University of Fort Hare
Together in Excellence

An Assessment of Quality, Class and Grade of Service (QoS, CoS and GoS) over Worldwide Interoperability for Microwave Access (WiMAX) Networks Through Performance Evaluation of Bandwidth

A dissertation submitted to the Department of Computer Science in
fulfillment of the requirements of the degree of

Master of Science
in
Computer Science

Researcher:

Phumzile Nomnga

Supervisor:

Mr. M.S. Scott

Co-Supervisor:

Prof. K. Sibanda

DECLARATION

I, the undersigned, hereby declare that the work contained in this dissertation is my own original work and has not been previously submitted at any educational institution for a similar or any other degree. Information extracted from other sources is acknowledged as well.

Name: Phumzile Nomnga
Date: 27th November 2016

ACKNOWLEDGEMENTS

Glory be to **God**, for providing me and all the minds behind this work with the necessary strength, zeal and patience from day one, throughout the years and up until the last day. Indeed God is good.

I will forever remain grateful to my Supervisor, **Mr. Mfundo Shakes Scott**, for the father-figure role he has played to me over the past couple of years of research and learning. Heavenly commends are more precious than the earthly commends, I hope good Lord will commend you one day. In the same vein, I would also like to extent my words of gratitude to my Co-Supervisor, **Prof Khulumani Sibanda** for his uncompromising and courageous inputs on this work which, when addressed, would give further assurance that research quality and standards were met.

To my sponsors, **GMDRC**, **Telkom CoE** in **ICT4D** and **NRF**, I sincerely thank you for settling my university fees.

To my parents, **Mr Xolile Adolphus Nomnga** and **Mrs Nomvume Adelaide Nomnga**, my pillars of strength, my pride, and joy; I will always cherish your firm and excellent parenting principles and values. Subsequently, this work serves as my fitting gesture of gratitude for all that you have done for me since my first day on this side of the world.

To my extended family whom have become my nuclear family in the form of **Granny**, **Daps Mbuyi**, **Daps Doris** and the families of **Sakata**, **Madolo**, **Khala** and **Waqu**; because of you I have grown to appreciate the fact that it takes a family to raise a child even before the involvement of a village. I thank you all, family.

To my siblings**, Bongiwe**, **Simphiwe** and **Khayakazi** and my Brother-in-law **Mzamo** without forgetting **Itha** and **Qaqi**, for lack of a better word, **thank you** for your support and understanding when research work had to take me away from you during these years and in spite of this, you showed the young brother some love. *Ndiyabulela Qiyaz.*

Friends and Colleagues, **Ndiya, Kalawa** aka **Bawokazi, Paps**, **Sikiza**, **Zulu** and many others whose support lifted me up, I appreciate it beyond the words could ever explain. The phrase "***That's what friends are for***" finds relevance and meaning in you. A special thanks to **YamYam**.

Last but certainly not least, my 5 year old daughter, **Asemahle** aka **Sese**, to whom I dedicate this work and achievement; I thank you baby for your love. Just a thought about you, when research began to take its toll, was enough to lift me up and spur me to work again. You are one sweet and beautiful soul I could not fail, **Sese**.

ABSTRACT

The Dwesa WiMAX network provides broadband communications over wireless connectivity for various types of multimedia traffic, such as emailing, browsing, VoIP, file transfer, etc. to the community members. The community members of Dwesa use schools' computer labs to access the network and generate the aforementioned multimedia packets on dedicated timeslots and thus cause network congestion during such timeslots. Against this background, WiMAX implementation has faced several challenges in living up to its objectives in RMAs. Quality of Service (QoS) degradation as a result of high traffic demands remains one of the challenges thwarting WiMAX implementation. The GoS is also bound to get compromised as connectivity demands arise consistently with more subscribers connecting to the network, making it difficult to measure the success a subscriber is expected to have in accessing the network. The CoS and SchedType play a significant role in the redistribution of the available bandwidth to all bandwidth requests. This research project exploits this avenue to assess the resultant degradation of QoS and GoS caused by the inconsistent availability of bandwidth as redistributed by the CoS combination with a SchedType. The four CoS which are, namely, the UGS, rtPS, nrtPS and BE were implemented with the different SchedTypes, namely, MBQOS, FCFS and rtPS. Although the implementation process was conducted in a simulated environment using NS-3, the simulated network emulated the network setup implemented in Dwesa. The outcomes of the implementation suggests that certain combinations of the CoS's with SchedTypes can lead to degradation of QoS whilst some combinations can redistribute the available bandwidth to ensure the provisioning of guaranteed QoS.

TABLE OF CONTENTS

TABLE OF ACRONYMS

Acronym	Definition
AODV	Ad hoc On Demand Distance Vector
API	Application Programming Interface
ATM	Asynchronous Transfer Mode
Ave.	Average
BCD	Blocked Calls Delayed
BE	Best Effort
Bps	Bits per second
BR	Bandwidth Request
BRH	Bandwidth Request Header
BS	Base Station
bsDev	NetDeviceContainer (Base Station)
bsNode	Base Station Node
BSPK	Binary Phase Shifting keying
BWA	Broadband Wireless Access
CAC	Connection Admission Control
CBR	Constant Bit Rate
CID	Connection Identifier
Clang	C language
CoE	Centre of Excellence
CS	Convergence Sub layer
CoS	Class of Service
DCD	Downlink Channel Descriptor
DL	Down Link
DSDV	Destination Sequenced Distance Vector
DSL	Digital Subscriber Line
DSR	Dynamic Source Routing
E1	E carrier 1
ertPS	extended real time Polling Service
FCFS	First Come First Serve
FDD	Frequency Division Duplex
FED	Forward Error Correction
FIN	Finish
FTP	File Transfer Protocol
GHz	Gigahertz
GNU	GNU’s Not Unix
GCC	GNU Compiler Collection
GMS	Grant Management Sub-header
GoS	Grade of Service
IEEE	Institute of Electrical and Electronics Engineers

IEEE 802.16	Fixed Broadband Wireless Access System
IEEE 802.16e	Mobile Broadband Wireless Access System
ICT	Information and Communication Technology
ICT4D	Information and Communication Technology for Development
ID	Identity
IP	Internet Protocol
Iperf	Internet Performance Working Group
ITU	International Telecommunication Union
LCC	lost Call Cleared
LCD	Lost Call Delayed
LM	Link Manager
LOC	Line of Sight
MAC	Media Access Control
CPS	Common Part Sub-layer
Mbps	Megabits per second
MAN	Metropolitan Area Networks
MAN	OFDM
OFDMA	Orthogonal Frequency Division Multiplex-Access
SC	Single Carrier
mOCSA	Merging One Column Striping with non-increased Area
MPEG	Motion Pictures Experts Group
MS	Mobile Station
nrtPS	None real- time Packet Service
NS-3	Network Simulator version 3
NWG	Network Working Group
OFMDA	Orthogonal Frequency-Division Multiplex Access
OCSA	One Column Striping with non- increased Area
OS	Operating System
PC	Personal Computer
Pcap	Packet Capture
PDR	Packet Delivery Ratio
PDUs	Protocol Data Units
PHY	Physical Layer
PLR	Packet Loss Ratio
PMTs	Performance Monitoring Tools
PMP	Point-to-Multipoint
PS	Physical Slots
PSTN	Public Switch Telephone Network
QoS	Quality of Service
RST	Reset

rtPS	Real- time Polling Service
R&D	Research and Development
RMA	Rural Marginalized Area
RNG-REQ	Ranging Request
RNG-RSP	Ranging Response
RU	Rhodes University
SDU	Service Data Units
SFID	Service Flow Identifier
SLL	Siyakhula Living Labs
SS	Subscriber Station
ssDevs	NetDeviceContainer (Subscriber Station)
ssNodes	Subscriber Station Nodes
SU	Subscriber Unit
T1	Transmission Career 1/digital signal 1
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TDD	Time Division Duplex
TORA	Temporally- Ordered Routing Algorithm
ToS	Type of Service
UCD	Uplink Channel Descriptor
UDP	User Datagram Protocol
UFH	University of Fort Hare
UGS	Unsolicited Grant Service
UL	Up- Link
VoIP	Voice over Internet Protocol
VSAT	Virtual Aperture Terminal Satellite
Wi	Fi – Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WMAN	Wireless Metropolitan Area Networks

LIST OF FIGURES

LIST OF TABLES

CHAPTER ONE

INTRODUCTION AND BACKGROUND

1.1. Introduction

In this research project, through a network simulation process, an in-depth study and analysis of the WiMAX network deployed at Dwesa, one of the rural communities of the Eastern Cape Province of South Africa, was conducted. The undertaking of this research project was informed by the poor Quality of Experience (QoE) expressed by the Dwesa residents and observed by the network technicians from the University of Fort Hare (UFH). The Dwesa WiMAX network proved to be considerable slow with the growing internet access numbers in the community and this translated to lower throughput of the network. This was experienced even more during the computer literacy and computing trainings conducted by project collaborators, the Rhodes University and UFH. During these trainings, as computer laboratories are hosted in the local schools, residents would generate a significant amount of data as required by training exercises in their respective schools at once. The bandwidth provided across the SS channels would prove less capacitated to process all bandwidth requests of less or more similar priority level and thus result to network deterioration and complaints by the residents. Towards alleviating this setback, at the wake of growing internet access generally, this research study was undertaken

This study was also in line with the actual ICT infrastructure deployed at Dwesa and some of the ICT and network services provided by the existence of such a platform. Consecutively an in-depth investigation of QoS, CoS and GoS and their purpose and importance in WiMAX networks was undertaken prior to making any research propositions, scenarios or cases. Furthermore, an investigation of suitable bandwidth monitoring tools for WiMAX networks was carried out in order to select the appropriate ones to assist in the assessment of QoS, CoS and GoS and performance evaluation of bandwidth. “For large-scale wireless networks, such as WiMAX networks, deployments are expensive and cover very large areas, making simulation models very important both for development and planning purposes” as well articulated by Thomas (2011).

Additionally, Chaudhari and Karule (2014) identify two immediate challenges in setting up the WiMAX network in the form of time, as the process tends to be lengthy and simultaneously costly for researchers. However, they point to testing the WiMAX networks using free and open source software available for this particular purpose as the amicable solution. Consequently, the deployed Dwesa WiMAX networks setup was emulated and simulated into NS-3 model. The advantage of using this model is that “the users of NS-3 can construct models and simulate them on computers using models of traffic generators, protocols such as TCP/IP, devices and channels such as Wi-Fi, WiMAX and analyze or visualize the results,” as noted by Chaudhari and Karule (2014). Therefore this research project as well focused on compatibility of NS-3 to construct models and simulate in computer making use of traffic generators models, TCP/IP protocols and WiMAX as a channel for analysis and virtualization of the results.

According to Chaudhari and Karule (2014) WiMAX is a wireless communication standard designed for creating MANs. WiMAX is expected to provide high data-rate services over a service area as large as a MAN. It is similar to the Wi-Fi standard; however it supports a far greater range of coverage and WiMAX coverage can cover up to 48 kilometres radius with theoretical data rates between 1.5 Mbps and 75 Mbps per channel (Shuaib, 2009). WiMAX networking has “the potential for use in broadband Internet services, video and audio streaming, and is an alternative to the PSTN for voice services” assert Lansbergen and Koolstra (2002). In this research project, the QoS, CoS and GoS are introduced as key features of WiMAX networks to be tested. QoS refers to the capability of network elements to provide a degree of assurance such that its traffic and service requirements would meet satisfaction. CoS refers to classification of services in different classes and management of each type of traffic with a particular way such as UGS for VoIP and rtPS for Video, amongst others. Whilst GoS refers to the measure of success a subscriber’s connection is granted in accessing a network and transmit data packets without interruption more especially during busiest hours of the network add Lansbergen and Koolstra (2002).

To test the performance of these key features of WiMAX network, twelve scenarios with each being a combination of a certain CoS and scheduler types (FCFS, MBQOS and rtPS) were created. After the aforementioned scenarios were successfully implemented and tested, bandwidth performance evaluation tests and experimentation were undertaken, thereby constituting the required assessment. On the basis of the scenarios implemented the implications they have on the key features of WiMAX network are drawn out based on the results obtained from monitoring the network QoS parameters using **Jperf**, a graphic shell for **Iperf** traffic generator and **Wireshark** for capturing packets in real time. In this chapter, a discussion on the area of research and research problem is conducted and research questions, research objectives and research context are presented. Last but not least, a conclusion of this chapter is drawn to provide a summative introduction and background of this research project.

1.2. Research Site/Location

The research location for this research project is Dwesa, a rural community of the semi-developed Willowvale town located in the coastal part of the Eastern Cape Province, Republic of South Africa. The University of Fort Hare and Rhodes identified this areas in 2006 for research purposes in the ICT4D field and subsequently a comprehensive computing infrastructure was deployed, assert Tarwireyi *et al* (2007). In a developing country, where infrastructure development is yet to be rolled out in many parts of the rural and marginalized areas, the Dwesa community is one of the fortunate rural communities to have a converged WiMAX/Wi-Fi network operating on daily basis. The deployed WiMAX network setup is depicted in Figure 1.1 below:

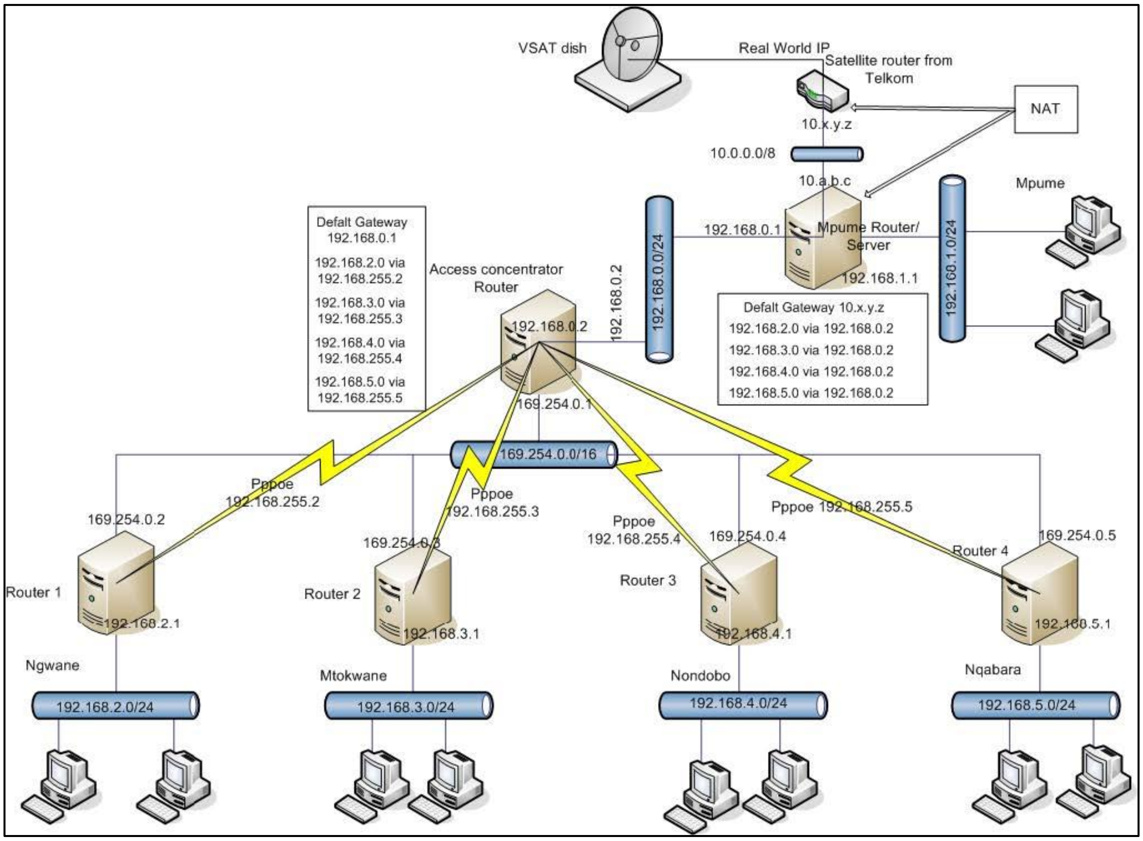


Figure 1.1: "Network Diagram of the Local Loop Access Network in the SLL" (Sieborger and Terzoli, 2010)

This research project exploited this opportunity with the objective of optimizing the network of this community and simulated such a network into a laboratory environment using NS-3 in order to allow testing and experimentation without compromising functioning of the core network.

