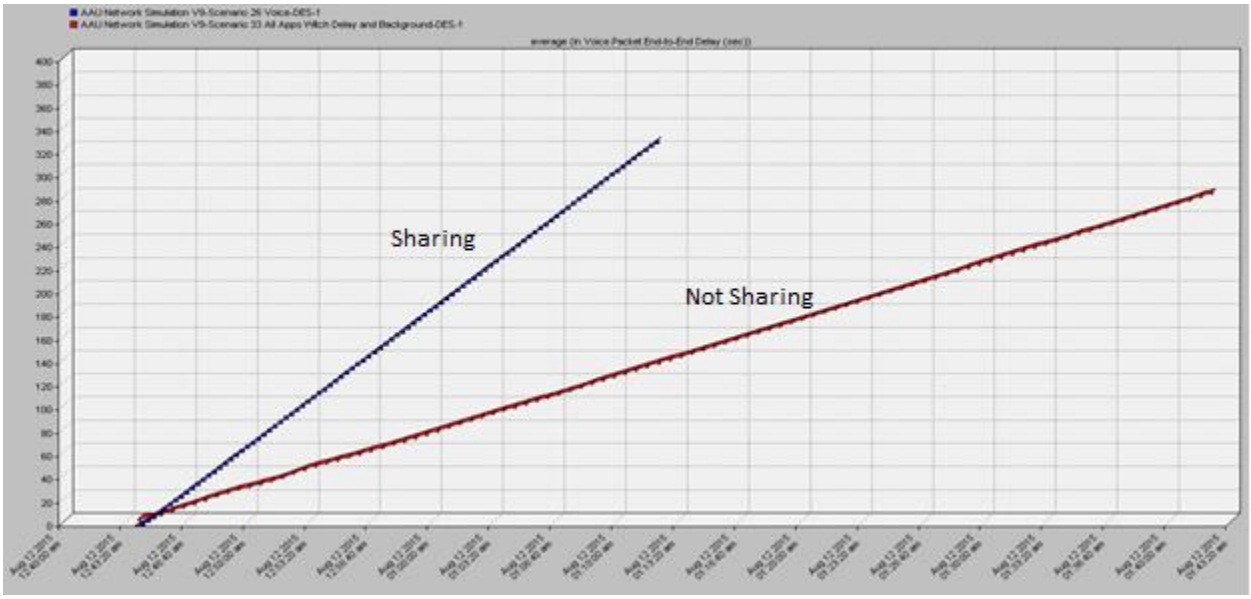


Figure 4.13: Voice Packet End-to-End Delay (Sec) comparison result of Voice application when the application use the WAN link by itself and when sharing the WAN link with other critical application.

Voice	WAN Link	
	Not Shared	Shared
Statistic	**Average**	**Average**
Voice Jitter (sec)	0.060461	0.025368
Voice MOS Value	1.0314	2.5687
Voice Packet Delay Variation	24,428	43,620
Voice Packet End-to-End Delay (sec)	331.46	281.24
Voice Traffic Received (bytes/sec)	7,014.0	6,950.3
Voice Traffic Received (packets/sec)	350.70	347.52
Voice Traffic Sent (bytes/sec)	36,809	25,039
Voice Traffic Sent (packets/sec)	1,840.4	1,251.9

Table 4.17: Voice Statistics of when the WAN link with and without sharing it.

When voice application share network links with other applications they have almost have no impact on voice application. On Table 4.17 Jitter decreased from 0.060461 to 0.025368 second. The second result have better quality than the first or it have better speech quality. Voice Packet Delay Variation increased from 24,428 to 43,620 this means when packet sent and received the delay lack consistency in other word there is delay fluctuation. Number of packet received per

second is closer 350.70 and 347.52. The number of packet sent per second decreased from 1,840.4 to 1,251.9. This shows that even if there is other applications using the WAN link but voice application not significantly affected.

5.1.5.3 Voice When Delay Changes

Voice	Delay		
	Distance Based	0.222 Sec	0.444 Sec
Statistic	**Average**	**Average**	**Average**
Voice Jitter (sec)	0.060461	0.040748	0.048282
Voice MOS Value	1.0314	1	1
Voice Packet Delay Variation	24,428	21,104	22,396
Voice Packet End-to-End Delay (sec)	331.46	308.07	320.20
Voice Traffic Received (bytes/sec)	7,014.0	7,008.6	7,006.8
Voice Traffic Received (packets/sec)	350.70	350.43	350.34
Voice Traffic Sent (bytes/sec)	36,809	42,485	44,345
Voice Traffic Sent (packets/sec)	1,840.4	2,124.2	2,217.3

Table 4.18: Comparison result of Voice application when the link delay changes.

Jitter indicates a variation in delay between arriving packets. Users often experience uneven gaps in speech pattern of the person talking on the other end and sometimes there are disturbing sounds over a conversation coupled with loss of synchronization. The time taken for a caller's voice at the source site to reach the other caller at the destination site is called latency. Network latency contributes to delay in voice transmission resulting in huge gaps between the conversation and interruptions [81].

The jitter on Table 4.18 scores only minor increment from 0.040748 to 0.048282 second. Voice Packet End-to-End Delay (Sec) also incremented from 308.07 to 320.20 second. The result on Table 4.18 help us to conclude when the delay increases voice application only decreases the quality but the amount of packet Sent/Received stay the same.

5.2 Discussion of Results

This section discusses the impact of network changes on Data and Multimedia application in group by comparing each other.

Effect of background traffic on HTTP, Email and Database Applications.

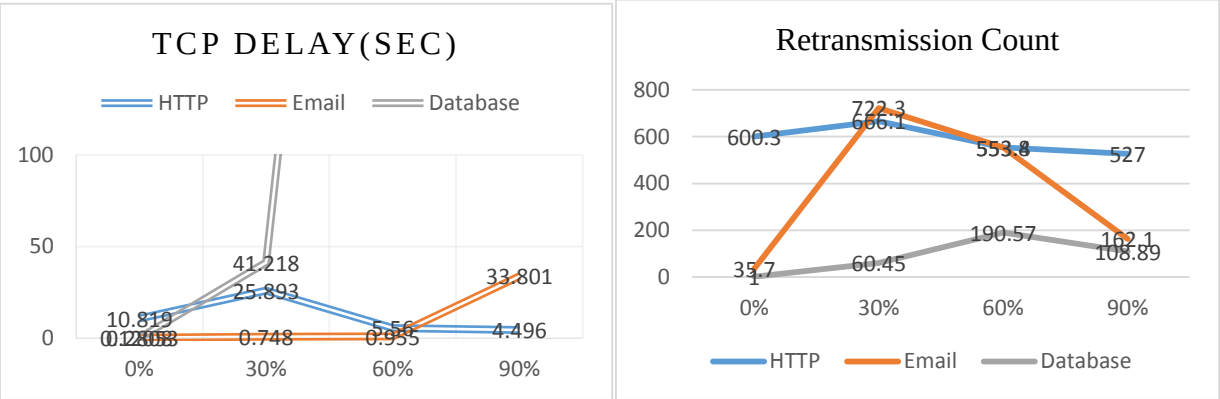


Figure 4.14: Comparison result of HTTP, Email and Database Applications when there is background traffic from 0% to 90 % of the bandwidth.

When the background traffic takes the bandwidth from 0% up to 90% data applications (HTTP, Email and Database) show different behavior. All of the three data applications are similar when there is no background traffic. But when the background traffic consumes 30% of the bandwidth TCP Delay becomes 0.748, 25.893 and 41.218 for Email, HTTP and Database Application respectively. If we measure the change by Retransmission Count 1, 35.7 and 600.3 the background traffic is 0 for Database, Email and Http respectively. Then all applications decrease in performance when there is background traffic but the magnitude is not identical for all application.

Database application highly affected than any other Data application by background traffic According to TCP Delay Metrics then HTTP application is the second data application affected by background traffic finally Email application lastly affected by background traffic based on TCP Delay.

According to Retransmission Count Metrics HTTP application highly affected than any other Data application by background traffic. Email application is the second data application affected by background traffic. Finally Database application is the last affected by background traffic based on TCP Delay.

Effect of sharing Link with other applications on HTTP, Email and Database Application.

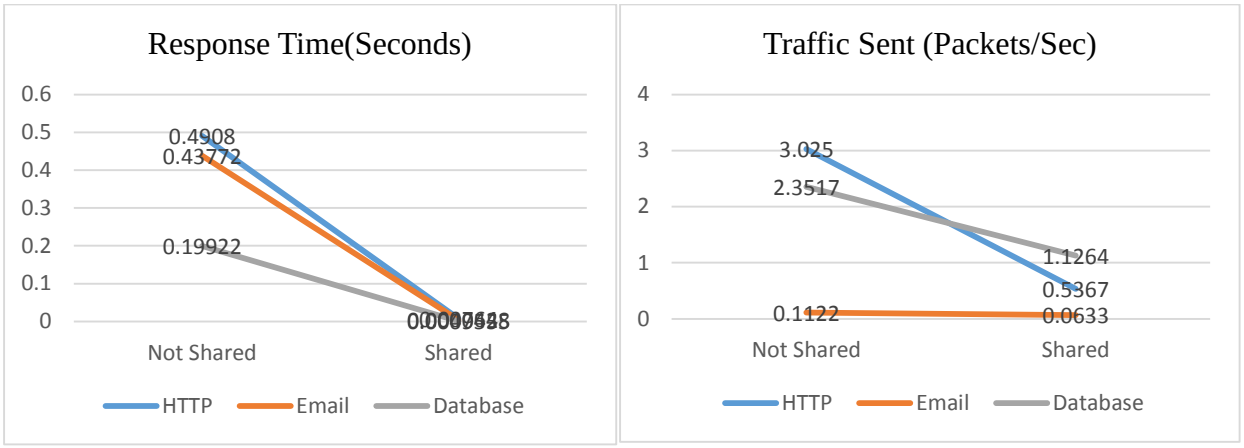


Figure 4.15: Comparison result of HTTP, Email and Database Applications when the network links shared by other application.