1.3. Research Background

The IEEE 802.16 standard comprising MAC and PHY layer specifications, unconventional to DSL and cable-modem as traditional wired networks amongst others, was introduced in computer networking to provide Internet services over WMANS (Cao et al, 2005). According to Awal and Boukhatem (n.d) "WiMAX is based on IEEE 802.16e-2005 standard and WiMAX Forum NWG specification." Furthermore, both the IEEE 802.16-2004 and IEEE 802.16e-2005 standards provide pecification for the PHY and MAC of the radio link. However, owing to the fact that interoperability has to provide end-to-end services such as mobility, reliable security, guaranteed QoS, IP connectivity and session management, this is inadequate to construct an interoperable broadband wireless network. This is due to the fact that; "the IEEE 802.11 standard has been the pioneer wireless communication technology and a huge commercial success" (Policy, 2007). Although the introduction of 802.11 standard achieved sound successes in wireless communication, comprehensive improvement and guaranteed QoS had to be instigated, a process that saw to the introduction of a new standard in the form of IEEE 802.16 (Eklund et al, 2002).

In light of the current technology development, IEEE 802.16 is viewed by researchers as a solution to the wired broadband infrastructure more especially for RMAs such as Dwesa and other areas that cannot be easily accessed in urban areas. To this effect, researchers point at the IEEE 802.16's capability to provide guaranteed QoS as the reason behind their optimism for a new solution in IEEE 802.16 standard." This proclamation can be substantiated by the fact that WIMAX handle the increasing network traffic to ensure that mobile Internet access is persistent and achieved at a high-speed in its efforts to cover a considerable large area(s). Additionally, WiMAX, first standardized in 2004, can provide broadband communication over wireless methods for numerous forms of multimedia traffic that include video streaming, VoIP, FTP, amongst others, asserts Kafhali (2014). WiMAX is considered as a technology that is providing a fast local direct connection to the network for residents using a cable or telephone, a solution termed "Last Mile" (Khosroshahy *et al*, 2006). From its inception, the idea of numerous multimedia types transmission was at the centre of the 802.16 standard. This meant that WiMAX should accommodate both the requirements of low-data-rate applications in the form of web surfing and extremely-high-data-rate applications in the form of video or audio streaming and VoIP, and handle extremely demanding traffic over the Internet (Cao et al, 2005). On certain occasions, the 802.16 may need to handle all of these services when the users incur extreme traffic loads in the network.

The UFH together with RU CoE in ICT4D have deployed VSAT to provide Internet connection to rural communities in Dwesa and a WiMAX local loop to extend the network so that it covers up to 50 Kms to link up various nodes in this community. It is through benchmarking from such nodes that the Dwesa WiMAX was successfully simulated into a laboratory environment, called NS-3. WiMAX builds on the experiences and problems of 802.11 wireless networks, commonly known as Wi-Fi. It was developed to solve most of the WLAN shortcomings such as QoS, high-speed data rates and long distance connectivity coverage and security, assert Gray (2007) and Ranga and Terzoli (2009).

1.4. Research Problem

The Dwesa WiMAX network provides broadband communications over wireless connectivity for numerous multimedia traffic including VoIP and FTP amongst others to the community members. The Dwesa community members use schools' computer labs to access the network and generate the aforementioned multimedia traffic during the dedicated timeslots for computer trainings meant for community members and school learners and thus cause traffic congestion on the network. The more the network congestion increases, inversely, the more bandwidth availability decreases leading to prolonged waiting of bandwidth allocation to the connection request by the users. The longer the waiting period takes, the more frustration is experienced and expressed by the trainees/users. This happens at the back of bandwidth availability promised for connection requests once connection is granted.

Using the QoS parameters, namely throughput, average jitter, average delay and packet loss, this research project assessed the extent to which the individual CoS and scheduler types might be central to the QoS and GoS experienced by the trainees/users. Management of bandwidth availability or lack thereof to the connection requests during the training timeslots is also assessed through different combinations of CoSs and scheduler types. Additionally, the long standing intent with the Dwesa WiMAX network has been to monitor the traffic routed to all access points and at the same time evaluate the effectiveness and robustness of this extended converged WiMAX/WiFi network (Ndlovu *et al*, 2009). Last but not least, the Dwesa network lacks a dynamic network optimization technique that is cost-effective to ensure that its users continue to enjoy the guaranteed QoS the network is supposed to provide in spite of the traffic congestions at dedicated timeslots for computer trainings.

1.5. Research Questions

This research project sought to address the underlying WiMAX network impediments that continued to compromise QoS, CoS and GoS through addressing the following research questions:

- What is the impact of the problems caused by the inconsistent availability of bandwidth on the QoS, CoS and GoS on the Dwesa WiMAX network?
- What mechanisms could be employed in optimizing the Dwesa WiMAX network to improve user experience when traffic congestion is increased due to inhabitants' training activities?
- What cost-effective approach, suitable for the Dwesa WiMAX network, could be developed with the aim to optimize the network?
- What are tangible possibilities does the Dwesa network peculiar activities (ICT4D) and conditions have for a prolonged period of time to ensure guaranteed QoS and improved GoS?

1.6. Research Objectives

The objectives of this research project are categorically outlined as follows:

- To assess the impact and extent of the problems caused by the inconsistent availability of bandwidth on the QoS, CoS and GoS on the WiMAX network deployed at Dwesa.
- To provide a network optimization technique to accommodate inconsistent availability of bandwidth caused during traffic congestion as a result of user training activities.
- To recommend the suitable bandwidth cost-effective approach for the Dwesa WiMAX network on the basis of the research findings for the network optimization.

- To assess how the Dwesa WiMAX network peculiar activities (ICT4D activities) and conditions can be addressed in a prolonged period of time.

1.7. Research Context

The assessment of inconsistent availability of bandwidth and resultant degradation with respect to QoS, CoS and GoS on WiMAX network constitutes the primary focal point of this research project. Understanding first the inconsistent availability of bandwidth and their effects on the network performance as informed by various multimedia traffic and amount of traffic generated therefrom remains critical for any interventionist mechanism. Subsequently, the optimization of the network performance to survive inconsistent availability of bandwidth becomes the secondary focal point of this research project. Considering various ways of network optimization which include over-provisioning, IP and Ethernet efforts and mapping protocols, a best suitable technique given the results of the inconsistent availability of bandwidth analysis shall be employed. Furthermore, using network traffic analyzers to collect data from the network played a substantial role towards achieving the results of this research project. The network traffic analyzers deployed listened to the network traffic in transmission and identified unique flows in it. Such flows were differentiated by endpoint IP addresses, TCP/UDP port numbers and input interfaces and were summarized through **Iperf**.

In context, this research project is a work to ensure that the Dwesa WiMAX is resilient to the bandwidth demands that users engender during busy network schedules. Through understanding the extent of inconsistent availability of bandwidth they cause to the network and employing suitable optimization technique(s), the network performance should ensure less or no resultant degradation of GoS and QoS than experienced before. Running experimentation on the laboratory environment, NS-3, results ought to be implemented on the live network, the Dwesa WiMAX network.

1.8. Dissertation Content Overview

This section gives a skeletal overview of the structure and content of the dissertation as a consolidation of this research project

1.8.1. Chapter One: Introduction and Background

This chapter introduces the area of interest or focus of the research project, the dissertation statement, the rationale behind the research project, research questions and the objectives of this research project. Also, this chapter introduces the research site/location and research background with more focus given to the key features of WiMAX that this research project implemented. Last but not least, this chapter concludes with presenting the research context in which this research project is undertaken.

1.8.2. Chapter Two: Related Work

In this chapter, critical analysis of the literature relevant to this research is concluded. This is also the correct section to substantiate on the need for the development of a new literature in the form of this research. Related work discussion on WiMAX networks, interoperability in WiMAX network, CoS, GoS, QoS, bandwidth allocation and request mechanism, CAC, routing protocols, performance monitoring and IEEE 802.16 standard overview is carried out.

1.7.3. Chapter Three: Research Methodology and System Requirements

This chapter provides the methods that are suitable to successfully undertake this research project and yield the expected results. Also, this chapter also provides an insight on how the research methodologies guided the research process from one research phase to the next to address the research questions. Furthermore, the technologies, systems requirements and system architecture that underpin this research project are dealt with under this chapter.

1.7.4. Chapter Four: System Design, Simulation and Testing

This chapter presents the process of designing the laboratory environment, deploying and testing the network analyzer tools on the WiMAX network and assessing productivity and workability of Dwesa WiMAX technologies. Furthermore, this chapter presents and discusses the WiMAX network technical concepts in order to provide guidance to simulation implementation and testing processes. To ensure that the research outcomes are suitable for implementation at Dwesa, this chapter further discusses the alignment of the simulation to the context of Dwesa WiMAX network. This chapter concludes by presenting the outcomes of the monitoring and testing process which will pave a way for results presentation and evaluation.

1.7.5. Chapter Five: Implementation and Results

This chapter basically presents two aspects of the research project in the form of implementation and results obtained. It provides detailed clarity on how the implementation process was carried out and how the WiMAX components were put together in an effort to simulate the Dwesa WiMAX network. It further presents the results generated as a result of several scenarios, with each scenario examining WiMAX network performance of a combination of a certain CoS and WiMAX UL and DL primary scheduler. This chapter concludes by analytically discussing the results obtained from several scenarios from a broader perspective and gives some inferences that were drawn from the outcomes.

1.7.6. Chapter Six: Conclusion and Future Work

This chapter provides a summary of what has been done and achieved through the research project through addressing the research problems and research objections identified. Furthermore, it provides a summary of the contributions i.e. achievements and impacts that the research has made as it relates to the body of knowledge of this research field and further make propositions for future research projects. Finally, this chapter presents the problems encountered in undertaking and concluding this research project and also presents the identified research areas under various fields related to this research as part of the future work.

1.9. Conclusion

This chapter introduced the research project in its entirety through introducing the different components that are the key and cornerstones of this work. Such components include WiMAX networks in general, the Dwesa WiMAX network, QoS, GoS, and CoS, amongst others. In addition, this chapter presented the research site/location and discussed the research background to uncover the underlying research principles behind the undertaking of this research project. Furthermore, the research problems that this research sought to address was presented as informed by the research background provided. Subsequently, the research objectives and research context were explained in detail. Chapter one concluded by presenting an outline of the structure of the entire thesis. In the next chapter, the literature review is presented.

CHAPTER 2

RELATED WORK

2.1. Introduction

In the context of WiMAX; when the BS or SS creates a connection, it associates the connection with a service. According to Belghith and Nuaymi (2008) a service flow provides “unidirectional transport of packets either to UL packets that are transmitted by the SS or to downlink packets that are transmitted by the BS. It is characterized by a set of parameters as a SFID, service class name **(UGS, rtPS,ertPS, nrtPS, or BE),** and QoS parameters (such as maximum sustained traffic rate, minimum reserved traffic rate, and maximum latency).” This chapter therefore presents a holistic review of the related work of this research project, which was conducted to review the existing theories, systems and applications across several research fields that are central and underpins every connection created by the BS or SS as aforementioned. This holistic literature review process was extended beyond these BS or SS connection characteristics to other contributing factors such as CAC, routing protocols and performance monitoring tools. Therefore, the literature review presented in this chapter is presented with respect to the aforementioned several research fields at in-depth level.

Consequently, this section consists of the eight sections: the first section entails an in-depth discussion and analysis of the existing related theories and systems. The second section entails a presentation and discussion of WiMAX networks in general with respect to QoS, CoS and GoS as its key features. Subsequently, a discussion on network bandwidth is presented as a third section. In section four of this chapter, the literature review is conducted with respect to the CAC. Section five conducts a review of the impact of the routing protocols on WiMAX network QoS. Furthermore, section six entails a discussion of the performance monitoring tools and the network analyzers. The second last section presents an overview of the IEEE 802.16 standard. Finally, section eight presents a conclusive discussion on the entire related work discussed under this chapter.

2.2. Related Works

The conducted literature review for this research project revealed that several research projects have been undertaken by researchers affiliated with several institutions of higher learning and industries in a quest to provide improved guaranteed QoS on WiMAX networks. Their attempts to address the phenomenon of guaranteed QoS are characterized by several interventions on bandwidth resource allocation and management mechanisms and CAC algorithms. In this section, such intervening research projects' outcomes are presented and reviewed accordingly.

A discussion on bandwidth allocation to provide guaranteed QoS has been initiated by researchers in the field of WiMAX networks. It has been discussed that; "admission control is a network's QoS mechanism that determines whether a new session (or connection), with given bandwidth and delay requirements, can be established or not" (Wang et al, 2012). For providing QoS, this technique has been practiced on both wireline and wireless networks. In WiMAX networks, "whenever a new session wants to make use of the wireless network, an admission control request is sent by the SS to the BS" (Haider and Harris, 2009). Furthermore on this aspect, this admission control request will be" accepted by the BS if there is enough available bandwidth, QoS guarantees for bandwidth and delay can be met and the QoS of existing connections is not disturbed," assert Aun and Harrs. However, this fundamental principle of providing guaranteed QoS is not specified in the CoSs and CACs by the IEEE 802.16 standard which provides grounds for this research project to be undertaken in a quest to propose other mechanisms to satisfy the QoS requirements.

Another work that was undertaken was a project on the "GoS in End-to-End Service of Quality of Service Broadband Networks" and its aim was to measure accurately the network QoS parameters and objectively assess the GoS in parallel (Gupta et al, 2012). This allowed the researches to "actually quantify the relationship between QoS parameters and GoS for two applications (file transfer and VoIP) and identify their QoS requirements". Upon the conclusion of their research project, the researchers were able to conclude that, using the obtained results, "it is possible to predict an application GoS based on the corresponding measured network QoS parameters and understand the reasons of possible application failure" (Gupta et al, 2012). This affirms the correctness of using various QoS parameters to assess the GoS on WiMAX network and understand the impact of different applications on the overall network performance.

Furthermore, a research study was conducted on different quality parameters that influence the service performance of a WiMAX network using NS-3 as the laboratory environment. Chaudhari and Karule (2014) articulate that the rationale behind NS-3 selections is that NS-3 is dynamically developed on multiple advances and entirely uses C++ as a programming language. It is characterized by various features including simulation core engine, a set of models, example programs, and tests that allow dynamism in development and testing. NS-3 has been designed such that the results obtained through its running in laboratory experimentation can be published. This is due to its capability to validate and test the obtained results using the relevant embedded tools for these processes. Furthermore, Chaudhari and Karule (2014) summarizes the attributes of NS-3 as “strict application and adherence to the IEEE specifications, global use and input, continuous academic, commercial, and open for public inspection of its source code, academic validation through robust academic ethos; and extensive testing.” These summarized development requirements depicts the underlying features put in place in NS-3 as an open-source platform for R & D to optimize the WiMAX network performance. This work confirms the appropriateness of using NS-3 as the laboratory environment that is a suitable for R&D usage, the generated results are validated and tested through the corresponding models put in place and can be published.

On the other hand, Thomas (2011) conducted a research project which used NS-3 simulation model to provide analysis on a new WiMAX OFDMA downlink sub-frame mapping algorithm called merging OCSA, or mOCSA. In addition, Thomas described that this “included a description of the pre-transmission and post-transmission processing that occurs on packets, and a breakdown of significant sections of the model. These sections included the state machines driving each base station or subscriber station on the network, the system used to classify traffic from IP and MAC addresses to WiMAX Connection IDs, the physical layer, the system used to generate and communicate bandwidth grants and the schedules for both DL and UL subframes, various timers and headers that are used, and several remaining classes that did not fit into any of the above categories.” This study, as Thomas placed it, demonstrated that mOCSA produces consistently better maps in terms of average wasted blocks, average unmapped blocks, and average unmapped bursts that form an integral part of packet transmission between BS and SS.

In an effort to improve the QoS on WiMAX networks, several studies have been undertaken with due consideration given to various QoS mechanisms. Amongst those studies, a research to improve the QoS on WiMAX networks by scheduling algorithms was undertaken by Khoei et al (2014). This study was focused on investigating FIFO, Weight Fair Queuing (WFQ), Priority queue (PQ) and Modified Deficit Round Robin (MDRR) as the main scheduling algorithms in a PMP WiMAX setup. The investigation looked at the individual performance of the aforementioned main scheduling algorithms against the different CoSs and applications. The results of the investigation affirmed that, usage of CoS alone cannot guarantee QoS on WiMAX network but for various applications the guaranteed QoS can be improved through proper selection of scheduling algorithms. However, the study did not necessarily exploit the advantage of the three SchedTypes in the form of MBQOS, FCFS and rtPS and thus left another potential mechanism to substantiate its proclamation that CoSs alone cannot provide QoS in WiMAX network and using various scheduling algorithms plays important an role in improving QoS.

Sharma and Chawla (2014) conducted a performance analysis on various QoS parameters that included packet loss, throughput, average delay and average jitter across UGS, rtPS, nrtPS and BE as CoSs. In their performance analysis, they compared each QoS parameter over each CoS when video traffic was transmitted across increasing number of WiMAX network nodes. In this study they only focused on the video traffic and increasing the number of WiMAX network nodes. It is acknowledged in their paper that it would be of great interest to further consider other application data such as VOIP and FTP traffic amongst others and suffice it to say that the consideration of scheduler types, which forms a key part of this research study, could be of great interest as well.

Kaarthick et al. (2009) performed an analysis specifically on how video packets and other application data are distributed in a video conference over WiMAX network across various CoSs. The study also took into consideration the capacity of WiMAX equipment in handling the VoIP flows and studied how throughput, average jitter, average delay and packet loss are affected for various service flows. However, the focus was centred on QoS provisioning with no performance optimization intentions for an environment where available bandwidth is limited.

Another study that analyzed the QoS parameters, namely, throughput, packet loss, average jitter and average delay over various service classes as defined in WiMAX was conducted by Anouari and Haqiq (2012) in a simulation platform. In this study, different VoIP codecs were considered to determine how each CoS fared on QoS parameters with UGS proving to be the best performance parameter for VoIP services, whilst it was equally observed that all CoS in this study performed optimally well so long as the node number is kept at or below six. The outcomes of this research remain to be seen if they would remain the same or change completely if MBQOS, FCFS and rtPS as SchedTypes were to be taken into consideration.

Chauhan et al. (2013) conducted a comparative analysis of the traditional WiMAX scheduler types in the form of FSFC, MBQOS and rtPS against Hybrid MBQOS algorithm which the study proposed for maintaining throughput and to optimize the performance of WiMAX. In this study, the simulation outcomes depicted that the proposed approach, Hybrid MBQOS, tended to reduce the response time between the SS and BS, the packet loss, average delay and further improved the network's throughput compared to MBQOS, FCFS and rtPS. However, the study did not reveal under which service classes these outcomes were drawn. It remains to be seen if the outcomes would remain the same across all defined WiMAX CoSs, UGS, rtPS, nrtPS and BE.

The related works reveal that several research studies around the concepts of QoS parameters, services classes, bandwidth usage and scheduling types as defined in WiMAX have already been conducted. In so doing, the objectives were to improve WiMAX throughput and reduce packet loss, average delay and jitter. However, these studies fall short in conducting a comprehensive study which would reveal the impact of all the underlying WiMAX features in achieving improved throughput and optimization of the WiMAX network. Thus it established the underlying research interest of this research work to consider how each combination of the service class and scheduling type contributes to the required QoS and GoS in an environment where bandwidth costs should be kept at as minimal as possible.

2.3. WiMAX Networks

The ideology of QoS was and has always been at the centre of WiMAX since its inception such that various mechanisms and techniques were put in place. These mechanisms and techniques include the various CoSs, specifically put in place to ensure that quality is not compromised in various applications. Although this form of technique is best implemented in point-to-multipoint (PMP) mode, the ideology of QoS was never discarded for the Mesh mode. Accordingly, a message-to-message mechanism was introduced for the Mesh mode, articulate (Carvalho et al, 2013). In a Mesh mode setup, many WiMAX BS nodes are deployed in order to allow sustainable interconnection of multiple mobile clients which results to a network that grants mobile clients' communication over a wide coverage area. In this mode, there is no necessity to deploy an intermediate node to facilitate clients' communication as communication can simply take place between any pair of nodes that need to communicate at a given time. This communication or clients' broadband access remains true in a Mesh mode regardless of the wireless support settings that are single-hop or multi-hop in form as said by Kas *et al.* (2010). In addition to this, Carvalho *et al.* (2013) assert that, in comparison to the PMP mode, in order for the WiMAX MAC layer to distribute SDUs and MAC PDUs of diverse QoS requirements, a scheduling service is used. Furthermore, Carvalho adds that the role of a scheduling service is to determine, in a peculiar manner, the necessary mechanism the BSs will apply in allocating UL and DL transmission occasions for the PDUs.

Furthermore, from its inception as a mobile wireless technology, WiMAX made an impression to provide both guaranteed QoS and high throughput as placed by Ahmad and Habibi (2010). Additionally to this impression and as part of quality, the unavoidable expectation from clients to experience whenever a connection attempt is made is the GoS. According to Ahmad and Habibi (2010), the GoS can be understood as the probability that, given varying application bandwidth demands certain QoS requirements will be guaranteed from the beginning to the end of a connection session. Following this background, it manifests as a matter of critical importance for this research project to conduct an in-depth discussion and dedicate some considerable amount of time on these three key features of WiMAX which are, namely, CoS, GoS and QoS. This creates conducive grounds for subsequent investigation and analysis of these three WiMAX key features going forward with this research project. A closer look at these WiMAX key features begins with the CoS followed by GoS and QoS accordingly. These three WiMAX key features are roughly discussed in the following sections of this chapter.

2.3.1. Interoperability in a WiMAX Network

According to Paolini *et al* (2007), in addition to the commitment to offer both guaranteed QoS and high throughput, ensuring full interoperability by forms of certification and vendors' ad-hoc testing stands as a strong commitment within the WiMAX industry. Therefore, it is essential for network operatives to fully comprehend the process of establishing interoperability for this standard in order to realize how to best integrate different products, solutions and applications as developed by a diverse range of manufacturers to meet the interoperability commitment. However, it is said that understanding and complying alone does not guarantee the interoperability of the IEEE 802.16 standard as certain products may comply with the standard, yet fail to interoperate.

Therefore, to realize the interoperability commitment the requirement for extensive tests should be viewed as fundamental in order to establish interoperability at different WiMAX network levels

Following compliance with the testing requirement, interoperability is further verified by the WiMAX Forum Certification programme focusing on the PHY and MAC layers. These layers are crucial for "essential over-the-air transmission, the management of connections, and security and mobility management, including handoffs, power control and QoS," as articulated by Paolini et al (2007). This process of certification can only be done for the BS and SSs where interoperability is established between for WiMAX network operation. Beyond interoperability, certification is done to ensure that the certified nodes, the BS and SSs, can easily communicate with any other WiMAX nodes of similar frequency and channel width.

2.3.2. Class of Service

Scheduling services are a means of implementation of various mechanisms for data handling and transmission on a connection as supported by the MAC scheduler. Placed at centre of this service implementation is the necessary traffic management system or mechanism. To address this, the IEEE 802.16 standard divides all possible services to be transmitted into five different classes as informed by a set of QoS requirements to be met for each service. Firstly, the five service flows as defined by (Policy, 2005) are presented in Table 1.1 below to determine the level of service required associated with each CoS. The process of determining the appropriate CoS for a given service requires that a connection be assessed by the routers and switched deployed in the network, considering several factors. According to Gupta *et al.* (2012), when considering these factors, routers and switches look at the service type, the source and destination identification and the type of priority the service entails and thus determine a proper CoS need. Table one below presents the four classes of services and their QoS requirements outlined below:

Table 1.1: WiMAX CoS and QoS requirements

QoS in WiMAX	**Applications**	**QoS Requirements**
UGS	T1/E1 transport VoIP	• Maximum Sustained Rate • Maximum Latency

		Tolerance • Traffic Priority
rtPS	Video Conferencing, Audio Streaming, Telemedicine, e-Learning	• Minimum Reserved Rate • Maximum Sustained Rate • Traffic Priority • Maximum Latency Tolerance
nrtPS	FTP, Document Sharing	• Minimum Reserved Rate • Maximum Sustained Rate • Traffic Priority
BE	e-Mail, Web Browsing	• Maximum Sustained Rate • Traffic Priority

The **UGS** is developed and implemented to service the requirements of the real-time CBR applications, i.e. the T1/E1 and VoIP applications. Furthermore, the request/grant process generates substantial levels of overhead and latency and unsolicited data grants are allocated for the purposes of eliminating their rise. Lastly, as maximum sustained traffic rate is normally declared during connection establishment stage and fixed bandwidth grants are assigned by the BS in each frame to ensure real-time service requirements are met, as explained by Carvalho *et al.* (2013).

The **rtPS** is developed and implemented to provide the necessary support to the real-time services that periodically generate variable-size data packets, such as MPEG video. Using **rtPS**, the SS requests bandwidth due to unicast polling opportunities provided by the BS. To ensure that latency requirements are met during transmission the unicast polling opportunities should be as frequent as possible (Toy, 2015).

The **nrtPS** is designed for delay tolerant applications traversing in the channel. Indifferently from the **rtPS,** the **nrtPS** also dedicates periodic slots for the bandwidth request opportunity, but the difference lies on the much longer periods in **nrtPS**. In this class of service, unicast polling opportunities are allowable but the difference lies at the waiting period of a few seconds between two such opportunities, a period considerable enormous compared to **rtPS,** asserts (Yin and Pujolle, 2008).

Finally, the **BE** is developed and implemented to support data streams of no mandatory minimum service level and flexible enough to be handled as and when a space is available. When the BS intends to send any data streams, both opportunities of contention request and unicast polling can be used by the SSs. According to IEEE 802.16 (2004) the mandatory QoS service parameters for this CoS include “maximum sustained traffic rate, traffic priority, and request/transmission policy” This implies that BE is designed to provide the least significant amount of QoS support as compared to the rest of other service classes.

2.3.2. Grade of Service

GoS can be understood as the probability that, given varying application bandwidth demands, certain QoS requirements will be guaranteed from the beginning to the end of a connection session. This understanding is further explained by Inayatullah *et al.* (2006) as “a benchmark used to define the desired performance of a particular trunked system by specifying that the user is given access to a channel if a specific number of channels are available in the system.” It is actually the measure of the success rate that a client will have in accessing a trunk system during the busiest hour combined with the clients’ success rate that after connection there will be no connection interruption for the whole transmission duration. It is always better studied during the busiest hours of the trunk system when the traffic intensity is extreme as opposed to a scenario where the network is quite and all connections are guaranteed uninterrupted transmission. Therefore, GoS provides a unique and different perspective from a scenario of incoming versus outgoing calls, and can be viewed from connection requests in a specific source to a specific destination as explained by

Gupta *et al.* (2012). (Inayatullah *et al*. 2006) elaborate by looking at the GoS as that quality aspect that a client always looks for as and when making a network connection in both loss system and delayed call or connection system. Thus Inayatullah *et al*. conceive this concept of a GoS as the likelihood that a connection experiences obstruction, or the likelihood of a connection being delayed longer than the standard queuing/waiting period.

2.3.4. Quality of Service

Under this subject, the in-depth study conducted produced the following understanding about the QoS on WiMAX. On WiMAX, QoS can be understood to refer to the sustained capability of the network functioning that, at the least, meets the standard minimum requirements to satisfy its traffic and services. The WiMAX network functioning has been planned and has to continue to maintain its QoS requirements of an access network designed to meet the clients' needs of multimedia applications as and when they are transmitted..

According to Wang *et al* (2012), in order for WiMAX to continue to live by its commitment to provide guaranteed QoS for data transmissions, no connection must be established in the network without the BS applying a CAC scheme to determine whether or not such a connection should be established with respect to the available network resources. To this effect, the network dictates of the CAC scheme prove to be critical for both the provisioning of guaranteed QoS for admitted connections and how the network resources are distributed and utilized across the board. Interestingly, and in spite of the critical role that a CAC scheme plays in ensuring efficiency in the overall network performance, the specification to CAC scheme implementation mechanism is not defined by the IEEE 802.16e standards and is open for contextualization.

There are several parameters or metrics put in place to measure QoS on WiMAX and ranges from technical parameters to non-technical parameters. Technical parameters include delay, jitter, packet delivery ration, packet loss ratio whereas non-technical include expense, availability, security and perceived quality (Mehta and Gupta, 2012).