When data application share the network link with other applications the amount of Traffic Sent and received decreased significantly but the Response Time amazingly shows improvement. This is because of a decrease in the amount of packet sent and received. It is very difficult to select who highly affected based on Response time because they show almost the same result but When we list data applications from highly affected to lower according to Traffic Sent /Receive (Packet/Sec) HTTP application is highly affected when link shared than any other Data application; then Database application is the second affected one finally Email is the lest affected when the network link shared by all application.

Effect of Delay Change on HTTP, Email and Database Applications.

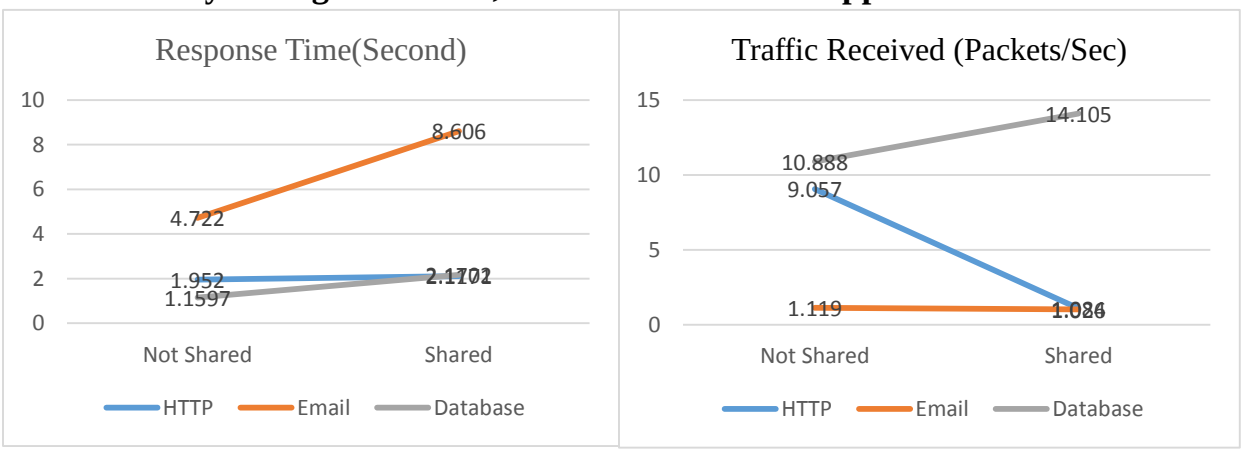


Figure 4.16: Comparison result of HTTP, Email and Database Applications when Delay changes from 0.222 to 0.444 second.

When the Link Delay changed from 0.222 to 0.444 second based on the Response Time (Second) HTTP highly affected from Data Applications then Database affected secondly then Email is the least affected data application affected by Delay change based on Response Time (Second) Metrics. When we measure the impact of Delay change using the amount of Packet sent and received per second HTTP holds the first place then Email is the second data application affected by Delay change finally Database application is the least affected Data application by Delay change.

Effect of background traffic on Voice and Video Applications.

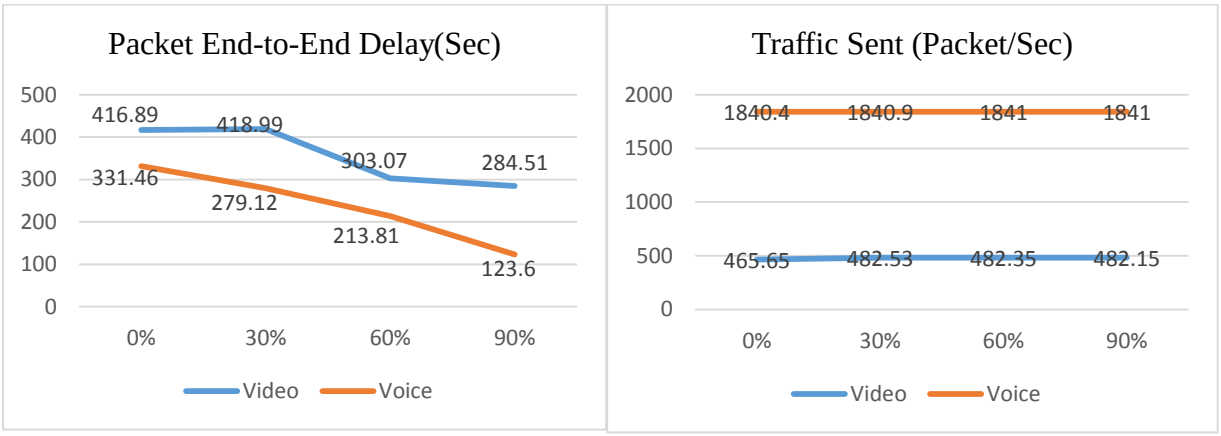


Figure 4.17: Comparison result of Video and Voice Applications when the background traffic takes 0 up to 90% of the bandwidth.

Multimedia application behaves very differently than Data applications the reason is both use different protocols. Based on Packet End-to-End Delay (Sec) and Traffic Sent (Packet/Sec) both multimedia applications which is Video and Voice behaves exactly the same when the background traffic consumes 0 up to 90% of the bandwidth.

Effect of sharing Link with other applications on Video and Voice Application.

Metrics	When Sharing Link			
	Video		Voice	
	Not Shared	Shared	Not Shared	Shared
Packet Delay Variation	11286	4444	24428	43620
Packet End-to-End Delay(Sec)	416.89	0.2413	331.46	281.24
Traffic Sent (Packet/Sec)	465.65	241.85	1840.4	1251.9

Table 4.19: Comparison result of Video and Voice application when the network link shared by other application.

Multimedia applications dropped both its quality and performance when it shares the Network link with other application. Packet Delay variation decreases for both Video and Voice application but video application affected more than audio. When measure the same scenario with Packet End-to-End Delay (Sec) voice application not affected as Mach as Video application. the final measurement when multimedia application share link with other application is Traffic Sent and Received (Packet/Sec) according to this Video application losses half of its Sent and Received packet after sharing the link with other application but Voice application didn't show decrement as Mach as Video application.

Effect of Delay Change on Video and Voice Applications.

Metrics	When Delay Changes			
	Video		Voice	
	0.222 Sec	0.444 Sec	0.222 Sec	0.444 Sec
Packet Delay Variation	5879	2197	21104	22396
Packet End-to-End Delay(Sec)	1.552	1.026	308.07	320.2
Traffic Sent (Packet/Sec)	280.07	308.73	2124.2	2217.3

Table 4.20: Comparison result of Video and Voice application when the Delay Changes.

The delay change have the same result showed when it shares the link with other applications. The same is true Video application affected more than Voice application when the delay changes.

Generally Video is highly affected by sharing network link and Delay change with other application than Voice application by all measurement this is because of two reasons the first one is Video application uses complex protocol than Voice application the second reason is Video Application sent large size of data as compared with Voice application.

Multimedia application behaves very differently than Data applications the reason is both use different protocols Multimedia uses UDP and Data uses TCP. The main difference between these two protocols is that TCP provides reliability and congestion control services, while UDP is orientated to improve performance.

5.3 AppDoctor Diagnoses Result

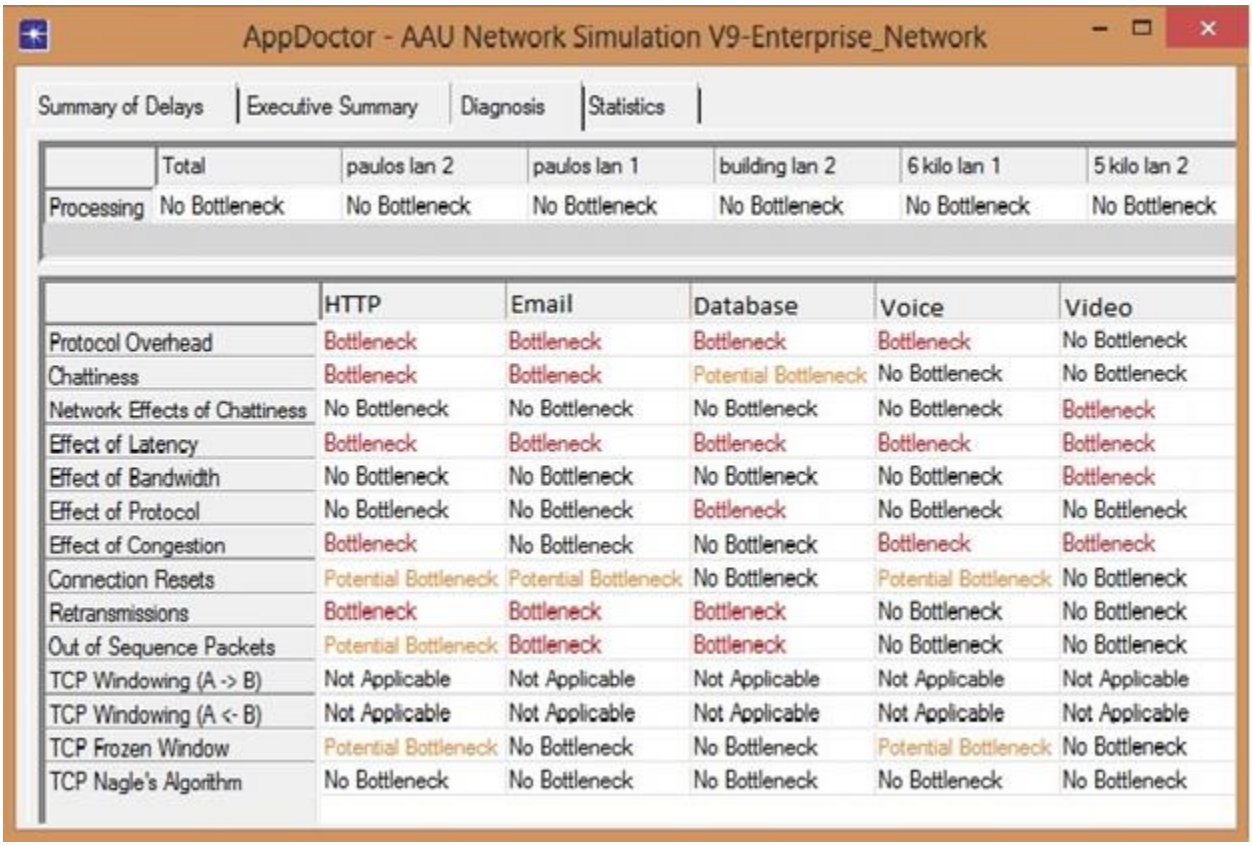
AppDoctor diagnoses tabbed page highlights bottlenecks and potential bottlenecks in application and network. AppDoctor uses specific thresholds to determine bottlenecks and potential bottlenecks. For most diagnoses categories, AppDoctor reports a bottleneck if the calculated value is above the threshold. The two exceptions are Chattiness Bottleneck and TCP Windowing Bottleneck; for these categories, AppDoctor reports a bottleneck if the calculated value is below

the threshold. AppDoctor reports a potential bottleneck if the calculated value is close to the threshold, but does not exceed it. AppDoctor never reports a full bottleneck for the Connection Resets Potential Bottleneck, because some applications (such as HTTP) use resets in their normal operations [80].