- **Delay** - Delay or latency could be defined as the time traversed by the packets from one end to another. According to Ruhani *et al* (2011) delay can be as a result of various sources which include "propagation delay, source processing delay, network delay and destination processing delay." Ruhani *et al*. have calculated end-to-end delay observing the elapsed time taken during modulation of the signal and the time traversed by the packets from one end to another. To this effect, consideration is also given to the effect of noise on the packet as increased noise may negatively influence the arrival time of the packet. Therefore, end-to-end delay could be determined from the time it takes packets to be delivered to their desired destination, the actual difference of packet arrival and packet start time. Equation 1 below depicts the formula to calculate the average end to end delay:

$$Delay = \sum Packet\ Arrival_i - \ Packet\ Start_i \qquad (1)$$

- **Jitter** - Vikram and Gupta (2012) describe Jitter "as the variation in delay or packet delay variation which its value of jitter is calculated from the end-to-end delay". This means, the less Jitter occurrence the better the network performance and high chances of provisioning the guaranteed QoS and thus place the importance of measuring Jitter critical. Furthermore, to measure network stability and consistency, Jitter can be used as due to its ability to measure the variation in the time between packets arriving. Equation 2 presents the formula to calculate jitter.

$$Jitter = \frac{\sum_{i=0}^{n} square\left(Delay_i - \ \overline{Delay}\right)}{N} \qquad (2)$$

Where "n" is the total number of packets

- **PLR**-Packet loss can be caused by various problems which include the bit errors in wireless networks that are faulty, insufficient buffers as a result of channel overload during congestion times or just caused as a result of noise as articulated by (Talwalkar and Ilyas, 2008). Additionally, Vikram and Gupta (2012) discussed that PLR has a potential to negatively affect the perceived QoS hence the "value of packet loss should be kept at minimum level" as expounded in ITU standards so as to meet the WiMAX commitment to provisioning of high throughput and guaranteed QoS. Equation 3 below presents a formula to calculate Packet Loss Ratio:

$$PLR = \frac{\sum_i Packets\ Loss}{\sum_i Packets\ Sent}\ X\ 100 \qquad (3)$$

- **PDR** - Packet delivery ratio relates to the total number of packets that was successfully transmitted to the desired end-node. Equation 4 below presents the formula to calculate the PDR:

$$PDR = \frac{\sum_i Packets\ Delivered}{\sum_i Packets\ Sent}\ X\ 100 \qquad (4)$$

- **Throughput-** Vikram and Gupta (2012) defines Throughput as a “measure of number of packets successfully delivered in a network and it is measured in terms of packets or second.” Ideally, in order for any network performance to be regarded as satisfactory, the value of throughput should be kept high or else its QoS would be deemed poor or substandard. Equation 5 presents the formula to calculate throughput:

$$Th = \frac{\sum_i Packets\ Delivered}{\sum_i Packet\ Arrival - Packet\ Start\ time_i} \qquad (5)$$

- **Availability-**relates to the duration, short or long, at which the network connectivity is remains possible and uninterrupted between an ingress point (entering point) and a specified egress point (exiting point) as described by Islam *et al* (2011).
- **Security-** for this parameter Khatkar *et al* (2013) contend that “various factors need to be considered including the vulnerability of the network, the threat of the attack, the value of the data to be secured and the costs involved” in order to keep the perceived risks and costs at an equilibrium state.
- **Perceived Quality**- The perceived QoS can be defined and determined through an inclusive measure of various network performance parameters such as the received throughput, packet loss, average end-to-end delay and average jitter as were considered by Afzali *et al* (2010).

These are the metrics that the research project has monitored in order to determine the repulsive effects of inconsistent availability of bandwidth on WiMAX QoS as influenced by the CoS, GoS and CAC.

2.4. Bandwidth Allocation and Request Mechanisms

According to Prasad and Velez (2010) SSs uses various methods to notify the BS about their intentions to send data on the UL whenever a need arises. Common amongst these methods is the sending of a BRH which is provided by the BS after issuing an UL channel access grant to the SS intending to send data. Additionally, the process of requesting connection and subsequent allocation of bandwidth is characterized by efficiency, low- latency and flexibility. As requests are made per connection, the BS uses the fairness algorithms of the UL scheduler to ensure all requests are treated fairly, contend Prasad and Velez (2010). Although requests are made per connection, grants are not sent directly to the connections but to the SS. Furthermore, Prasad and Velez assert that although the BS allocates bandwidth to the SS as per their requests, the SS do not send explicit acknowledgments back to the BS to indicate whether a bandwidth request message is successfully transmitted or distorted or what the amount of bandwidth a SS was granted. The acknowledgments are sent as and when a SS a determined that its BR was corrupted and therefore a grant was not received, then initiate a process called contention resolution. The contention resolution process will continue to unfold up until such a time a grant was finally received within the timeout and the SS will begin to use the allocated bandwidth for UL transmission of data packets. In an event additional bandwidth was needed, the SS will piggyback an additional allocation of bandwidth.

Observing the following grant from the BS allows the SS to determine how much bandwidth is awarded to it. BS uses different scheduling algorithms for bandwidth allocation and as such granting times varies, a grant may be given at any given time. Although SS must initiate a contention resolution process upon realizing that its BR was corrupted, this process is prohibited if the SS uses the UGS service and in this case there are no explicit bandwidth requests issued by SS. Instead, for UGS flows the BS must periodically provide fixed size data. For the **rtPS** and **nrtPS** flows, this contention resolution process is polled through the unicast request polling. However, during traffic congestion, a few request polling opportunities are received by the **nrtPS** flows and only then **nrtPS** flows are allowed to initiate the contention resolution process whereas the **rtPS** flows are polled with or without the traffic congestion until the **rtPS** delay requirements are met, argue Prasad and Velez (2010).

2.4.1. Bandwidth Requests

According to Farooq and Turletti (2009), BS allocates bandwidth to SSs on the basis of the connection requests it has received as individual connection requests from various SSs. The process is started by sending a bandwidth request packet either in a special transmission opportunity allocated in UL sub-frame or in an UL grant allocated to the BS. In the BS, at the wake of many bandwidth requests, the BS uses its UL scheduler to decide which SS to allocate bandwidth next. When the UL scheduler has decided, the BS responds to the requests through sending grants in subsequent frames to the SSs not to connections per se although it had received the requests per connection. As soon as the SSs receive bandwidth as requested per connections, they then consider all the connection requests they have and determine, based on their dictates of their schedulers, which connection will transmit first than the others. In a similar manner in the BS, the scheduler gives priority to the packets that will be transmitted in DL sub-frame. It is also noteworthy to mention that standards leave the responsibility of deciding the scheduling algorithms for the schedulers solemnly to the manufacturers, concludes Farooq and Turletti (2009).

According to Ma and Denko (2008), the bandwidth request process has to follow one of the two defined types of bandwidth request, namely, the incremental and aggregate. The incremental bandwidth request maintains that the connection still needs the initial amount of bandwidth requested although its current bandwidth need might have changed and this type of request is common in piggyback bandwidth requests. On the other hand, the aggregate bandwidth request entails the total bandwidth amount each connection will need to successfully transmit packets. These aggregate bandwidth requests are periodically sent not every time there is a connection request, conclude Ma and Denko (2008).

2.4.2. Polling

Polling is another mechanism the BS uses for bandwidth allocation to the SS in addition to the piggybacking request on transmitting data unit. Thontadharya *et al.* (2011) describe polling as a process of periodic allocation of the BS's part of the UL channel capacity through issuing a “grant” or “transmission opportunity” in the UL map to all the SS intending to transmit packets. In this process CIDs are very crucial as they help identify whether polling done was individual (unicast) or group based (multicast) or broadcast. CIDs achieve this through specifying the polling nature done in the uplink map transmit opportunity information element, asserts Cicconetti (2010). In an event the polling nature is multicast or broadcast then an appropriate method of the contention bandwidth request is specified for the purposes of collecting bandwidth request responses. In the case of a Unicast poll, the bandwidth request responses are collected through a single CID associated with a particular SS from which the requests come. Contrary to the bandwidth allocation procedure where requests are made per connection but grants are made per SS not per connection; polling is done on SS basis and bandwidth is allocated accordingly because CIDs are always used for bandwidth requests. As for polling broadcasting, the BS broadcasts to all UL connections and polls are used and accessed randomly. . According to Thontadharya *et al.* (2011) the bandwidth request, transported by a BRPDU and circulated by the BS in response to broadcast polling are called contention bandwidth request. The broadcast polls are common in and only used by the **BE** and **nrtPS** scheduling services, contends Cicconetti (2010).

2.4.3. Bandwidth Grants

According to Ma and Denko (2008) UL channel access by SS's grant transmission opportunities is controlled by the BS. Contrary to the bandwidth request procedure, where requests are made per individual connections, bandwidth grants are made on a per-SS basis. This is a procedure of bandwidth granting is a true example of an aggregated grant method to the SS. In this case, the SS may unavoidable receive lesser bandwidth than it actually requires for its connection requests. In such events, the SS may choose one of the two options, either to perform a backoff and request again or discard the data packets. Prior the decision to grant bandwidth is made, the BS scheduler considers two priorities, namely, the QoS parameters and the status of the current packet queues, adds Ma and Denko (2008). Thontadharya *et al* (2011) assert that when bandwidth is finally granted, the SS local scheduler now has a duty to decide which connections are granted bandwidth first.

2.5. Connection Admission Control

CAC remains as one of the fundamental mechanisms to provide guaranteed QoS and GoS on WiMAX networks through its ability to provide hard resource reservations during network traffic congestion. In Liu *et al* (2008), it is contended that the introduction of a CAC scheme for diverse levels of QoS is central to the efficient resource management for both existing and new flows. Furthermore, it improves the general performance of the network and minimizes the unviability likelihood of the continuing flows whilst at the same time ensuring that the GoS is guaranteed to the essential applications. CAC achieves this through its dynamic admission criteria and adaptive QoS strategy.

In WiMAX networks, CAC schemes are very crucial in that they are used by BSs to determine, on the basis of the available network resources, whether or not a new connection should be established as per the received connection request, explains Wang *et al* (2012). The connection-oriented MAC used by WiMAX requires all SSs to send a connection request to the BS prior the transmission or receival of any data. It is then upon receival of the connection request that the BS uses its CAC scheme to determine, on the basis of the available network resources, that if support for connection can be granted without compromising the required guaranteed QoS for other ongoing transmissions. According to Wang *et al* (2012), common and central to all CAC scheme is the fundamental operational principle of high connection blocking rate rather than high dropping rate, a principle to maintain the commitment of provisioning of guaranteed QoS and GoS. This has led to many researchers working on methods to design and improve the CAC schemes for this standard as observed by Ahmad *et al* (2010). On the basis of the literature review conducted with regards to the CAC schemes, it is safe to state that CAC schemes vary from network to network depending on certain defined policy requirements. It is safe to state this as Kannisto *et al* (2007) contend that the scheduling policy algorithms to allocate slots called scheduling policy remain undefined for this standard but rather open for network operators to develop them as deemed suitable for their networks.

On a critical point of view, the fact that scheduling policy remain undefined for this standard opens a loophole for depleted provision of guaranteed QoS. This possess a potential for any defined CAC scheme to be deployed to the network irrespective of whether or not it provides QoS to the best capability of the bandwidth available. This becomes the basis for this research project to explore this loophole to the core on Dwesa WiMAX network.

2.6. Routing Protocols

Afzali *et al* (2010) defines routing as "a service in which the router evaluates the possible paths to transmit packets to their destination, and determines the best route this packet should follow." And "protocols are the set of rules through which two or more devices (mobile nodes, computers or electronic devices) can communicate to each other" as Ali and Ali (2010) define them. The router achieves this mandate through implementing the network layer protocol(s) called routing protocol(s). According to Graziani and Johnson (2008), central to the mandate of the routing protocol is the best paths determination for each route to include in the routing table. Currently, various routing protocols are designed with several and unique differences and similarities between them which set their advantages and disadvantages when implemented. In the large pool of routing protocols, namely, DSDV, AODV, DSR and TORA, no single protocol can be deemed as the best of the rest. According to Carvalho *et al* (2013) this claim is made at the back of the understanding that routing protocols are characterized by numerous peculiarities and as such no protocol can be considered to be optimal across all networks. Furthermore, the characteristics of each routing protocols responds to a particular set of requirements required by a particular application. Routing takes two forms, namely, the static routing and dynamic routing and are considered in the following two sub-sections.

2.6.2. Dynamic Routing

The Dwesa WiMAX network is configured using static routing and is configured in such a way that each SS is accessed through a single route and communicates back through that very same route. Therefore, it is safe to conclude that the Dwesa WiMAX network is what is referred to as a Stub network as it is depicted in Figure 1.1 in chapter 1. In Graziani and Johnson (2008), a stub network is defined as a network that is accessed through a single route. This therefore implies that the routing protocols that would influence the QoS and GoS in the network could not be considered in this investigation.

2.7. Performance Monitoring Tools

The Dwesa WiMAX Network core router is a FreeBSD machine which is installed in the BS OS located in Ngwane Senior Secondary School. According to Sminorff *et al* (2012), the FreeBSD machine has an implementation of Cisco’s NetFlow called **ng_netflow**. On a router that is running FreeBSD, Cisco's NetFlow export protocol can be implemented by the **ng_netflow** in order to filter the network flow. The process of filtering by the **ng_netflow** node involves listening for incoming traffic and identifies unique flows in it. In order to identify different flows, each flow is required to have **endpointIP** addresses, TCP/UDP port numbers, ToS and input interface. In NetFlow version 5/9 UDP datagrams, should a flow expire in the process it gets exported out of the node. According to (2016, 1995) the reasons for flow expiration can be one of the following:

- RST or FIN TCP segment.
- **Active timeout:** Because the default is 1800 seconds (30 minutes) no flow is allowed to live beyond default period.
- **Inactive timeout:** Because the default is 15 seconds, no flow is allowed to stay inactive beyond the default period.

Given the software deployment in the Dwesa WiMAX network, another network traffic analyser that can be installed on a FreeBSD machine and which the research project utilized as well is the **Softflowd**. According to (Google code archive - long-term storage for Google code project hosting, no date), “**Softflowd** is flow-based network traffic analyser capable of Cisco NetFlow data export.” In Softflowd traffic is tracked either by listening on a network interface or reading a packet capture file. Softflowd is flexible enough to allow the tracked traffic to be collected via NetFlow to a n external software application or be summarized internally.

2.8. Overview of IEEE 802.16 Standard

According to Farooq and Turletti (2009) “the IEEE 802.16 standard defines MAC and PHY layers specifications for a Broadband Wireless Access (BWA) network.” They explain that the “physical medium is divided into frames of fixed length and each frame is further divided into DL and UL sub-frames for the DL and UL traffic.” They add that the frame structure is dependent on the on the essential PHY layer. Furthermore, because of different operational environments WiMAX defines multiple PHY layers accordingly. Based on single carrier modulation, for the 10-66 GHz band, the wireless MAN-SC PHY is defined. To accommodate a direct LOS propagation using frequencies below 11 GHz, the following three methods are provided: “wireless MAN-SCa using single carrier modulation, Wireless MAN-OFDM using OFDM, and Wireless MAN-OFDMA using Orthogonal Frequency Division Multiple Access" as discussed by Farooq and Turletti (2009). Lastly, they present that WiMAX supports two duplexing techniques: "FDD, where DL and UL sub-frames take place at the same time but on different frequencies, and TDD, where DL and UL sub-frames take place at different times and usually share the same frequency."

Cicconetti *et al* (2006) articulate that the MAC layer of the standard was designed to service stations that are far apart from each with high data rates, where the SSs are just required to listen to the BS rather than listening to other SSs, as it is the case in IEEE 802.11. They add that transmissions of the corresponding SSs are scheduled in advance by the BS and the IEEE 802.11 MAC works on reservation and contention-free basis. Furthermore, the SSs' connection struggling can only be expect during the first channel connection attempt where connection is still evaluated by the CAC scheme. WiMAX BS is designed to serve numerous SSs whilst at the same time providing the guaranteed QoS in the connection level for both UL and DL traversing packets due to its reservation-based resource allocation method. Additionally, they argue that WLAN based on IEEE 802.11 standard tend to struggle every time prior transmitting and with reduced efficiency owing to the increased number of stations joining the network. Therefore, they argue "that in such a contention-based resource reservation scheme, QoS could hardly be considered in the early standard until the advent of 802.11e." However, the majority of the WLAN networks being deployed currently no longer practices any QoS mechanism, they conclude.

According to Xiaojing (2007) sharing of radio channel resources among numerous admissions of diverse user base is central to the MAC protocol. She explains that in WiMAX, the MAC layer is divided into three Sub-layers, namely, the Service-Specific Convergence, Common-Part Sub-layer, and Security Sub-layer. She presents the details of these three sub-layers by saying, the classification and association of external SDU with a proper MAC service flow identifier and connection identifier is the principal mission of the Service-Specific Convergence Sub-layer. The different traffic types are supported by the flexibility and efficiency of the MAC layer protocol. As it relates to the Common-Part Sub-layer, the layer operates independently of the transport mechanism, which is the kernel bearing all the MAC characteristics. She adds that it fragments and segments each MAC SDU into MAC PDUs, system access, bandwidth allocation, connection maintenance, QoS control, and scheduling transmission, etc. Lastly, she concludes by saying the

MAC also comprises of a distinct security Sub-layer for handling of authentication, secure key exchange, and encryption. Figure 2.1 below presents the IEEE 806.16 protocol layering as discussed above:

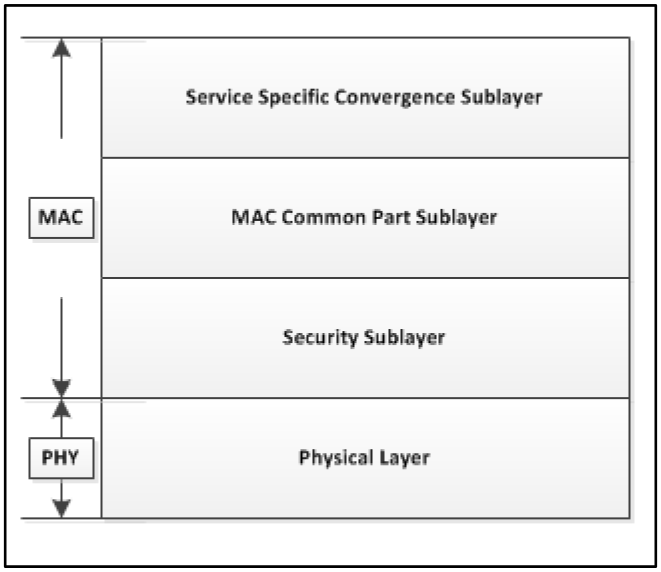


Figure 2.1: IEEE 806.16 Protocol layering

2.9. Conclusion

This chapter presented a holistic review of the related work of this research project, which was conducted to review the existing theories, systems and applications across several research fields that are central and underpins every connection created by the BS or SS. This process was followed by an in-depth discussion on various contributing factors to WiMAX networks such as bandwidth allocation and request mechanisms, CAC, routing protocols, and PMTs. This chapter concludes by presenting a bird’s-eye view of the IEEE 802.16 standard to provide a clear view of how WiMAX network operates. In the next chapter a detailed description of the research methods, network technologies, system requirements and system architecture that guided this research project are presented.

CHAPTER 3:

RESEARCH METHODOLOGY AND SYSTEM REQUIREMENTS

3.1. Introduction

This chapter discusses the methods and procedures which this research project employed in order to obtain its outcomes as discussed in successive chapters. It further provides a detailed account of the design technologies that were utilized during the assessment study and the system requirements and functionalities.

3.2. Research Methodologies

This section relates to the methods and procedures that were employed to conduct the assessment of QoS, CoS and GoS on the simulated Dwesa WiMAX network through performance monitoring of bandwidth. The experimental and computer simulated methodologies were adopted and followed until the final conclusion of this research work. The experimental approach was used to study and analyze the network performance and behaviour under different conditions and parameters. In order to emulate and build a WiMAX network for laboratory environment implementation, the computer simulation method was used. Furthermore, it depicts the processes of research construction, data collection and data analysis procedure.

3.2.1. Research Construction

The understanding of a literature review as a systematic search to discovering the already known facts about the intended research topic as Fraenkel *et al* (1993) explain has necessitated that the literature review process be taken as a primary step for the undertaking of this research project. An in-depth literature review on key concepts of this research project which include QoS, CoS and GoS on WiMAX networks, performance monitoring and network bandwidth was conducted to further expand the prior-understanding of these key concepts pertaining to their meaning in the broader networking environment.

Moreover, this effort was taken precisely to ensure that the scientific findings of this research project do not duplicate the already established facts in the research field but rather unearth new original ideas that will expatiate the existing body of knowledge. Most significantly, this process of literature review was necessarily undertaken to justify the relevance of this research project in the midst of the related existing knowledge in the research field. This process of literature review was fundamental for it has informed the design and methodology for this research project.

3.2.2. Data Collection

The nature of data that needed to be collected for this research project comprised technical data and as such this section relates to the collection of data packets. To address this concern, NS-3 model was used as the laboratory environment for system design and testing, a scenario that brought about the generation and collection of network packets as the technical data. **Iperf** and **Wireshark** were extensively used for technical data collection. This process was informed of the following two methods:

- NS-3 should be used to generate network traffic amongst the Base Station Node (bsNode) and Subscriber Stations Nodes (ssNodes). This process should see the full exploitation of Iperf as the network traffic generator.
- Packets' transmission should be captured and visualized for network performance analysis and Wireshark should be used to this effect.

To test QoS, CoS and GoS through performance evaluation of bandwidth from the data collected from the simulations conducted, the following methods were employed:

- **Observations and Capturing**- observe in real-time using **Jperf** how the simulations perform from time to time on the provisioning of QoS and GoS under different network circumstances and capture images of distinctive network performances for presentation.
- **Statistical comparisons**- compare the statistics with respect to the bandwidth performance on QoS, CoS and GoS as value changes were made to certain variables such as ***MaxBytes, PacketSize***, ***Remote, Protocol*** and ***SchedTypes***.
- **Analysis of variables values**- analyze the impact made by value changes to certain variables to the overall network performance.
- **Mathematical Summarization of Data**- using means and frequencies to summarize the data that was collected, observed, compared and analyzed.

3.2.3. Procedure for Data Analysis

Through exploiting **iperf** and **wireshark** the obtained data was subsequently analyzed against the aforementioned research. To guide the assessment and optimization, the following questions were posed with respect to the obtained data:

- What is the bandwidth allocation per channel during low network traffic congestions and how does it impact on the guaranteed QoS provided using throughput as a metric?
- What is the bandwidth allocation per channel during high network traffic congestions and how does it impact on the guaranteed QoS provided using throughput as a metric?
- What amount of network traffic can cause jitter, delay, PLR and PDR on the network?
- What leads to a compromise of guaranteed QoS when there is CAC and bandwidth?
- What amount of bandwidth is required at the Service Specific Convergence Sublayer to ensure no CoS compromises the guaranteed QoS? and
- What amount of bandwidth is required to guarantee GoS during network traffic congestions?

To test and implement the proposed optimization techniques the aforementioned questions were posed again under similar network conditions and thus the analysis yielded the results presented in chapter 5 of this research project.

3.3. Network Technologies

In this section a reference to the underlying technologies that were used during the assessment such as **NS-3**, **iperf** and **Wireshark** is made:

3.3.1. NS-3

The **NS-3** simulator is implemented due to its active development on multiple fronts and entire coding in C++. **NS-3** comprises of key features such as simulation core engine, models sets, sample programs, and practical tests. The practical tests conducted in the **NS-3** testing environment can be validated with various models and testing tools and verified with validation results for publication as presented in Leclerc and Crosby (2010). NS3 is an open-source software for R & D purpose and its features for development efforts as outlined by Chaudhari and Karule (2014).

3.3.2. Wireshark

Wireshark – Is very a famous and powerful tool for traffic capture and tracing. It shows exhaustive information about packets, and it is the best tool for collecting experimental data, or for testing applications which work via the network. One can see which information is transmitted in any packet, source and destination addresses, amongst other attributes as explained by Bakharev (2010).

3.3.3. Iperf and Jperf

Bakharev (2010) alludes that "one more important tool for network performance analysis is **jperf**. It is a graphic shell for **iperf** traffic generator. Generally, **iperf** have command line interface and sometimes it is not very useful. **Jperf** does not provide new opportunities for **iperf** except one. **Jperf** has an analyzer of **iperf** output, which can plot graphs. It sounds quite usual, but this tool has some feature. It plots graphs in real time, so one can visually observe how bandwidth changed in real-time, during transmission."

3.4. System Requirements

Under this section, a discussion on the system requirements of the network technologies deliberated above is conducted. The system requirements are discussed precisely to bring about a clear understanding about the compatibilities of the design technologies of the Dwesa network. This is achieved through a presentation of the network environment, the system requirements of the design technologies and a discussion on whether or not they can be met at Dwesa. The process begins with NS-3 as the platform for simulating the Dwesa WiMAX network, followed by **iperf** as the network generator and **Wireshark** as the network traffic analyser for this research project.

3.4.1. NS-3 Model

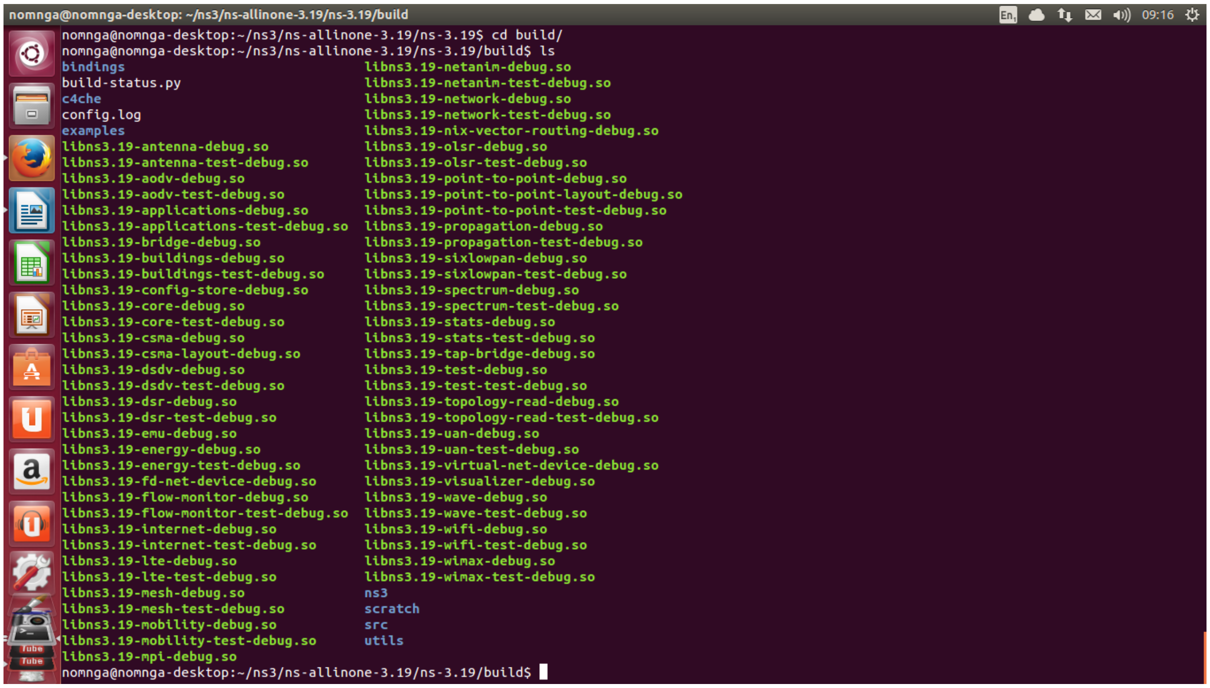
According to (*NS-3: WiMAX models*, n.d) the model is developed on GNU/Linux platforms and comes with minimum requirements to implement basic simulations which include a GCC or clang compiler and Python interpreter. Amongst other operating systems, NS-3 is supported on Linux x86 and x86_64: GCC versions 4.2 to 4.8. Additionally, (*NS-3: WiMAX models*, n.d) asserts that support means "the project tries to support most or all of the built options on these platforms unless there is a good reason to exclude the options, and at least the debug build will compile." Furthermore, it is presented on (*NS-3: WiMAX models*, n.d) that the **NS-3** support for optional features, with some options such as threading primitives and real time simulator enabled and others such as Python bindings, **NS-3** click integration, tap bridge, PyViz visualizer, amongst others, are not enabled. However, different features are supported in different platforms and Table 3.1 below reflects this difference:

Table 3.1: Option Status (NS-3: WiMAX models, n.d)

Option	**Linux**	**FreeBSD**	**Mac OS X**
Optimized Build	Y	Y	Y
Python bindings	Y	Y	Y
Threading	Y	Y	Y
Real-time simulator	Y	Y	N
Emulated Net Device	Y	N	N
Tap Bridge	Y	N	N
Network Simulation cradle	Y	?	N
Static builds	Y	Y	Y

Where Key: **Y** = supported; **N** = not supported; **?** = unknown; **dev** = support in ns-3-dev