After applications tested when there is background traffic, when the link shared by all applications and when delay changes on simulation environment. The result generated from the simulation feed to AppDoctor to diagnosis the problem. Figure 4.18 shows the result obtained from AppDoctor for each critical applications. Generally data application (HTTP, Email and Database) experiencing the following bottlenecks Protocol Overhead, Chattiness, Effect of Latency, Protocol and Congestion, Retransmissions, Out of Sequence Packet Bottlenecks. Multimedia applications shows Protocol Overhead, Network Effect of Chattiness, Effect of Latency, Congestion and Bandwidth bottlenecks.

After importing application traffic flow data, network topology, bandwidth, latency and response time of AAU network to AppDoctor the result on Figure 4.18 found. AAU network experiencing the Bottleneck and Potential bottleneck listed on the Figure.

This study identifies AppDoctor results for each critical applications. This is because to identify and resolve the problems specific to individual application.



	Total	paulos lan 2	paulos lan 1	building lan 2	6 kilo lan 1	5 kilo lan 2
Processing	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck

	HTTP	Email	Database	Voice	Video
Protocol Overhead	Bottleneck	Bottleneck	Bottleneck	Bottleneck	No Bottleneck
Chattiness	Bottleneck	Bottleneck	Potential Bottleneck	No Bottleneck	No Bottleneck
Network Effects of Chattiness	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	Bottleneck
Effect of Latency	Bottleneck	Bottleneck	Bottleneck	Bottleneck	Bottleneck
Effect of Bandwidth	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	Bottleneck
Effect of Protocol	No Bottleneck	No Bottleneck	Bottleneck	No Bottleneck	No Bottleneck
Effect of Congestion	Bottleneck	No Bottleneck	No Bottleneck	Bottleneck	Bottleneck
Connection Resets	Potential Bottleneck	Potential Bottleneck	No Bottleneck	Potential Bottleneck	No Bottleneck
Retransmissions	Bottleneck	Bottleneck	Bottleneck	No Bottleneck	No Bottleneck
Out of Sequence Packets	Potential Bottleneck	Bottleneck	Bottleneck	No Bottleneck	No Bottleneck
TCP Windowing (A -> B)	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable
TCP Windowing (A <- B)	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable
TCP Frozen Window	Potential Bottleneck	No Bottleneck	No Bottleneck	Potential Bottleneck	No Bottleneck
TCP Nagle's Algorithm	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck

Figure 4.18: AppDoctor analysis result of AAU Network

Protocol Overhead Bottleneck

Large amounts of protocol overhead are increasing the utilization of the network. Protocol headers add overhead to each application message. Protocols also send packets that have no application data (such as TCP acknowledgement packets). This overhead can introduce delays by increasing congestion in the network; these delays can be especially significant if we are sending a large number of small application messages.

Chattiness Bottleneck

The application is sending many small requests and responses. Many small requests and responses make inefficient use of tier and network resources. The data sent per application turn is small. This might cause significant network delay. Additionally, a significant portion of application processing time can be spent processing requests and responses.

Network Effects of Chattiness Bottleneck

The application is incurring significant networking delays due to many application turns. Each time the conversation changes direction (an “application turn”), the packets incur a network delay while traversing the network. Mildly chatty applications suffer over high-latency networks (like WANs), while very chatty applications can suffer even over low-latency networks (like LANs).Interactive applications (such as telnet) tend to be chattier than non-interactive applications

Effect of Latency Bottleneck

The application is experiencing a significant bottleneck due to the time it takes packets to propagate across the network. Each time the application conversation changes direction (an "application turn"), the application waits for packets to propagate across the network. Propagation delay is typically a function of the speed of light and the distance traveled. Devices latencies also add to propagation delays.

Effect of Bandwidth Bottleneck

The application is experiencing a significant bottleneck due to the transmission speed. A packet's transmission delay is a function of the size of the packet. Lower transmission speeds cause larger delays.

Effect of Protocol Bottleneck

The application is experiencing a significant bottleneck due to protocol effects. Network protocols (such as TCP) frequently perform flow control, congestion control, and other effects that can throttle the rate at which applications send data. Other protocol effects that can impact application performance include retransmissions and collisions.

Effect of Congestion Bottleneck

The application is experiencing a significant bottleneck due to congestion effects. If high amounts of network traffic result in links that are heavily utilized (regardless of the available bandwidth on the link), the network will induce a variable queuing delay. This delay might throttle the rate at which applications send data. Other congestion effects that can affect application performance include retransmissions and collisions.

Connection Resets Potential Bottleneck

The application is experiencing an excessive number of connection resets. AppDoctor never reports a full bottleneck for this category, because some applications (such as HTTP) use resets in their normal operations. Connection resets can occur for a variety of reasons, including an

application is closing a connection, Delayed or duplicate connection control packets are received. And an application is trying to open a connection on a port where no application is listening. However, that some protocols such as HTTP frequently have many connection resets, which is normal.

Out of Sequence Packets Bottleneck

The Out of Sequence Packets bottleneck is similar to the Retransmissions bottleneck, in that both bottlenecks indicate the same underlying condition: that the transport protocol (such as TCP) is retransmitting packets at the source tier.

These two bottlenecks differ in that ACE Analyst usually detects retransmissions at the source, while it detects out-of-sequence packets at the destination tier. Thus if you captured traffic at both the source and the destination tiers, the statistics for these two bottlenecks should be about equal. If you captured at the source or destination tier only, you might see a discrepancy between these bottlenecks and their resulting statistics

Retransmissions Bottleneck

Retransmissions are significantly affecting application response times. Transport protocols such as TCP will retransmit packets if packets are lost or excessively delayed. This leads to longer application response times because: Data must be transmitted more than once and When TCP observes packet loss, it infers that the network is congested. This causes TCP to reduce the rate at which applications can send traffic. Retransmissions increase the likelihood of "TCP Windowing" bottlenecks because retransmissions cause TCP to shrink the Congestion Control Window.

Loss channels such as Frame Relay or ATM can allow applications to "burst" above sustainable data transmission rates. These "bursts" allow greater data rates, but packets within a burst are more likely to be dropped.

TCP Windowing Bottleneck

TCP Windowing is a significant bottleneck for data sent between host A and host B. TCP is reducing the rate at which the application can send data because of congestion control, flow control, send window sizes, and/or receive window sizes. When an application is sending bulk data over a high-bandwidth and high- latency network, TCP window sizes must be large enough to permit TCP to send many packets in a row without having to wait for TCP ACKs. TCP will only send data if the amount of sent-but-not-yet-acknowledged data is less than the minimum of the congestion control window, sender window, and receiver window sizes.

TCP Frozen Window

The advertised TCP Receive Window has dropped to a value smaller than the Maximum Segment Size (MSS). This is affecting your application response time. The advertised TCP Receive Window has dropped to a value smaller than the MSS. When this occurs, the sender cannot send any data until the receive window is one MSS or larger.

TCP Nagle’s Algorithm

Nagle's algorithm is present and is slowing application response times. Nagle's algorithm is a sending-side algorithm that reduces the number of small packets on the network, thereby increasing router efficiency. Nagle's algorithm is causing excessive numbers of delayed ACKs and is slowing down the application.

CHAPTER FIVE

OPTIMIZATION FRAMEWORK AND ALGORITHM

This chapter presents the optimization techniques used to resolve application network related problems and used to improve the performance of applications on network. This chapter also present the proposed application optimization framework and algorithm. The evaluation of the proposed optimization Framework also discussed in the final section of this chapter.

The phrase network and application optimization refers to an extensive set of techniques that organizations have deployed in an attempt to optimize the performance of networks and applications as part of assuring acceptable application performance. The primary role these techniques play is to Reduce the amount of data sent over the WAN, Ensure that the WAN link is never idle if there is data to send, Reduce the number of round trips (transport layer or application turns) necessary for a given transaction, Mitigate the inefficiencies of older protocols and Offload computationally intensive tasks from client systems and servers [50].

6.1 Optimization Techniques

This section describes the techniques used to optimize the network based on the problem and application. This study follows optimization technique which totally applied on server side so the technique need to be deployed on data center.

With the different techniques available for WAN optimization it is important to understand that simply turning on all available optimizations in every situation does not provide the best performance gains. Several considerations must be taken into account when determining the best set of optimizations to use for any given situation. The available optimization techniques are like a set of tools available to solve a problem. Selecting the right set of tools to use for a given situation is critical for maximum performance [50].

Section 4.1.1.1, 4.1.2.1and 4.1.3.1shows the impact of background traffic in Data Applications (HTTP, Email and Database) and Multimedia Applications (Video and Voice). According to the analysis result Data application highly affected by background traffic than multimedia applications. Even if the magnitude varies background traffic affects all critical applications. To increase the performance of critical application we need to decrease the effect of background traffic using optimization technique. There are well known network optimization techniques on application side but the following optimization techniques suggested to minimize the effect of background traffic on applications network access.

- Minimize chattiness of application traffic.
    - Minimize the chattiness of application by making applications send fewer, larger application messages.
    - Sending more packets in parallel.
- Small and infrequent keepalive
    - Keepalive means the duration of a protocol packet allowed to stay in the network. To make the network more optimize this study suggested to make the keepalive configuration of the router to be less than 10 seconds or to disable it at all.
- Controlling broadcast message

The above optimization techniques applied on the simulation environment by modifying some configuration on the network for instance increasing TCP Window Size to 64Kb/s, enabling window scaling, changing the Keepalive time of OSPF to 10s and configuring switch's to route packets.