In line with the background given above under this section of **NS-3** model, different options require additional support and different NS-3 options, for Debian/Ubuntu system, require the following list of packages and libraries:



```
nomnga@nomnga-desktop:~/ns3/ns-allinone-3.19/ns-3.19$ cd build/
nomnga@nomnga-desktop:~/ns3/ns-allinone-3.19/ns-3.19/build$ ls
bindings                              libns3.19-netanim-debug.so
build-status.py                       libns3.19-netanim-test-debug.so
c4che                                 libns3.19-network-debug.so
config.log                            libns3.19-network-test-debug.so
examples                              libns3.19-nix-vector-routing-debug.so
libns3.19-antenna-debug.so            libns3.19-olsr-debug.so
libns3.19-antenna-test-debug.so       libns3.19-olsr-test-debug.so
libns3.19-aodv-debug.so               libns3.19-point-to-point-debug.so
libns3.19-aodv-test-debug.so          libns3.19-point-to-point-layout-debug.so
libns3.19-applications-debug.so       libns3.19-point-to-point-test-debug.so
libns3.19-applications-test-debug.so  libns3.19-propagation-debug.so
libns3.19-bridge-debug.so             libns3.19-propagation-test-debug.so
libns3.19-buildings-debug.so          libns3.19-sixlowpan-debug.so
libns3.19-buildings-test-debug.so     libns3.19-sixlowpan-test-debug.so
libns3.19-config-store-debug.so       libns3.19-spectrum-debug.so
libns3.19-core-debug.so               libns3.19-spectrum-test-debug.so
libns3.19-core-test-debug.so          libns3.19-stats-debug.so
libns3.19-csma-debug.so               libns3.19-stats-test-debug.so
libns3.19-csma-layout-debug.so        libns3.19-tap-bridge-debug.so
libns3.19-dsdv-debug.so               libns3.19-test-debug.so
libns3.19-dsdv-test-debug.so          libns3.19-test-test-debug.so
libns3.19-dsr-debug.so                libns3.19-topology-read-debug.so
libns3.19-dsr-test-debug.so           libns3.19-topology-read-test-debug.so
libns3.19-emu-debug.so                libns3.19-uan-debug.so
libns3.19-energy-debug.so             libns3.19-uan-test-debug.so
libns3.19-energy-test-debug.so        libns3.19-virtual-net-device-debug.so
libns3.19-fd-net-device-debug.so      libns3.19-visualizer-debug.so
libns3.19-flow-monitor-debug.so       libns3.19-wave-debug.so
libns3.19-flow-monitor-test-debug.so  libns3.19-wave-test-debug.so
libns3.19-internet-debug.so           libns3.19-wifi-debug.so
libns3.19-internet-test-debug.so      libns3.19-wifi-test-debug.so
libns3.19-lte-debug.so                libns3.19-wimax-debug.so
libns3.19-lte-test-debug.so           libns3.19-wimax-test-debug.so
libns3.19-mesh-debug.so               ns3
libns3.19-mesh-test-debug.so          scratch
libns3.19-mobility-debug.so           src
libns3.19-mobility-test-debug.so      utils
libns3.19-mpi-debug.so
nomnga@nomnga-desktop:~/ns3/ns-allinone-3.19/ns-3.19/build$ 
```

Figure 3.1: List of Packages and Libraries Needed as System Requirements

3.6. Conclusion

This chapter defined the research methodologies that guided this research work from its initial stage to its logical conclusion, providing details with regards to how the research progressed from the phase of research construction, via data collection to data analysis. Furthermore, the chapter presented all the network technologies that underpinned this research work such as **NS-3**, **iperf** and **Wireshark**. It discussed the functionalities of these network technologies, how they were integrated together to produce a smooth running simulated network and also shed light to their expected individual output performance. The overall system requirements for a robust and scalable simulated network were also presented through a dedicated focus on each and every network technology that the research work implemented. Finally, in this chapter, the system architecture, modelled on the deployed Dwesa WiMAX network, was presented both from a high-level and underlying technical perspective. In the next chapter, the system design and testing process is presented.

CHAPTER 4

SYSTEM DESIGN, SIMULATION AND TESTING

4.1. Introduction

This chapter discusses the system design, simulation and testing. The discussion starts by presenting the system architecture underlying this research project in which a conceptual model that defines the structure and functioning of the simulated network is defined. Subsequently, this chapter discusses the fundamental technical concepts of WiMAX networks from a theoretical perspective, a process which then places the correct foundation for the simulation implementation and the testing simulation process. The chapter further discusses the alignment of the simulation implementation process to the existing Dwesa WiMAX network to ensure that the outcomes of this research project are suitable for implementation at Dwesa.

This section serves as a bridge between the theoretical and technical aspects of this research project. It builds from the underlying scientific theory espoused in the research methodologies into the technical components, WiMAX models and interfaces that constitute the system design of this research project. In this regard, this chapter deals with the system design process, step-by-step, from planning to development and testing in detail. The details provided also depict the extent to which certain developmental rules and procedures were followed in designing the simulation network suitable to address the objectives of this research project.

4.2. System Architecture

According to Sieborger and Terzoli(2010) the AlvarionBreezeMAX technology was selected for deployment in Dwesa WiMAX network and formed part of the systems that were tested during the initiation phase in 2005 by SAAB Grintek, collaborating with the Telkom CoE in ICT4D of the UFH. Consequently, the collaboration saw to the provisioning of a micro BS and five SU kits, as mentioned in Chapter 2. Below, is Figure 3.2 which presents the system architecture of this research project as modelled on the WiMAX network deployed in Dwesa:

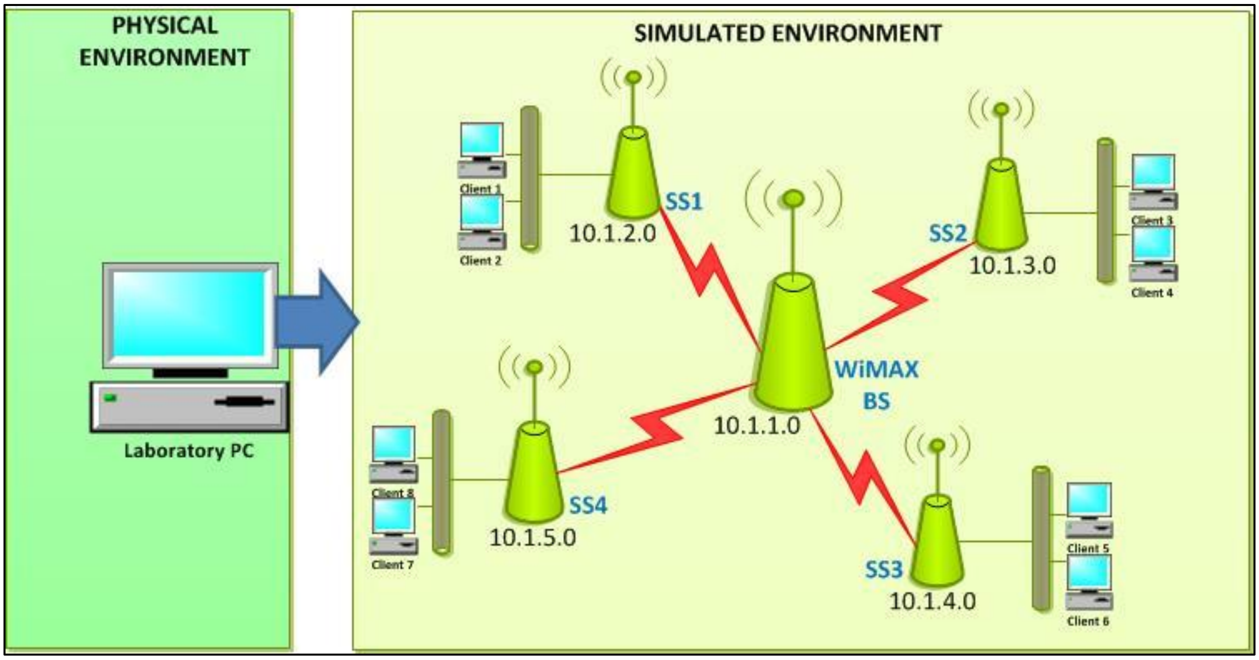


Figure 3.2: The Dwesa WiMAX Logical Network Diagram

Figure 3.2 above presents a scenario where the system architecture comprised of two environments, namely, the physical and simulated environment. The physical environment comprises of a work station in which the whole system is configured and manipulated. The simulated environment is hosted within this physical environment. The physical environment has been configured such that it supports all the system requirements needed for smooth running of the simulated environment. The simulated environment comprises of four ssNodes (SS1, SS2, SS3 and SS4) and one bsNode (BS). The bsNode is assigned an IPv4 address of **10.1.1.0** and ssNodes (SS1, SS2, SS3 and SS4) are assigned IPv4 addresses of **10.1.2.0**, **10.1.3.0**, **10.1.4.0** and **10.1.5.0** respectively. This IPv4 addressing is facilitated by **IPv4AddressHelper**, a simple IPv4 address generator class. The bsNode uses these IP addresses to locate the ssNodes in network and allocates bandwidth to them through the support of the Bandwidth Manager, a class which manages the bandwidth request and grant mechanism. According to (*NS-3: WiMAX models*, n.d) bsNodes' bandwidth manager works partners with the uplink scheduler for the determination of the total available bandwidth and allocation size per service flow. A service flow is "a MAC transport service that provides unidirectional transport of packets in the downlink or uplink" as described by Ahmadi(2010). Effectively, the implementation of these three classes, namely, the IPv4AddressHandler, bandwidth manager and service flow, ensured that the bsNode is connected to the ssNodes, can allocate bandwidth, and unidirectional transportation of packets was possible.

4.3. WiMAX Simulation Environment

According to Chaudhari and Karule(2014), the NS-3 platform provides the 802.16 model as an effort to ensure precise application of the 802.16 MAC and PHY level requirement for the PMP mode and the Wireless MAN-OFDM PHY layer. The following three layers form the primary composition of the model:

- The convergence sub-layer (CS)
- The MAC Common Part Sub-layer (MAC-CPS) and
- The Physical (PHY) layer

4.3.1. MAC Convergence Sub layer

According to Farooq and Turletti(2009) the 802.16 MAC layer is composed of two sub-layers, namely, the CS and the core MAC layer referred to as MAC-CPS. They continue to explain that sub-division continues in the CS layer leading intotwo layers in the form of Packet CS and the ATM CS. The module used for this research exploits the Packet CS, in order to utilize the higher layers' packet-based protocols. Additionally, Farooq and Turletti (2009) shed light that packets from the higher layers and peer SSs are received by the CS which is also responsible for their classification and processing to appropriate connections. They contend that CS is crucial in supporting the MAC CPS towards establishing the associated transport connection QoS parameters and provisioning of guaranteed QoS. CS achieves this mission through directing the transport connections to their appropriate service flows. Finally they assert that the CS uses a simple IP packet classifier on the basis of the end-node MAC address

4.3.1.1. IP Packet Classifier

According to Chaudhari and Karule (2014), on the basis of an agreed criteria, an IP packet classifier is employed to direct incoming packets to their appropriate connections. The duty of the IP packet classifier is to keep a register of directing rules that link each IP to each service flow. The classifier must further analyze the IP and the TCP/UDP headers in order to direct the incoming packets (the upper layer flows) to the suitable WiMAX connection queue. For both the BS and SS the classifier is implemented through the **IpcsClassifier** and **IpcsClassifierRecord** classes.

4.3.2. MAC Common Part Sub layer

Farooq and Turletti (2009) defines this layer as the main sub-layer of the WiMAX MAC and plays a major role in the MAC fundamental functioning. Additionally, they state that the module applied the PMP mode which helps the BS to facilitate interconnection and communication between multiple SSs. Furthermore, "framing and addressing, generation of MAC management messages, SS initialization and registration, service flow management, bandwidth management and scheduling services" are all said to be the key functions performed by the MAC CPS. Other classes that implement MAC functions do not do so directly but through other classes such as the"Link Manager (for both SS and BS), UL Scheduler, Scheduler (for both SS and BS), Connection Manager, Service Flow Manager, Burst Profile Manager and Bandwidth Manager," as articulated by Farooq and Turletti.

4.3.2.1 Framing and Management Messages

Chaudhari and Karule (2014) present that this module uses fixed time periods to implement and define a frame and frame boundaries, respectively. Further sub-dividing of a frame yields two sub-frames in the form of the DL and UL sub-frames.The TDD mode is applied by the standard creating a scenario for DL and UL operation at the same frequency but at separate times. Subsequently, the DL and UL sub-frames are then used to DL and UL burst allocation respectively. Following the WiMAX tolerance for burst packets sending and receiving in a particular Dl or UL burst, a packet burst is a MAC layer transmission unit hence the implementation of a special packet burst by the module. Essentially, what defines a packet burst is a registry of packets. (Chaudhari and Karule, 2014) continue to say the responsibility of generating the DL and UL sub-frames lies with the BS DL and UL schedulers implemented by the classes BS Scheduler and UL Scheduler, respectively. For the DL scenario, the simulation of the sub-frame is through consecutive bursts transmission whereas for the UL scenario, with accordance to time the sub-frame is divided into numerous slots. The bursts transmitted by the SSs in such slots are then aligned to slot boundaries. To ensure efficiency in bandwidth management, the dividing of a frame into integer number of symbols and PS is done. How the underlying PHY layer implementation is arranged determines the symbols' number per frame. Finally, the specification of symbols in unit entails DL or UL burst size.

4.3.2.2. Network Entry and Initialization

This phase comprises of two sub-phases, namely, scanning and synchronization, and initial ranging. Chaudhari and Karule (2014) succinctly describe this phase as one that is performed by LM of the BS and SS alike. Their description alludes that theLM component of the SS and BS performs the entire phase. Whenever the SS communicates its intentions to participate in the network, the first step to take is to search for a suitable channel through conducting a DL frequencies scan. Once a PHY frame is detected the searching of a suitable channel is deemed complete and SS synchronization with the BS becomes the next step. This step is deemed complete as soon as the DL-MAP message is received by the SS and synchronization lasts up until DL-MAP and DCD messages are not received.Now that the SS is synchronized with the BS successfully, to acquire UL channel parameters SS must first wait for a UCD message. Upon successful acquisition, the first sub-phase of the network entry and initialization is deemed complete. As part of enjoying the synchronization benefits, the SS must locate in the UL sub-frame a special grant referred to as initial ranging interval but first has to wait for a UL-MAP message. This special grant is always the first grant to be allocated by the BS UL Scheduler at regular intervals if present. According to Farooq and Turletti(2009) the norm is that using the BPSK 1/2 modulation/FEC burst profile at 0.5ms intervals this special grant is always directed towards the Broadcast CID notwithstanding the user rights to modify the simulation script value.This assertion can be further demonstrated through Figure 4.1. below:

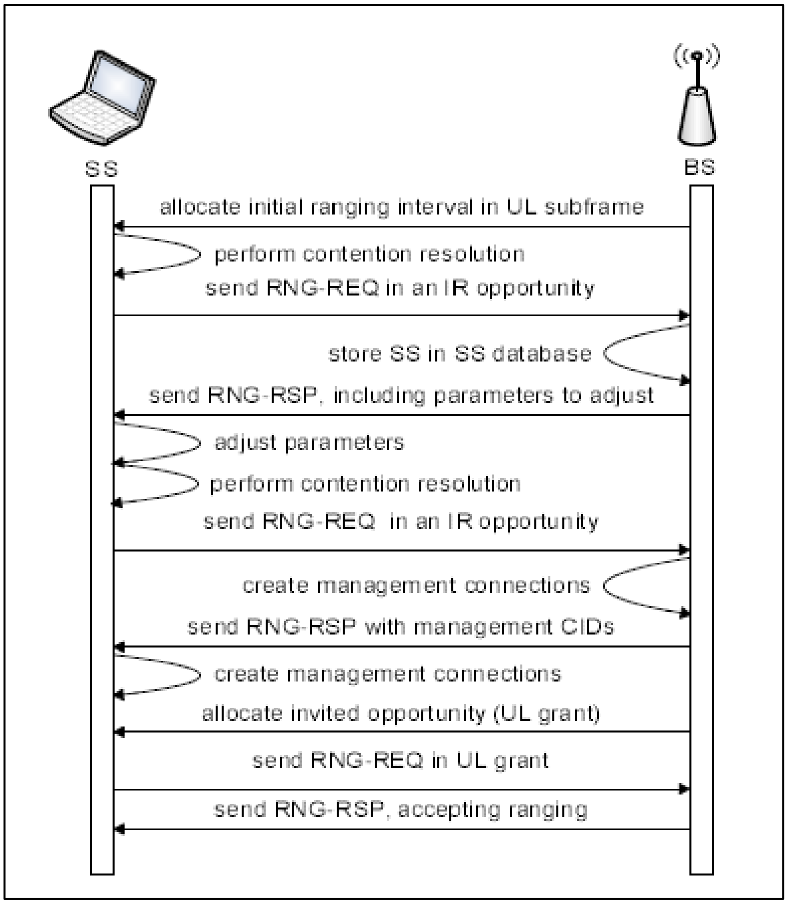


Figure 4.1 Initial Ranging Process by Farooq and Turletti (2009)

In explaining Figure 4.1, Farooq and Turletti (2009) assert that the a minimum of one transmission opportunities initial composes the ranging interval. The BS determines the fixed size of an opportunity and generously affords time for sending of RNG-REQ message, the sending is inclusive of the overhead.An RNG-REQ message is sent out by the SS once located, following a backoff based contention resolution process performance. Once the RNG-REQ message is received by the BS, a response in the form of a RNG-RSP message is sent to notify the SS whether or not the ranging is accepted, or at times the indication can direct for the adjustment of the SS timing offset or power parameters. This is the time at which theSS is added on the SSs database by SS Manager component. This is the component the BS uses in its functions to comprehensively manage all the participating SSs. On the other side, the SS and BS will continue to exchange the RNG-REQ and RNG-RSP messages respectively, up until there is a determination on the initial ranging being accepted or rejected. Figure 4.1 above further presents a simplified process of initial ranging phase. It has to be noted that until the SS is allocated with the basic and primary management connections by the BS through the RNG-RSP message, the initial ranging is mainlyimplemented on a connection that is predefined. This propels for the allocation of a unicast, an invited ranging opportunity, to the SS which is the “always available first UL grant” for the purposes of allowing the ranging process to continue.

4.3.2.3. Connections and Addressing

According to Chaudhari and Karule (2014) at the MAC layer, communication is done with respect to connections. Subsequently, the WiMAX uses this scenario and thus consider the unidirectional transmision of traffic between the SS and BS's MAC entities to define a connection. As a consequence, two types of connections are defined for this standard, namely, “the management connections for transmitting control messages and transport connections for data transmission,” add Chaudhari and Karule. To identify a connection, a 16-bit CID is used. It has to be noted that when packets are queued for transmission on a particular connection, each connection should maintain its own transmission queue. Albeit, creating and managing connections for all SSs is bestowed responsibility of the BS’ Connection Manager component. However, the Basic and Primary management connections which form two WiMAX key management connections, are established for the SS during the ranging process. Following all **unicast** DL and UL grants being mapped towards SS's Basic CID, the Basic connection becomes crucial throughout the operation of SS. Additionally to management connections, a minimum of one transport connection is allowed for a SS forward data packet and this is managed at the SS by theSS Connection Manager component. The form of the management connection is bidirectional as per the standard definition, meaning a single CID serves for a pair of DL and UL connections. In practical terms, in a DL direction, once a BS receives the CID, an identical connection is created by the SS using the very same CID the BS had received.

4.3.2.4 Scheduling Services

In line with the four scheduling services, namelythe UGS, rtPS, nrtPS and BE, defined by 802.16-2004 standard, the NS-3 module supports their implementation and testing.As discussed in Chapter 2, the aforementioned scheduling services performin a different manner from each other and this can be drawn from the manner in which they facilitate bandwidth requests from the BS and also how bandwidth is subsequently allocated to their SSs. However, a service flow is strictly connected to one scheduling service at a time using the service flows' to identify an appropriate scheduling service for this purpose. Upon service flow successful creation and based on the associated QoS parameter set, the necessary parameters in the form of grant size and grant interval are calculated by the UL Scheduler.

4.3.2.5 WiMAX Uplink Scheduler Model

UL allocations are decided upon by the UL Scheduler at the BS on the basis of the service flow's associated QoS parameters and the amount of bandwidth the SSs requests. Following this step, a complete scheduling service functionality is performed by the UL scheduler in conjunction with the Bandwidth Manager. According to Chaudhari and Karule(2014), three different versions of schedulers are defined by the current WiMAX module as follows:

- The FCFS, regarded as a simple priority-based. Regular grant allocations are done to service the real-time services in the form of **UGS and rtPS** by the BS based on the planned interval. Inversely, if available after service priority was given to real-time connections; only minimum reserved bandwidth is guaranteed to the service of the non-real-time services in the form of **nrtPS and BE**.

- Similarly to FCFS scheduler but with an exception of **rtPS**, in the second scheduler, depending on the available amount of data, rtPS connections can transmit the queued data packet. With this service flow, effective available bandwidth redistribution is implemented, wherein all rtPS with a minimum of one packet to forward through are prioritized through a mechanism called bandwidth saturation control. In this case, the nrtPS and BE Connections are only allocated the remaining bandwidth. Classes BS **SchedulerRtps** and UL **SchedulerRtps** implements this scheduler.

- The third and last scheduler is the MBQOS uplink scheduler. In this scheduler, three priority queues, namely low, intermediate and high are considered. Upon receiving requests, a priority check is conducted from high to low priority queue. The BE service flow bandwidth requests are considered as low priority queues whilst the **rtPS** and by **nrtPS** connections bandwidth requests are regarded as the intermediate priority queues. However, to guarantee that QoS requirements are met, the **rtPS** and **nrtPS** requests can advance to the higher priority queue. Following the advance, periodic grants and unicast request opportunities are stored in the high priority queue and are mandatory for scheduling in the following frame. In addition, the BS places a deadline for all intermediate queued rtPS bandwidth requests in order to guarantee the maximum delay requirement. Over a duration T window both **rtPS** and **nrtPS** minimum bandwidth connection requirements are guaranteed. Class **UplinkSchedulerMBQoS** implements this scheduler, MBQOS.

4.3.2.6. WiMAX Outbound Schedulers Model

The understanding behind the standard outbound schedulers' model is that in addition to the uplink scheduler the **BSScheduler** and **SSScheduler** are the outbound schedulers at BS and SS side, respectively, assert Chaudhari and Karule (2014). Their main responsibility is to determine the data packets that will be transmitted in a particular allocation. The **BSScheduler** is mandated to schedule the DL traffic while the **SSScheduler** schedules the UL traffic. Although the BS scheduler and the two outbound schedulers, BSScheduler and SSScheduler, conform to the principle of allocating bandwidth grants from highest to lowest priority queues, they all have been designed to operate as FCFS scheduler, as articulated by Chaudhari and Karule (2014).

4.3.3. Bandwidth Request and Grant Mechanism

As **rtPS**, **nrtPS** and **BE** makes bandwidth request through sending a bandwidth request packet, bandwidth request and grant mechanism forms a vital component of the standards' MAC contends Chaudhari and Karule (2014). Additionally, as the Bandwidth Manager components of BS and SS facilitate bandwidth request and granting methods, the whole mechanism operates in the following manner. The BS polls the SS with **rtPS**, **nrtPS** or **BE** flow through request opportunity allocation periodically. A request opportunity is essentially a UL grant, mapped to the basic CID of the SS albeit that its purpose does not serve the sending of data packets. In order for the UL grant to be distinguished from ordinary UL grants and serve its purpose, a robust burst profile must be defined in advance in order to send the bandwidth request it entails. Another feature of the UL grant is its fixed size that does not require sufficient bandwidth to be transmitted and entails a special bandwidth request header. Upon being received, the BS looks at the BR field of this header to determine the requested bandwidth amount. As and when the BS poll is received by the SS, the outbound scheduler of the SS go over rtPS, nrtPS and BE flows and any other flow that might have pending packets and fix the BR field size to that of data packet placed in front of the queue. The flow's CID is subsequently used as the headers' CID field. Once all these have been set, the UL grand is then forwarded prompting the BS to respond with allocating bandwidth accordingly using subsequent frames. The inclusion of bandwidth request header is mutually exclusive with generic MAC header. Forwarding of UL grants piggybacked with data is defined and allowed as an option in WiMAX. The outbound scheduler of the SS is then bestowed with a prerogative to elect a suitable flow it will use to forward its data in the allocated bandwidth grant. For the **UGS** and **rtPS** flows, the scheduler picks the flow whose interval is closest to elapse and achieves this through making use of the unsolicited grant interval and unsolicited polling interval parameters accordingly. As for **nrtPS** and **BE**, the non-real time flow, gets the opportunity to be serviced when there is still bandwidth left after allocations. SSs with non-UGS flow are required to indicate this to the BS so that they may be polled to send forward bandwidth request considering that all SSs are automatically allocated grants on the basis that they have at least one UGS service flow to service their UGS traffic. This is achieved by setting a GMS, the poll-me field of a kind of sub-header which is incorporated with the Generic MAC Header. Once the UGS packet with a GMS is received, the BS initiates the SS polling process for the indicated non-UGS flows.

4.3.4. WiMAX PHY Model

According to Pathak (2013), "the WiMAX air interface (PHY Layer) is based on OFDM and the WiMAX RF signals use OFDM techniques and its signal bandwidth can range from 1.25 to 20 MHz." Therefore, the symbol must correspond to the carrier spacing if orthogonality between the individual carriers is to be maintained. Mohamed *et al.* (2010) describes the usage of the OFDM include supporting of high-speed data, video, and multimedia communications across numerous industry broadband systems. Furthermore, processing of data frames forwarded by the upper layers to an appropriate format for wireless channel transmission forms the primary responsibility of the PHY layer. This goal is achieved through implementing "channel estimation, FEC coding, modulation, mapping in OFDMA symbols, amongst others" as Fernando *et al* (1998) explain. According to Farooq and Turletti (2009), there exist two different versions of the PHY layer provided by the module, the basic PHY implementation and OFDM PHY layers. To differentiate between the two, the basic PHY implementation, upon the MAC layer receiving bursts, it simply forwards those bursts even without considering the fundamental PHY layer details. Secondly, there exists an OFDM PHY layer designed in line with the Wireless MAN-OFDM specification. Packet bursts are converted to bit-streams and then split into smaller FEC blocks due to the channel encoding blocks, alternatively FEC blocks, that are dealt with by the OFDM PHY.

Farooq and Turletti (2009) presents the OFDM PHY operations as a scenario where a packet burst is converted into a plain stream of bits as and when it is received from the MAC layer. Subsequently, the plain bit-stream, dependent on the incumbent modulation scheme implemented, is further split into smaller FEC blocks to be sent, individually, to the OFDM module. Once received, the OFDM module conducts the necessary underlying PHY functionalities on the block and sends it out for block transmission by the channel layer. Upon the PHY layer receiving the block, operations to convert and link the blocks back to the bit-stream and packet burst are performed subsequently. Finally the packet burst is now sent to the MAC layer as Figure 4.2 demonstrate this process:

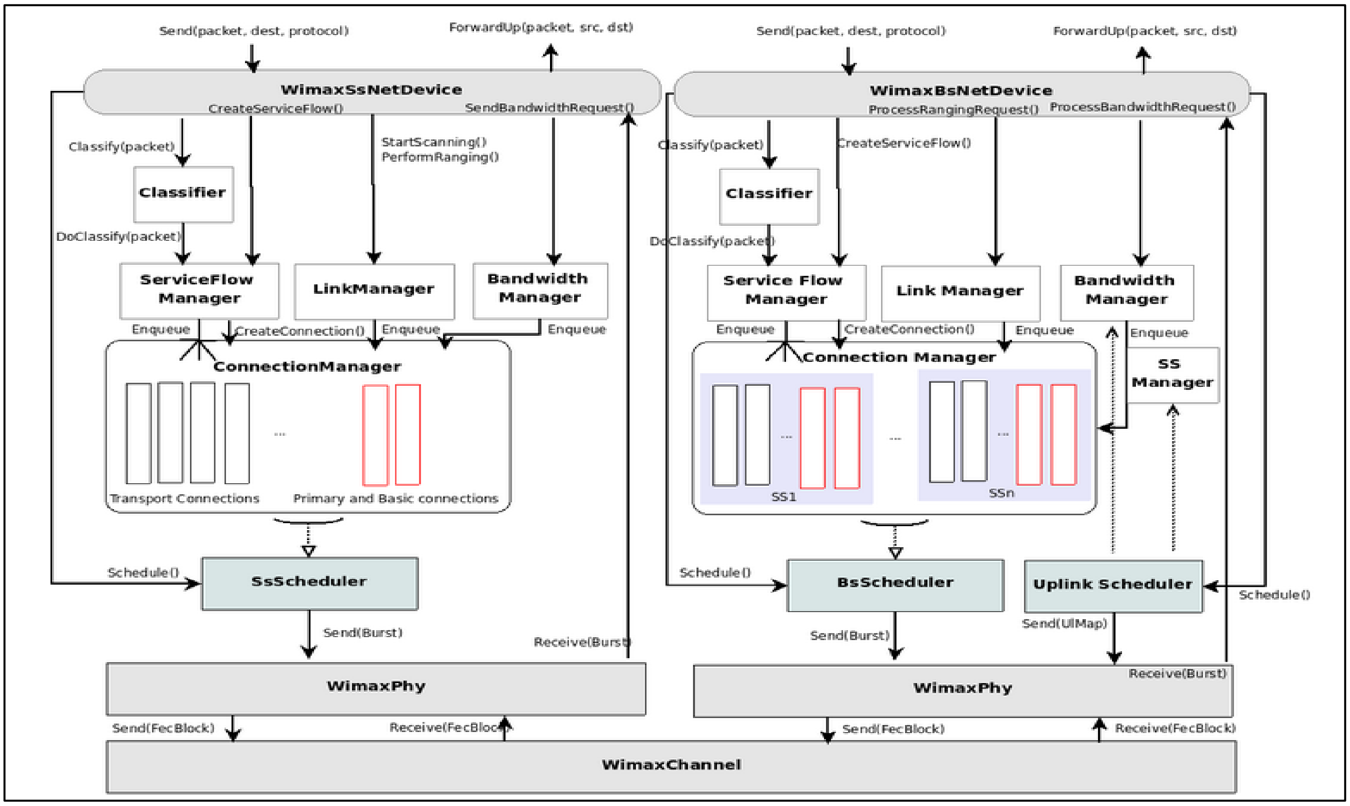


Figure 4.2 Overview of the WiMAX Model ("NS-3: WiMAX Models," n.d.)