Data Application						
Metrics	HTTP		Email		Database	
	Before	After	Before	After	Before	After
Response Time(Second)	27.951	0.01984	0.43772	0.0037362	0.24155	0.000949
TCP Delay (Second)	10.819	0.0293	0.2808	0.0293	0.1053	0.0293
TCP Retransmission Count	600.3	10.976	35.7	10.976	11	10.976
Traffic Received(Packet/Second)	3.0144	4.333	0.1122	0.1400	25.883	0.9233
Traffic Sent (Packet/Second)	3.0250	4.333	0.1122	0.1400	25.883	0.9400

Table 5.1: Comparison result before and after applying background traffic optimization for Data Applications.

Multimedia Applications				
Metrics	When Sharing Link			
	Video		Voice	
	Before	After	Before	After
Packet Delay Variation	11286	726.4	24428	0.024813
Packet End-to-End Delay(Sec)	416.89	0.2380	331.46	20.637
Traffic Received (Packet/Sec)	0.2711	81.81	350.70	266.10
Traffic Sent (Packet/Sec)	465.65	201.08	1840	1008.6

Table 5.2: Comparison result before and after applying background traffic optimization for Multimedia Applications.

When we come to latency Section 4.1.1.3, 4.1.2.3 and 4.1.3.3 shows the effect of latency. Latency demonstrated by using three method. The first one is distance based latency means the latency

determined based on the distance between source and destination. The second and third method is by configuring the latency 0.222 and 0.444 second on WAN links. The result shows that data application affected more than multimedia application so optimization technique need to be implemented on data application. To control latency we have two options the first one is controlling the network latency this is outside the scope of this study. The second method is controlling TCP Delay. TCP Delay is caused by two factors [80]. The first one is by TCP segment delay if TCP window size is small when an application is sending bulk data over a high-bandwidth and high-latency network. The second one is by High TCP retransmission count, means when there is congestion and TCP flow control mechanism is affecting communication [80].

Other network effect tested in this study is when application share the WAN link. Section 4.1.1.2, 4.1.2.2 and 4.1.3.2 describes the effect of sharing the link on critical application. Optimization technique needed to minimize the effect of other applications on critical application when they share the WAN link. To keep the performances of critical applications when the WAN link shared with non-critical applications the network traffic need to be controlled by giving priority to critical application using QoS.

Data Application						
Metrics	HTTP		Email		Database	
	Before	After	Before	After	Before	After
Response Time(Seconds)	2.1171	0.02087	0.707	0.003721	0.19922	0.0009617
TCP Delay (Sec)	5.56	0.029	0.955	0.029	41.218	0.029
TCP Retransmission Count	555.0	11.854	162.1	11.854	108.89	11.854
Traffic Received(Packet/Second)	1.084	0.4767	1.126	0.1667	1.12	0.8733
Traffic Sent (Packet/Second)	1.086	0.4767	1.126	0.1667	1.1264	0.9117

Table 5.3: Comparison result before and after applying Sharing link and Delay optimization for Data Applications.

Multimedia Applications				
Metrics	When Sharing Link			
	Video		Voice	
	Before	After	Before	After
Packet Delay Variation	11286	5522	24428	708.5
Packet End-to-End Delay(Sec)	416.89	0.8675	331.46	41.153
Traffic Received (Packet/Sec)	0.2411	81.97	300.7	293.40
Traffic Sent (Packet/Sec)	465.65	201.08	1020.4	1011.6

Table 5.4: Comparison result before and after applying Sharing link and Delay optimization for Multimedia Applications.

Based on the Bottleneck AppDoctor suggested different optimization techniques. The techniques are making applications send fewer, larger application message help to resolve protocol overhead, Chattiness and effect of latency bottleneck. Bandwidth and Congestion bottleneck can be resolved by sending less data, increasing link speed or using faster link and rescheduling the application to occur off-hours, or when there is less traffic.

For protocol bottleneck first checking for evidence of problems such as collisions at the MAC layer or retransmissions at the transport layer. If the application is using TCP check to see if the application is using Nagle's Algorithm or if "TCP Windowing" is reported as a bottleneck.

If there is Retransmission Bottleneck in the network reducing the likelihood of congestion in the network by increasing network capacity can resolve the problem. If possible determining where packet losses are occurring and determining if retransmissions are typical by analyzing the traffic also help to solve the retransmission problem. Another technique used to reduce Retransmission Bottleneck is by enabling TCP extensions (such as Selective Acknowledgements and Fast Retransmit/Fast Recovery) that increase the ability of TCP to cope with packet loss and retransmissions. The method of enabling these TCP extensions varies according to the operating system and we might need to upgrade the operating system version. If the network is using Frame Relay and/or ATM, decrease the "lossiness" by enabling traffic shaping. Traffic shaping keeps data transmission rates within the sustainable level which means packets are less likely to be dropped.

6.1 Proposed Optimization Framework

Optimization techniques try to minimize the effect of performance factors such as congestion, packet loss or latency on applications. In this study the optimization techniques are divided into two categories: WAN optimization and Quality of Service. Both techniques used to increase the network performance, the optimization technique will be selected and applied based on the network and application status.

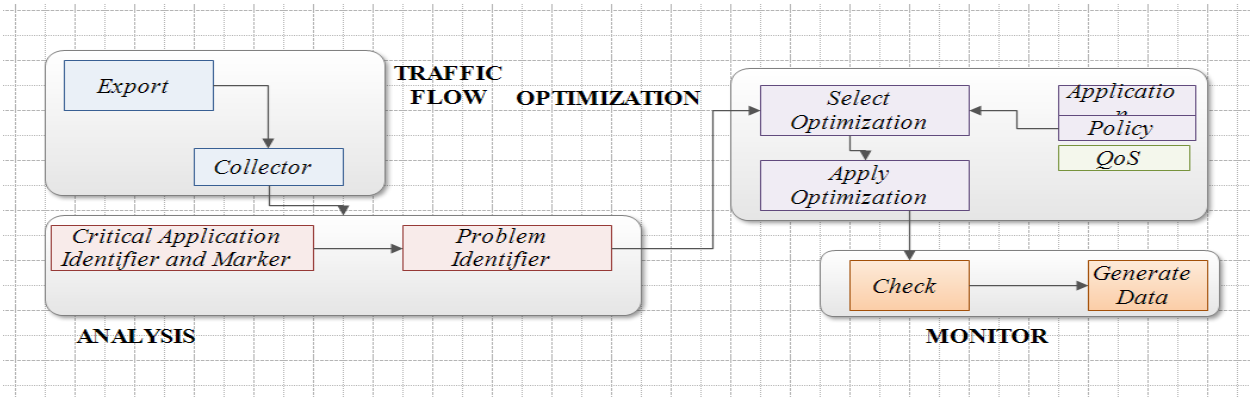


Figure 5.1: Proposed application aware WAN network performance optimization framework

First the problem identified by analyzing the network. In this study problems categorized in to two application bottleneck and application contention. If the problem is application contention QoS will be applied then the problem resolved by giving priority to critical applications. If there is bottleneck problem the type of bottleneck identified and optimization technique selected based on the application and the bottleneck.

The framework consists of four components Traffic Flow, Analysis, Optimization and Monitor which deployed on different parts of the network.

To elaborate the framework more precisely Figure 5.2 Show the step by step procedure with pseudo code, it accepts the input from online network traffic and configured network devices then both network performance and application statistics analyzed to identify the problem. For AAU network context this study categorized the problem in to two application contention and application bottleneck. Finally based on the problem and type of application optimization technique applied.

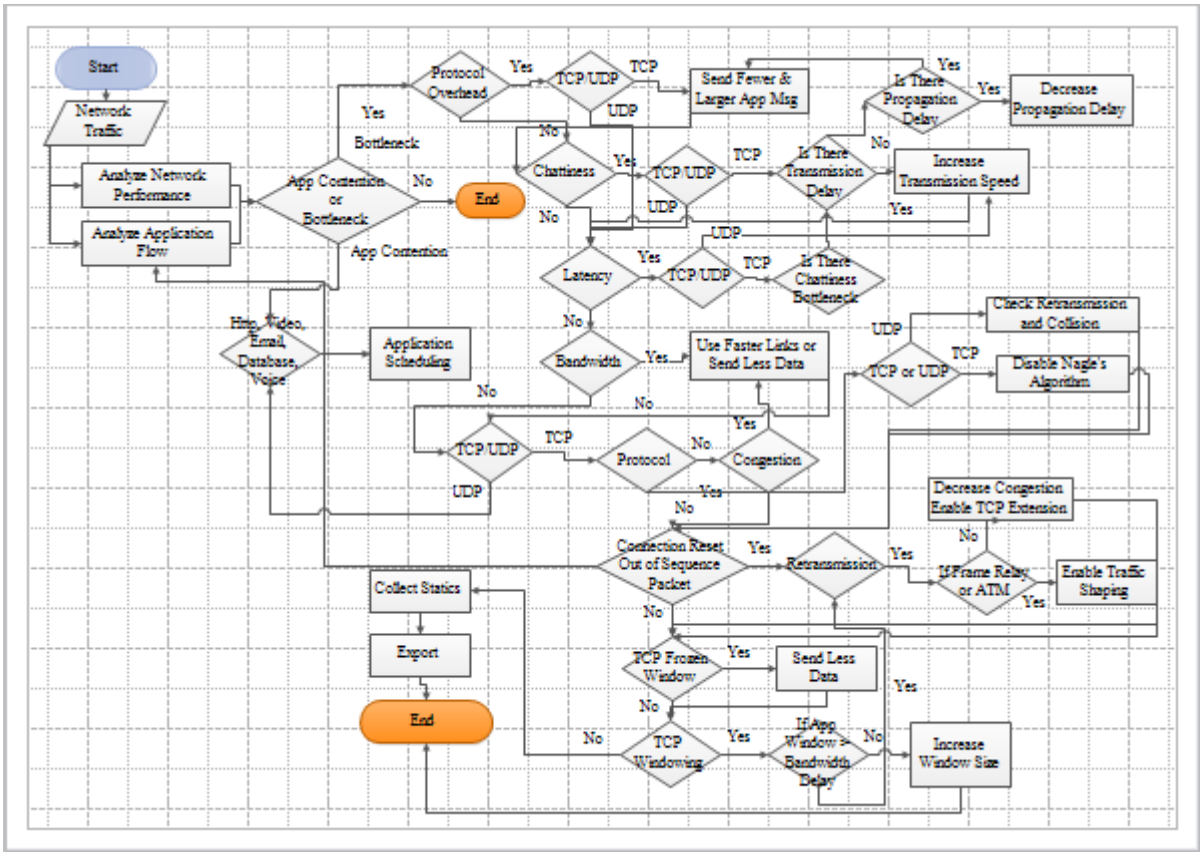


Figure 5.2: Pseudo code of proposed application aware WAN network performance optimization algorithm.

6.1.1 Traffic Flow

Traffic flow is the first step that helps identify applications and protocols that exist in the network. Various actions such as Analysis, Optimization, Control and Monitor will be performed after that by using the Traffic Flow output as an input on the next optimization phases with the end goal of improving both the application and network performance.

The traffic flow phase configures, collects, merge and export the traffic to analysis phase and it holds different components like Network traffic, Link statistics, Collect and merge and Export. Traffic flow is implemented on network devices and responsible for.

- Configuring devices to export traffic.
- Collect traffic from different sources.
- Merge the statistics based on the IP address, time, source, destination, application and layer 4 address.
- Store it for future use.
- Export traffic to analysis phase.

Traffic flow contains two major steps exporting and capturing traffic flows. Exporting is a configuration on layer 4 devices to export the traffic flow. The capturing process used to grasp the network traffics which generated on exporting phase then generating output used as an input to Analysis phase.

6.1.1.1 Export

This is the sub phase of Traffic Flow and which used to configure devices to export there traffic. There are two kinds of traffic; the online traffic and the NetFlow traffic. Online traffic is a request sent by application to access the network. NetFlow traffic on the other hand is a network traffic generated by a device after considering some matching criteria. Both traffics have their own role on the optimization process. NetFlow traffic is used to understand the general behavior of the network and the application but online traffic is the traffic on which the optimization applied. On Export stage the only thing we have done is configuring devices to export the two kinds of traffic.

```
1   //input Network Traffic
2   //output Generating NetFlow and mirroring
3   //Mirroring the Port Mirror the port to forwarded its traffic to the machine which the optimization applied
4   // Configure to generate Flow
5   If(the machine supports NetFlow)
6     Configure the machine to export NetFlow
7   ElseIf(The Machine didn't support NetFlow but supports SFlow)
8     Configure the machine to export SFlow
9   Else
10    Setup NetFlow recognizer and forward the port to that machine
```

Algorithm 5.1: Configuring devises to export network traffic

6.1.1.2 Collector

On collector stage there are three tasks; receiving exported traffic, storing the data and generating results by analyzing both the traffic and the stored data.

```
1   //input Exported Traffic
2   //output Selected Results, Store
3   //
4   Match //Match Packets based on source: destination ip address and source and destination port number
5   If (If the current and previous packets are similar in terms of Source_ip_address, Destination_ip_address,
    Source_port_no and Destination_port_no)
6   )
7       Store the result as a single Flow
8       Forward the Flow
9       Calculate   // Extract the following data then export to problem identification phase
10          Packet, TCP_retransmission  count,  MAC_collusions,
11          propagation_delay, TCP_delay,
12          Queue_delay, Connection_Reset,
13          Packet_Retransmission, Link_rate
14  Else
15   continue
16
```

Algorithm 5.2: Collector Algorithm used to collect traffic flow exported

6.1.2 Analysis

The analysis phase composed of two major components classification with marking and problem identification.

The analysis phase is accountable for the following tasks

- Accept organized traffic from traffic phase.
- Analyze the traffic.
- Identify the type of problem based on given threshold.
- Classify the packet according to the application type
- Mark the packet.
- Export the packet to optimization phase.

In analysis phase two major tasks are done in both network traffic and packets. First the network traffic received from traffic flow phase is analyzed to determine whether the type of network problem is contention or bottleneck if the problem is bottleneck then it identifies the type of bottleneck.

The second analysis is on packets; each packet passes on two processes. The first one determining or protocol the type of application. Second one is marking, marking is the process that colors the

packets (or just lets them through untouched) based on certain classification policies, marking have significant advantage on optimization phase. It will select optimization technique by referring the marking.

6.1.2.1 Critical Application Identifier and Marker

It accepts packet from traffic flow and identifies the type of application by looking layer 7 of the packet then marks the packet. All the optimization will be applied based on the tag marked on the header of the packet.

```
1   //input:        Packet that comes to layer 4
2   //output:       Tagged_Packet
3
4   If(Packet protocol == "TCP")   //If it is TCP based on the protocol type
5       If(Packet_Layer4_TCP == 80 or ==443)      //if it uses port 80 or 443 it is Http application
6           Packet_ToS_Precedence  == "101" && Packet_Id == 1//HTTP Packet Type of Service
7           Return Tagged Packet
8       Else if(Packet_Layer4_TCP == ( 25 or 110 or 143))     //if it uses port 110 or 143 it is Email application
9           Packet_ToS_Precedence  == "101" && Packet_Identification == 2    //Email Packet Type of Service
10          Return Tagged Packet
11      Else if(Packet_Layer4_TCP == ( 25 or 110 or 143))
12          Packet_ToS_Precedence  == "101" && Packet_Id == 3       //Database Application
13          Return Tagged Packet
14      Else
15          Packet_ToS_Precedence  == "000" && Packet_Id == 6       //Other Application
16          Return Tagged Packet
17  If (Packet protocol == "UDP")
18      If(Packet_Layer4_UDP == ( 25 or 110 or 143))
19          Packet_ToS_Precedence  == "101" && Packet_Id == 4       //Video Application
20          Return Tagged Packet
21      Else if(Packet_Layer4_UDP == ( 25 or 110 or 143))
22          Packet_ToS_Precedence  == "101" && Packet_Id == 5      //Voice Application
23          Return Tagged Packet
24      Else
25          Packet_ToS_Precedence  == "000" && Packet_Id == 6       //Other Application
26          Return Tagged Packet
```

Algorithm 5.3: Critical Application identifier and marker algorithm.

6.1.2.2 Problem Identification

Problem identification module is the most important phase of the optimization process, it identifies the type of application problem existed in the network. This study focuses in two types of problem application contention and application bottleneck. Problem identification output used as the basic ingredient when we select the optimization technique.

```
1   //input:        Tagged_Packet, TCP_retransmission  count,  MAC_collusions,  propagation_delay, TCP_delay, ,
2   //input:        Queue_delay, Connection_Reset, Packet_Retransmission, Link_rate
3   //output:       , Problem_Type, Bottleneck_Type Application_Type

4       If((TCP_delay  > TCP_delay_Threshold) & (TCP_retransmission  <
```

```
5           TCP_retransmission_Threshold)
6           Protocol_Overhead_Bottleneck = "Yes"          //There is Protocol overhead Bottleneck
7       Else
8           Protocol_Overhead_Bottleneck = "No"          //There is no Protocol overhead Bottleneck
9       EndIf
10      If(propagation_delay  > Latency_Bottleneck)
11          Effect_Of_Latency_Bottleneck = "Yes"          //There is  Effect Of Latency Bottleneck
12      Else
13          Effect_Of_Latency_Bottleneck = "No"          //There is no Effect Of Latency Bottleneck
14      EndIf
15      If (link_rate < Link_rate_Threshold)
16          Effect_Of_Bandwidth_Bottleneck = "Yes"          //There is Effect Of Bandwidth Bottleneck
17      Else
18          Effect_Of_Bandwidth_Bottleneck = "No"          //There is no Effect Of Bandwidth Bottleneck
19      EndIf
20      If ( (queue_delay > Queue_delay_Threshold) & (TCP_retransmission > TCP_retransmission_Threshold) & (
    MAC_collusions > MAC_collusions_Threshold)
21          Effect_of_Congesion_Bottleneck = "Yes"          //There is Effect of Congesion Bottleneck
22      Else
23          Effect_of_Congesion_Bottleneck = "No"          //There is no Effect of Congesion Bottleneck
24      EndIf
25      If (Connection_Reset > Connection_Reset_Threshold)
26          Connection_Reset_Bottleneck = "Yes"          //There is Connection Reset Bottleneck
27      Else
28          Connection_Reset_Bottleneck = "No"          //There is no Connection Reset Bottleneck
29      EndIf
30      If (Packet_Retransmission > Packet_Retransmission_Threshold)
31          Retransmission_Bottleneck = "Yes"          //There is Retransmission Bottleneck
32      Else
33          Retransmission_Bottleneck = "No"          //There is no Retransmission Bottleneck
34      EndIf
35      If ((Packet_Retransmission > Packet_Retransmission_Threshold) & (Effect_of_Congesion_Bottleneck ==
          "Yes" or Link_Error > Link_Error_Threshold))
36          Out_Of_Sequence_Packet_Bottleneck = "Yes"          //There is out of Sequence Packet Bottleneck
37      Else
38          Out_Of_Sequence_Packet_Bottleneck = "No"          //There is no out of Sequence Packet Bottleneck
39      EndIf
40      If ((propagation_delay  > Latency_Threshold) & (TCP_delay > TCP_delay_Threshold) & (if there is
    Frozen_Window)
41          TCP_Frozen_Window_Bottleneck = "Yes"          //There is TCP Frozen Window Bottleneck
42      Else
43          TCP_Frozen_Window_Bottleneck = "No"          //There is TCP no Frozen Window Bottleneck
44      EndIf
45
```

Algorithm 5.4: Application Network Problem identifier.

6.1.3 Optimization

Based on the analysis result the optimization phase will select optimization technique by considering the application type and the type of problem. This step is the most important step on the all WAN application optimization. There are different kinds of optimization techniques varies based on both network condition and specific application for example TCP flow optimization, Advanced Compression, Path optimization but this study select optimization technique by considering AppDoctor optimization recommendation and optimization technique easily

applicable on the real environment without abusing network devices like router CPU and other costly resources.

All the optimization technique will be applied based on the problem (bottleneck) and the application type. Optimization phase is responsible for

- Accepting input from analysis phase
- Chose optimization technique based on the bottleneck

```
1.      //input Protocol_Overhead_Bottleneck, Effect_Of_Bandwidth_Bottleneck, Effect_Of_Latency_Bottleneck,
2.      //input Effect_of_Congesion_Bottleneck
3.      //input Connection_Reset_Bottleneck, Retransmission_Bottleneck, Out_Of_Sequence_Packet_Bottleneck,
4.      //input TCP_Frozen_Window_Bottleneck
5.      //output Optimized application flow

6.    If(Application_Type == TCP)       //Identify if the application data or multimedia
7.      If(Protocol_Overhead_Bottleneck == "Yes" or Chattiness_Bottleneck == "Yes")
8.            Call Function TCP_Window
9.      EndIf
10.   If(Network_Effects_of_Chattiness_Bottleneck == "Yes")       //If there is chattiness
11.       If(Transmission_Delay_Bottleneck == "Yes")      //If there is Delay
12.           Call Function Background_Traffic
13.       EndIf
14.       If(Effect_Of_Latency_Bottleneck == "Yes")
15.           Call Function TCP_Window and Background_Traffic
16.       Else
17.           Call Function TCP_Window
18.       EndIf
19.     EndIf
20.       If(Effect_Of_Latency_Bottleneck == "Yes" )
21.           If(Chattiness_Bottleneck == "Yes")
22.              Go to Chattiness_Bottleneck      problem
23.           Else
24.             Select Closer Route and Background_Traffic and QoS
25.           EndIf
26.       EndIf
27.   If(Effect_Of_Bandwidth_Bottleneck == "Yes" or Effect_of_Congesion_Bottleneck == "Yes")
28.       Call Function QoS  and Background traffic
29.   EndIf
30.   If(Effect_Of_Protocol_Bottleneck == "Yes")
31.       If(There is  Collision or Retransmissions)
32.           Call Function Collision and Retransmission
33.       Else
34.           Disable Nagle's Algorithm      //only TCP windowing and Nagles will be resolved
35.           Call Function TCP Windowing
36.       EndIf
37.   EndIf
38.   If(Retransmissions Bottleneck == "Yes")
39.       Call Function TCP_Window, Background_Traffic and QoS      //all the optimization techniques applied
40.   EndIf
41.   If(TCP_Windowing_Bottleneck == "Yes")
42.       If(TCP_Window = Max_Unacknowledged Data Sent)
43.           Call Function TCP_Window   //Adjusting the window size
44.       EndIf
45.       If(Retransmission_Bottlneck == "Yes")
46.           Forward to Retransmission Bottleneck   // resolve the retransmission bottleneck by using application.
47.       EndIf
48.   EndIf
49.   If(TCP_Nagle's_Algorithm_Bottleneck == "Yes")
```

```
50.        Disable Nagles_Algorithn                    //solve nagles problem
51.    EndIf
52.   Function[TCP_Window]
53.  Function Background_Traffic ()                  //Adjusting the background traffic effect based the application
54.       If(Application == Data Application or TCP)
55.           If(Background Traffic > or equal to 40% of the bandwidth)
56.              Enable Collision Detection            // background traffic need to be adjusted.
57.              Enable QoS
58.       Else(Application == Multimedia Application or UDP)
59.       If(Application == Data Application or TCP)
60.           If(Background Traffic > or equal to 80% of the bandwidth)
61.              Enable Collision Detection
62.              Enable QoS
63.  EndFuntion
64.  Function QoS ()
65.          Start
66.          Applicaton Priority  //Assigning priority based on the application type
67.          Assign
68.                HTTP = Level 1
69.                Email = Level 2
70.                Database = Level 3
71.                Video = Level 4
72.                Voice = Level 5
73.                Other = Non-Critical
74.          If(Applicaton_Contention == "True")       //A command used to activate the Access Control list
75.              Enable ACL
76.                ACL {Assign Access Control List}
77.              Enable QoS
78.  EndFuntion
79.
80.
```

Algorithm 5.5: Optimizer Algorithm

6.1.4 Monitor

Successful application requires continuous process of identifying application on the network and ensuring acceptable business-critical application performance while controlling or eliminating non-critical applications. Controlling performance requires visibility into network and application behavior. Not only does monitoring verify that policies are correctly implemented, but data acquired through monitoring can drive the generation and enforcement of new dynamic policies.

The monitoring phase constantly check if the optimization is performing well and brought some improvement on the network by analyzing the traffics. Monitoring phase also responsible for generating data about the optimization and notifications to the system administrator within an interval of time.

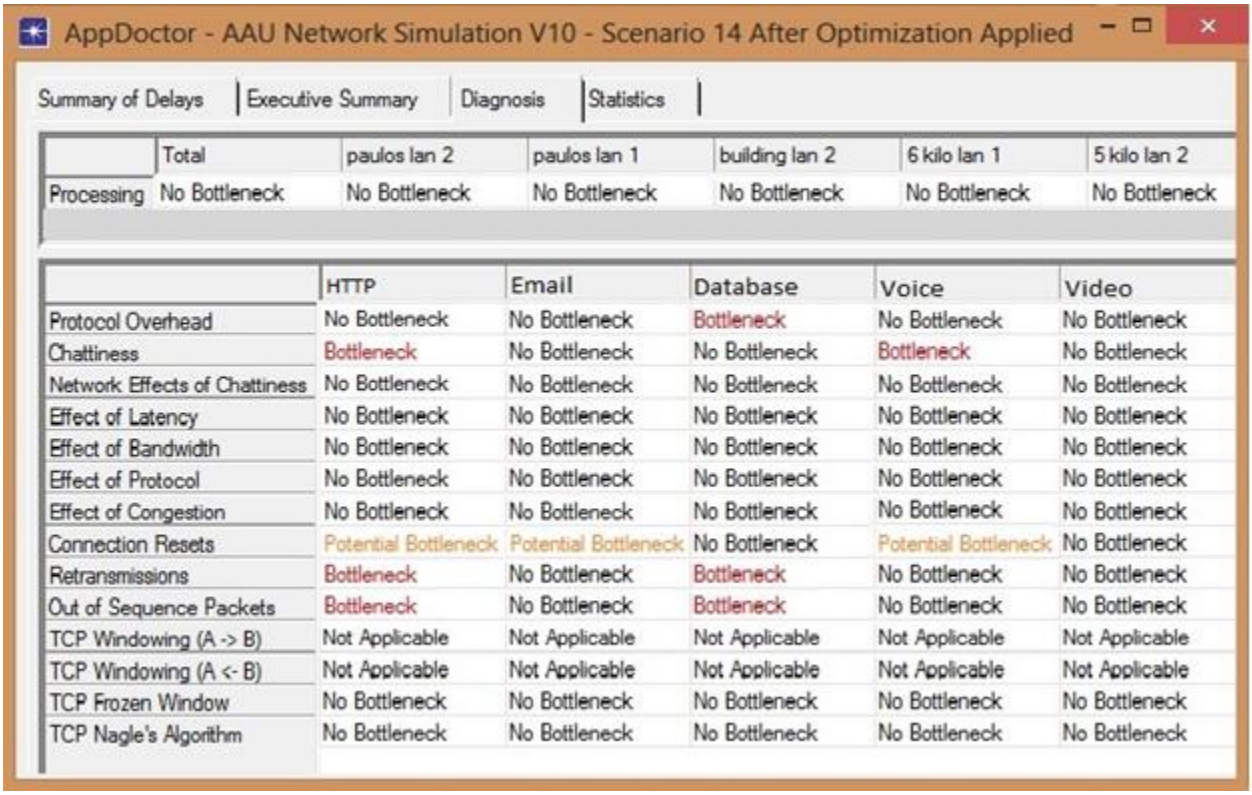
```
1        //input Optimized Network Traffic
2        //input Report
3        //Old Result form Problem Identifier
4     Function Measure()
5        TCP_retransmission  count,  MAC_collusions,
6        Propagation_delay, TCP_delay
7        Queue_delay, Connection_Reset, Packet_Retransmission,
8        Link_rate
```

```
9   Function Compare and Report()
10        If((New_TCP_delay  > TCP_delay_Threshold) & (New_TCP_retransmission  < TCP_retransmission_Threshold)
11          Report Protocol_Overhead_Bottleneck Steel Not Resolved
12      EndIf()
13      If(New_propagation_delay  > Latency_Bottleneck)
14         Report Effect_Of_Latency_Bottleneck Steel Not Resolved
15      EndIf()
16      If (New_link_rate < Link_rate_Threshold)
17         Report Effect_Of_Bandwidth_Bottleneck Steel Not Resolved
18      EndIf()
19      If ( (New_queue_delay > Queue_delay_Threshold) & (NewTCP_retransmission > TCP_retransmission_Threshold)
    & ( MAC_collusions > MAC_collusions_Threshold)
20         Report Effect_of_Congesion_Bottleneck Not Resolved
21      EndIf()
22      If (New_Connection_Reset > Connection_Reset_Threshold)
23         Report Connection_Reset_Bottleneck Not Resolved
24      EndIf()
25      If (New_Packet_Retransmission > Packet_Retransmission_Threshold)
26         Report Retransmission_Bottleneck Not Resolved
27      EndIf()
28      If ((New_Packet_Retransmission > Packet_Retransmission_Threshold) & (New_Effect_of_Congesion_Bottleneck
    "Yes" or New_Link_Error > Link_Error_Threshold))
29         Report Out_Of_Sequence_Packet_Bottleneck Not Resolved
30      EndIf()
31      If ((New_propagation_delay  > Latency_Threshold) & (New_TCP_delay > TCP_delay_Threshold) & (if there is
    Frozen_Window)
32         Report TCP_Frozen_Window_Bottleneck Not Resolved
33      EndIf()
34  EndFunction
35  Function Store()
36    Store Measured Results
37    Store Reports
```

Algorithm 5.6: Monitoring Algorithm

6.7 Validation

After applying proposed Optimization Framework on the simulation environment then AppDoctor generates the following result.



	Total	paulos lan 2	paulos lan 1	building lan 2	6 kilo lan 1	5 kilo lan 2
Processing	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck

	HTTP	Email	Database	Voice	Video
Protocol Overhead	No Bottleneck	No Bottleneck	Bottleneck	No Bottleneck	No Bottleneck
Chattiness	Bottleneck	No Bottleneck	No Bottleneck	Bottleneck	No Bottleneck
Network Effects of Chattiness	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck
Effect of Latency	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck
Effect of Bandwidth	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck
Effect of Protocol	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck
Effect of Congestion	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck
Connection Resets	Potential Bottleneck	Potential Bottleneck	No Bottleneck	Potential Bottleneck	No Bottleneck
Retransmissions	Bottleneck	No Bottleneck	Bottleneck	No Bottleneck	No Bottleneck
Out of Sequence Packets	Bottleneck	No Bottleneck	Bottleneck	No Bottleneck	No Bottleneck
TCP Windowing (A -> B)	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable
TCP Windowing (A <- B)	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable
TCP Frozen Window	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck
TCP Nagle's Algorithm	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck	No Bottleneck

Figure 5.4: Result after implementing some of the optimization techniques on AAU simulated environment

Retransmissions sometimes can't be resolved by optimization this is because the problems is not only the network but also related to the application installed on the user computer. The application reset its connection by its defect or error.

TCP Windowing (A → B) and TCP Windowing (A ← B) is not applicable because the study didn't able to find the packet source and destination.

When the Optimization Framework applied on simulated environment for all application the amount of Potential Bottleneck reduced by 60% and for HTTP application the amount of bottleneck reduced by 20%, For Email application the bottleneck reduced by 100%, For Database by 60%, For Voice by 66% and For Video by 100%. Generally the optimization model resolves approximately 69.2% of the total bottleneck problem the rest 30.8% doesn't solved because of several reasons like limitation on the optimization model capability, some problems can't solve by

simple optimization techniques for instance retransmission bottleneck may be occurred by application defect not by network problem and finally optimization model compatibility.

Strength of the Optimization Framework.

- It resolves 69.2% of application network.
- It improve both the performance and quality of the applications.
- It resolves not only bottlenecks but also application contention.
- It works based on the current status of the network.

Weakness of the Optimization Framework.

- It didn’t resolve all of the bottlenecks still 30.8% of the bottlenecks are unresolved by the proposed framework.
- The performance of HTTP application didn’t improved like other application.
- It is semiautomatic.

CHAPTER SIX

CONCLUSION AND FUTURE WORK

7.1 Conclusion

Network infrastructure need consistent analysis and optimization because of the demand and technology change. This study investigates application related network problems and their solutions in AAU network. The study totally focused on applications which is critical for the institution. The critical applications are HTTP, Email, Video, Database and Voice applications consecutively. AAU network and critical applications simulated and analyzed using OPNET modeler.

The analysis shows that Multimedia application behaves very differently than Data applications. The result also shows Data applications affected more than multimedia applications on different network factors. In addition HTTP application is most sensitive data application than others in network factors. Email application holds the second place finally Database application. When we come to Multimedia applications video application is more sensitive than voice application in network factors.

The root cause of the overall application problem on AAU network analyzed using AppDoctor. AppDoctor found out that for Data application the major bottleneck on the network is Protocol Overhead, Chattiness, Effect of Latency, Effect of Congestion, Effect of Protocol, Connection Resets and Retransmissions. For multimedia applications the major constraints or bottlenecks are Effect of Congestion, Network Effects of Chattiness, Effect of Latency and Effect of Bandwidth. To overcome the major bottlenecks diagnosis function of AppDoctor gives further insight into the cause of the bottlenecks and identify the problem precisely.

By considering the analysis result and optimization technique the study proposes a network application optimization framework for AAU network. The model are designed to improve the reliability, performance and delivery of critical applications across the network. The model follows comprehensive approach to solve application congestion and coalition problem by making five components work together which is Traffic Flow, Analysis, Optimization, Control and Monitor which deployed on different parts of the network. The model first accepts the packet and network traffic from traffic flow then the input will be analyzed to know if there is a bottleneck and to identify the application type. The next step is selecting the optimization based on the problem and application. Finally control and monitor phase executed to implement QoS if there is application contention and also to check if the optimization technique is effective.

The simulation results show that the proposed optimization resolves 69.2% of the bottleneck problem as a result the delay decreases and the amount of data sent and receive increases significantly with less error rate and this further enhances the network environment to behave more in a consistent manner with better quality and more additional feature which user can experience. The performance of optimizer.

7.2 Future Work

In this study application performance is optimized from the server side however there are application that need further performance optimization so it need further investigation in client side.

This study totally focused on application performances at AAU network but there are other works done in AAU on the network performance. In the future additional work need to be done integrating the application performance and the network performance.

This study analyzed a large-scale WAN network performance by using OPNET simulation tool, there are limitations from various aspects that retrain this study such simulation tools and hardware resources.

This study simulates and applied all the modifications, changes and optimization techniques in a simulation environment because of the reasons mention on the methodology section and the logical environment lack some physical behaviors like real distance, hardware faults and capacity and deference performances, so to make the result more accurate and applicable it need to be implemented on real network and real-time environments if there are enough hardware resources in the future.

Mapping the source and destination of a packet tried to cover in this study but it was unsuccessful in addition wireless network doesn't included because of economic and capacity reasons, since AAU Network have many wireless users it has significant impact on the final result of the research so future works need to be done including both.

This study can be improved in the future by enhancing different latest available applications and technologies, in addition the result may be improved by using latest trends, invasions and when more advanced application built.

In case of AAU research has been done on network performance and application performance area but related to security there is no research has been done yet so this area need to be explored.

Finally we strongly recommend the ICT office of AAU to look forward doing additional works to implement this study so as to resolve the current network problem and make the network more application aware than the current one.

References

[1] Pavel Masek, Dominik Kovac, Jiri Hosek, Mariya Pavlova, Ondrej Krajsa "Analysis of Network Parameters Influencing Performance of Hybrid Multimedia Networks", 2014

[2] W. Willinger, M. Taqqu, and A. Erramilli. "A Bibliographic Guide to Self-Similar Traffic and Performance Modeling for modern High-Speed Netowrks". 1996.

[3] Phat Hoang, "Internet Traffic Analysis", Master Thesis, Department of Electrical and Information Technology Faculty of Engineering, LTH, Lund University, Lund, Sweden, 2010.

[4] Cory Janssen, (2014) "Network Traffic Analysis [Online]".Available: http://www.techopedia.com/definition/29976/network-traffic-analysis, Accessed May 2015.

[5] Tomson, "A scalable architecture for network traffic monitoring and analysis using free open source software", Open Source Community, 2015.

[6] S. Chevul, L. Isaksson, M. Fiedleret al., "Measurement of Application-Perceived Throughput of an E2E VPN Connection Using a GPRS Network Wireless Systems and Network Architectures in Next Generation Internet", Lecture Notes in Computer Science M. Cesana and L. Fratta, eds., pp. 255-268: Springer Berlin / Heidelberg, 2006.

[7] M.Fomenkov, K.Keys, D.Morre, and K Claffy, "Longitudinal Study of Internet Traffic in 1998-2003", on Proceedings of the Winter International Symposium on Information and Communications Technologies WISICT‟04, January 2004.

[8] Shinya Furuta, "Measurement and Analysis of Network Traffic by Considering Applications Dynamism", Department of Information Networking, Graduate School of Information Science and Technology, Osaka University, February 16th, 2009.

[9] Tariq Aziz, Mohammad Saiful Islam. "Performance Evaluation of Real–Time Applications over DiffServ/MPLS in IPv4/IPv6Networks". Master's Thesis. School of Computing, Blekinge Institute of Technology, 371 79 Karlskrona, Sweden.

[10] Harish Kapri, "Network traffic data analysis", Department of Electrical and Computer Engineering, Louisiana State University and Agricultural and Mechanical College, December 2011.

[11] S. Alvarez, "QoS for IP/MPLS Networks", 1st ed. USA: Cisco Press, 2006.

[12] Cisco Systems inc, Cisco Systems NetFlow Services Export Version 9, 2012

[13] Cisco service data. Sheet CISCO SERVICES FOR CISCO NETWORK APPLICATION PERFORMANCE ANALYSIS SOLUTION. 2005

[14] Cisco 2014, online:- Available, http://www.cisco.com/, Accessed July 2015

[15] Vinod Mohan, "10 Best Practices to streamline Network Monitoring", SolarWinds Worldwide, LLC, 2013.

[16] YICHI ZHANG, "Residential Network Traffic and User Behavior Analysis", Master of Science Thesis Stockholm, Sweden, 2010.

[17] ANUKOOL LAKHINA, "Network-Wide Traffic Analysis: Methods and Applications," Ph.D. Dissertation, Boston University Graduate School or Arts and Sciences, August 2007.

[18] Victor S. Frost and Benjamin Melamed, "Traffic Modeling for Telecommunications Networks", IEEE Communications, Mar. 1994. http://ieeexplore.ieee.org/iel1/35/6685/00267444.pdf

[19] J. Schormans, and T. Timotijevic, "Evaluating the Accuracy of Active Measurement of Delay and Loss in Packet Networks Management of Multimedia Networks and Services," Lecture Notes in Computer Science A. Marshall and N. Agoulmine, Springer Berlin / Heidelberg, 2003, pp. 409-421.

[20] V. Paxson and S. Floyd, "Wide-area Traffic: The Failure of Poisson Modeling", IEEE/ACM Transactions on Networking, Jun. 1995. http://www.cs.ucsb.edu/~ravenben/classes/276/papers/pf95.pdf

[21] Hari Balakrishnan, Srinivasan Seshan. "Analyzing Stability in Wide-Area Network Performance", ACM SIGMETRICS conference on Measurement and Modeling of Computer System, University of California at Berkeley CA 94720, june 1997

[22] M. Crovella, "Network Traffic Modeling", In Tutorial at ACM SIGCOMM, Portland, August 2004.

[23] Tomas Sisohore, "Measuring & Modeling HTTP Media Stream In IP-Networks", Master Thesis, KTH Royal Institute of Technology, Stockholm, Sweden, 2007.

[24] Abdelnaser Adas, "Traffic Models in Broadband Networks", IEEE Communications Magazine, Jul. 1997. http://ieeexplore.ieee.org/iel5/35/13111/00601746.pdf?isnumber=&arnumber=601746

[25] A. Smith, "Comcast prevails over FCC in Web traffic fight", WWW.CNNMoney.com, April 6th 2010,http://money.cnn.com/2010/04/06/technology/net_neutrality_fcc_comcast/, retrieved April 9th, 2010.

[26] IETF Charter(ipfix). 2014 online:- Available, https://datatracker.ietf.org/wg/ipfix/charter/

[27] Krishnan Srinivasan, Sonics Inc, Erno Salminen: "A Methodology for Performance Analysis

of Network-on-Chip Architectures for Video SoC", Tampere University of Technology 2009. Available: http://www.design-reuse.com/articles/20623/performance-analysis-network-on-chip-video-soc.html. 2009

[28] Cristian Molaro,"Measuring application network performance using DB2 ping":online, Available: http://www.toadworld.com/platforms/ibmdb2/b/weblog/archive/2013/06/30/measuring-application-network-performance-using-db2-ping.aspx, Accessed July 2015

[29] K. Papagiannaki, N. Taft, Z. Zhang and C. Diot, "Long-Term Forecasting of Internet Backbone Traffic: Observations and Initial Models", In IEEE INFOCOM, San Francisco, April 2003.

[30] M. Crovella and E. Kolaczyk, "Graph Wavelets for spatial Traffic Analysis. In IEEE INFORCOM", San Francisco, April 2003.A. Nucci, R. Cruz, N. Taft, and C. Diot, "Design of IGP Link Weight Changes for traffic Matrix Estimation". In IEEE INFOCO, Hong Kong, April 2004.

[31] M. Roughan, A. Greenberg, C. Kalmanek, M. Rumsewick, J. Yates and Y. Zhang. "Experience in measuring backbone traffic variability: Models, metrics, measurements and meaning", In international Teletraffic conference(ITC-18), Berlin, September 2003.

[32] M. Roughan and J. Gottlieb, "Large scale measurement and modeling of backbone internet traffic", In SPIE ITCom, Boston, August 2002.

[33] S. Bhattacharyya, C. Diot, J. Jetheva, and N. Taft, "Pop-Level and Access-Link-Level Traffic Dynamics in a Tier-1 POP". In Internet Measurement Workshop. San Francisco, November 2001.

[34] Feldmann, A. Greenberg, C. Lund, N. Reingold, J. Rexford, and F. True. Deriving traffic demands for operational IP networks: Methodology and experience. In IEEE/ACM Transactions on Netrworking, pages 265-279, June 2001.

[35] Y.Zhang, M. Roughan, N. Duffield, and A. Greenberg. Fast Accurate Computation of Large Scale IP Traffic Matrices from Link Loads. In ACM SIGMETRICS, San Diego, June 2003.

[36] K. Papagiannaki, N. Taft, and C. Diot. Impact of Flow Dynamics on Traffic Engineering Design Principles. In IEEE INFOCOM, Hong Kong, April 2004

[37] Balakrishnan and chandrasekaran, "Survey of Network Traffic Models", behandrasekaran@wustl.edu, 2014.

[38] Stochastic processes for computer network traffic modeling Available at http://www.sciencedirect.com/science/article/pii/S0140366405000678, December 2005, Pages 1-23

[39] S.J. Lonberg, Published in: EUROCON 2003. Computer as a Tool. The IEEE Region 8 (Volume:1)Date of Conference: 22-24 Sept. 2003 Page(s): 316 - 320 vol.1 Print ISBN: 0-7803-

7763-X INSPEC Accession Number: 7861776 DOI: 10.1109/EURCON.2003.1248035, Publisher: IEEE

[40] K.Thompson, G.J.Miller, R.;Wilder, "Wide-area Internet traffic patterns and characteristics Network", IEEE , Volume 11, Issue 6, December 1997, pp.10-23

[41] Balakrishnan 2012, Online:- Available http://www.cse.wustl.edu/~jain/cse567-06/ftp/traffic_models3/, Accessed July 2015.

[42] Ejaz Ahmeda , Adnan Akhunzadaa , Md Whaiduzzamana , Abdullah Gania , Siti Hafizah Ab Hamida, , Rajkumar Buyyab,(2014, September) Network-centric performance analysis of runtime application migration in mobile cloud computing [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1569190X14001105, Accessed July 2015.

[43] Jie Sun, "Locality of Internet Traffic", Degree project in Communication Systems Second level, 30.0 HEC Stockholm, Sweden (jiesun@kth.se) KTH Information and Communication Technology, November 2012.

[44] R. H. Riedi and J. L. Vehel. "TCP traffic is multi fractal": a numerical study. Technical Report Research Report 3129, INRIA Rocquencourt, B.P. 105, 78153 Le Chesnay Cedex, France, 1997.

[45] P.Barford, J.Kline, D. Plonka and A. Ron. "A signal analysis of network traffic anomalies", In internet measurement workshop, Marseille, November 2002.

[46] Murray, David; Terry Koziniec (2012). "The State of Enterprise Network Traffic in 2012". 18th Asia-Pacific Conference on Communications (APCC 2012). 2012

[47] Techno Media,2014 online:- available, https://www.techopedia.com/definition/25597/computer-network, Accessed August 2015.

[48] Netflow, online:- Available , http://www.solarwinds.com/products/freetools/netflow_configurator.aspx, Accessed July 2015.

[49] KC Claffy, G.Polyzos, and H.Braun, "Tracking Long-Term Growth of the NSFNET", Communications of the ACM, vol.37, no.8, August 1994, pp.34-45.

[50] W.E. Lealand, M.S. Taqqu, W. Willinger, and D.V. Wilson. On the self-Similar Nature of Ethernet Traffic (Extended Version). Transactions on Networking, Pages 1-15, February 1994.

[51] DPS Telecom online, 2014:- Available, http://www.dpste, Accessed July 2015

[52] T.Karagiannis, A.Broido, N.Brownlee, KC Claffy, and M.Faloutsos, "Is P2P Dying or Just Hiding", IEEE COMM Globecom, November 2004.

[53] Dr. Jim Metzler, The 2009 Handbook of Application Delivery, Webotorials, , 2009

[54] Dr. Tim R. Norton, "End-To-End Response Time:Where to Measure?", Simalytic Solutions, LLC, 1999.

[55] G.S. Nagaraja, Ranjana R.Chittal, Kamod Kumar, "Study of Network Performance Monitoring Tools-SNMP," Sr. Lecturer, Dept. of Computer Science & Engg, R.V.C.E, Bangalore M. Tech,Dept. of CSE, VOL.7 No.7, July 2007.

[56] Luboˇs Pt ´aˇcek. Analysis and detection of Skype network traffic. DIPLOMA THESIS. MASARYK UNIVERSITY FACULTY OF I NFORMATICS. (2011)

[57] Harish Kapri. NETWORK TRAFFIC DATA ANALYSIS. Louisiana State University. The Department of Electricaland Computer Engineering. A Thesis Requirements for the degree of Master of Science in Electrical Engineering. Nagpur, India. December, 2011.

[58] D. Heyman and T.Laksham. "What are the implications of long-range dependence for VBR-Video traffic engineering?" IEEE/ACM Transactions on Networking. Pages 301-317, 1996.

[59] Damianos Gavalas. Mobile Software Agents for Network Monitoring and Performance Management. Phd Thesis. University of Essex. 2001

[60] C.Barakat and E.Altman, A Markovian. "Model for TCP Analysis in a Differentiated Services Network", INRIA, France, 2003.

[61] Network Instruments LLC(2008). Network Management White Paper. Monitoring and Managing Network Application Performance. www.networkinstruments.com

[62] Ádrian BonfáDrago, Anilton Salles Garcia, Maxwell E. Monteiro. A Methodology for Performance Management of Networks. Federal University of Espírito Santo. ISSN 1516-2338. 2001

[63] Application-Aware Networking at A Glance. The White Paper. MRV Communications. 300 Apollo Drive Chelmsford, MA 01824. www.mrv.com. 2013

[64] Muraleedharan N. Flow Based Traffic Analysis. C-DAC Bangalore. Electronics City. murali@ncb.ernet.in

[65] Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information.

[66] C.Barakat, P.Thiran, G.Iannaccone, C.Diot, and P.Owezarki, "A Flow-based Model for Internet Backbone Traffic", ACM SIGCOMM Internet Measurement Workshop, November 2002.

[67] KC Claffy, HW Braun, GC Polyzos, "A parameterizable methodology of Internet traffic flow profiling",- Selected Areas in Communications, IEEE Journal - OCTOBER 1995.

[71] Addison Wesley and Subramanian M, "Network Management Principles and Practice", 2001

[72] Dag, Online, Accessible at. http://dag.cs.waikato.ac.nz/.

[73] S. Savage. Sting: a TCP-based network measurement tool. In Proc. of the 1999 USENIX Symposium on Internet Technologies and Systems, October 1999

[74] Joel Sommers, Paul Barford, Nick Duffield, and Amos Ron. Improving accuracy in end-to-end packet loss measurement. In ACM SIGCOMM, August 2005.

[75] Online, Accessible at Iperf. http://dast.nlanr.net/Projects/Iperf/. Accessed 2015.

[76] Online, Accessible atTtcp. http://ftp.arl.mil/˜mike/ttcp.html, Accessed August 2015.

[77] P.Borgnat, G.Dewaele, K.Fukuda, P.Abry, and K.Cho, "Sketching the Evolution of Internet Traffic", IEEE Infocom 2009 proceedings, April 2009, pp.711-719.

[78] R. Swale, Voice over IP: Systems and Solutions. UK: The Institution of Engineering and Technology, 2001.

[79] T.Karagiannis, M.Molle, and M.Faloutsos, "Long-Range Dependence Ten Years of Internet Traffic Modeling", IEEE Internet Computing, Sep-Oct 2004, pp.57-64.

[80] OPNET Modeler Documentation, OPNET Modeler 2008

[81] NefFlow Analyzer, "Bandwidth Monitoring and Traffic analysis"- User Guide, professional and professional plus Edition, 2010.

[82] Yakob Gobena, "Developing WAN Optimization Model to improve the Performance of Business Critical Application, on UNECA", Master's Thesis in Addis Ababa University, July 2013.

[83] Tsegayea, "Network Analysis to optimize WAN performance", Addis Ababa University Information Science Department, 2014.

Declaration

I, the undersigned, declare that this thesis is my original work and has not been

Presented for a degree in any other university, and that all sources of materials for

The thesis have been duly acknowledged.

_____

Faris Awol

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

_____

Million Meshesha (PhD)