The systematic implementation of the aforementioned sections has successfully guided the process of simulating the Dwesa WiMAX network in the model. This implementation has yielded the following scenario.

4.4. The Simulated Dwesa Network Scenario

In line with the network setup of Dwesa WiMAX which actually consists of five (5) nodes wherein one (1) in Ngwane school serves as the BS (represented by node4 in Figure 4.3 below) and the other four (4) serve as the SSs (represented by nodes 0, 1, 2 and 3 in Figure 4.3 below). The simulated network using the NS-3 software comprises of five (5) nodes as well wherein node4 serves as the BS and the others as SSs. The following figures 4.3, 4.4, 4.5 and 4.6 shows the network topology and connection between the BS and SSs:

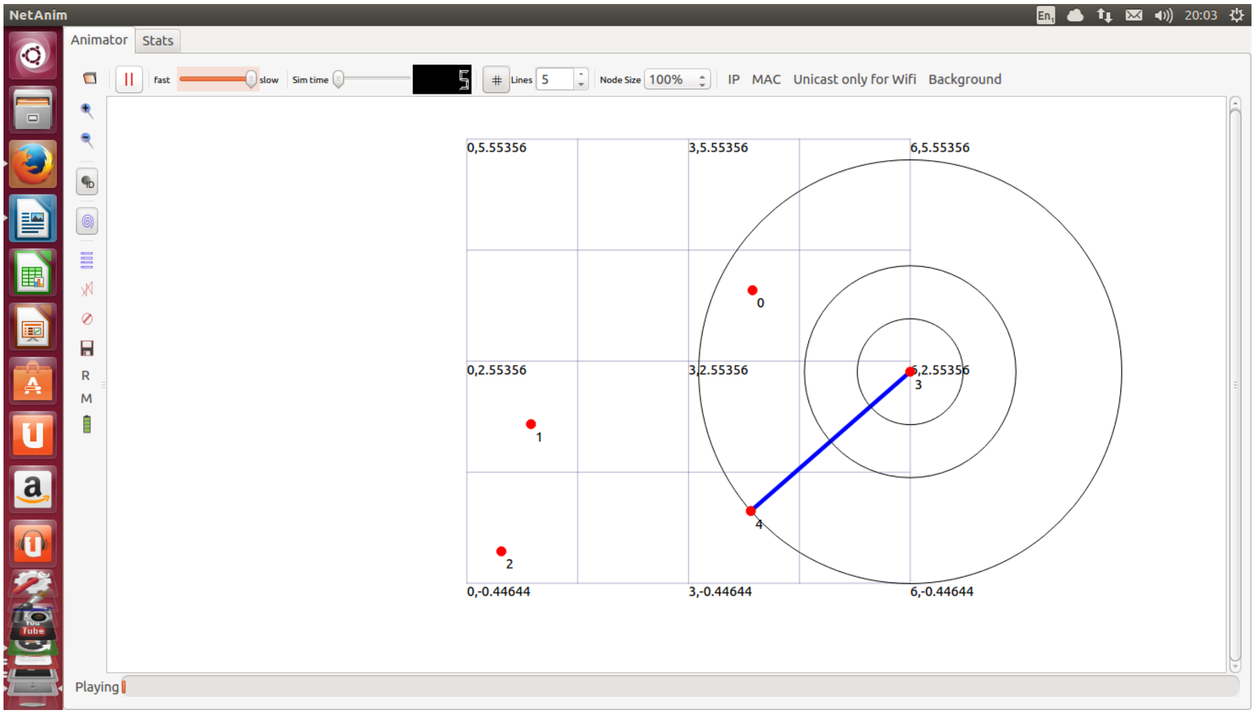


Figure 4.3 Network Topology and Connection between node4(the BS) and node3(the SS)

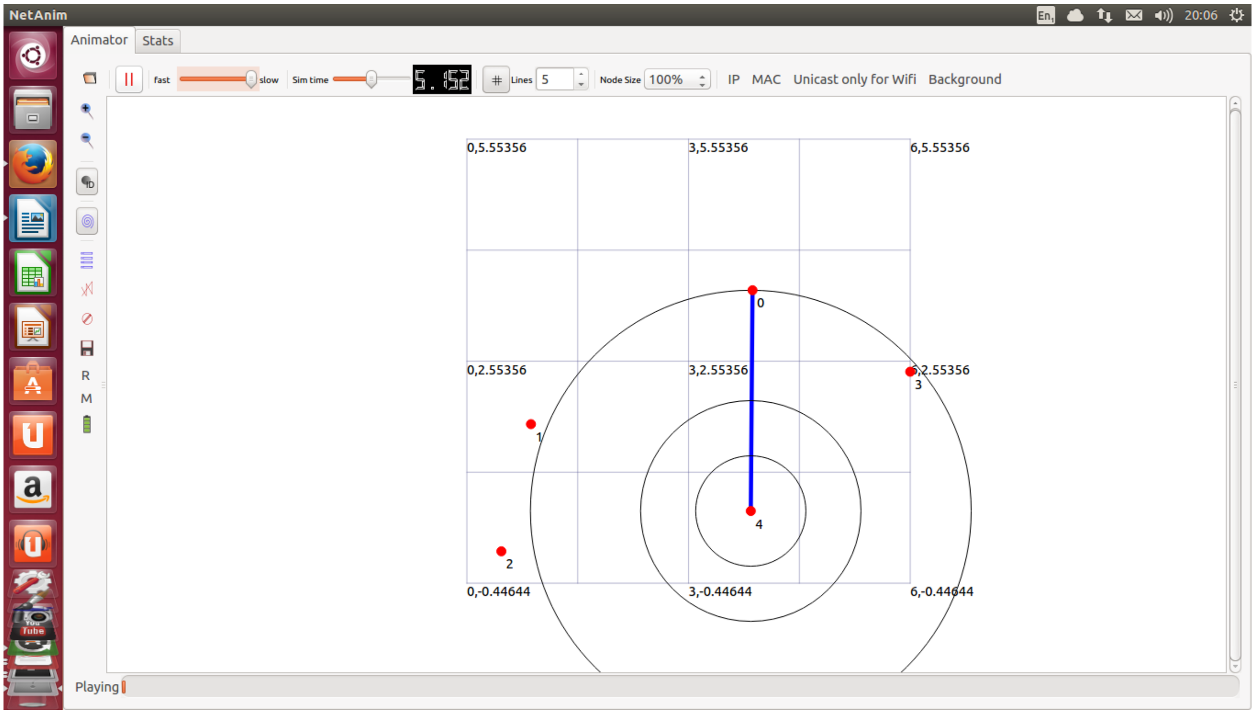


Figure 4.4 Network Topology and Connection between node4 (the BS) and node0(the SS)

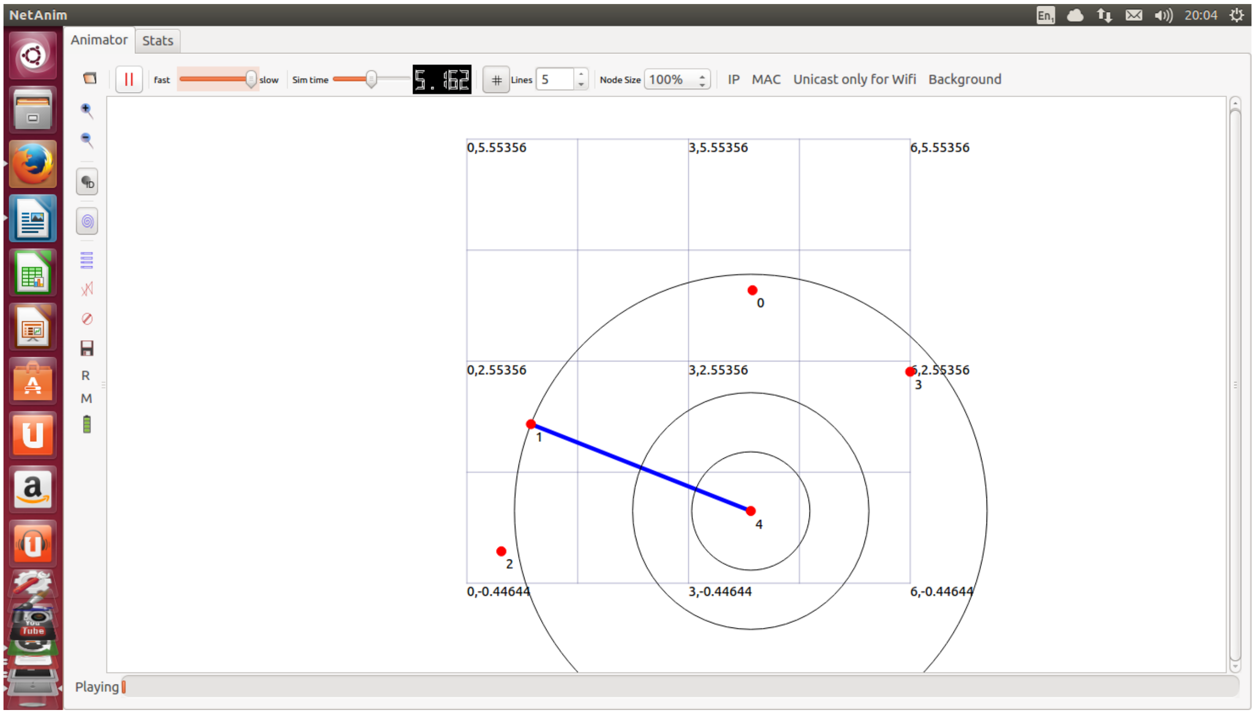


Figure 4.5 Network Topology and Connection between node4(the BS) and node1(the SS)

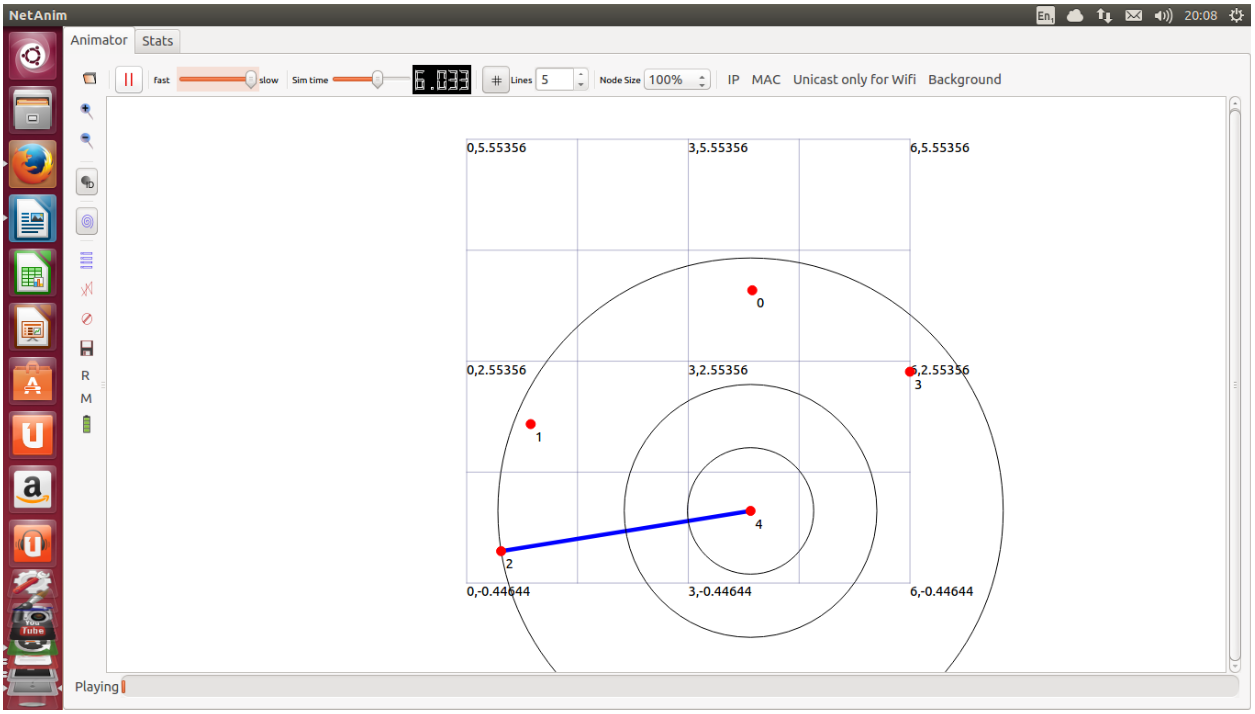


Figure 4.6 Network Topology and Connection between node4 (the BS) and node2(the SS

Table 4.1 below entails the simulation and node parameters that were set for this research project:

Table 4.1 The Simulation and Node Parameters Set for this Research Project

Sr No	**Efficiency Mode**	**Mobility and ranging enabled**
1	AC service class definition (QoS)	• UGS (VOIP(IP Telephony)) • rtPS(MPEG(High resolution video))
2	Modulation technique	Wireless OFDMA
3	Number of subscribers	4
4	Bandwidth	20Mhz
5	Duplexing Technique	OFDM
6	Scheduling Types	BE, rtPS, UGS, nrtPS
7	Maximum Sustained Traffic Rate(bps)	5Mbps
8	Maximum Reserved Traffic Rate(bps)	1Mbps

	OFDM PHY Profile	
9	Number of Rows	2
10	Modulation and Coding	QPSK1/2, QPSk3/4
11	Start Time(seconds)	Uniform(100,110)
12	Duration(seconds)	End of Simulation
13	Type of SAP	IP
14	PHY Profile Type	OFDM
15	BS MAC Address	5

4.5. Conclusion

This chapter discussed the system design in line with the system architecture of this research project, which is the Dwesa WiMAX network setup. Furthermore, the chapter presented the system testing process which depicts the details of the simulation and connection between the BS and each SS of the network. The following chapter entails the implementation process and results presentation.

CHAPTER 5

IMPLEMENTATION AND RESULTS

5.1. Introduction

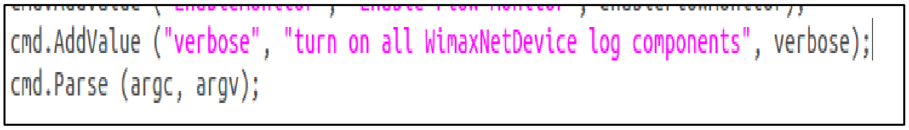
This chapter presents two aspects of this research project, the implementation process and the subsequent results obtained therefrom The NS-3 models and systems that were put in place for this research project are presented under the first section of this chapter, the implementation process. The second and last section of this chapter details the collective output of these models and systems as monitored through a series of scenarios during the implementation process as the overall results of this research project.

5.2. Implementation

Following the system testing process, NS-3 models and systems were implemented. These models and systems included the WimaxNetDevice, WiMAX attributes, tracing of network flows, framing and management of messages, the scheduling services, mobility, propagation types, network module, flow monitoring, and data collection. Under this section, a closer look at how these models and systems were implemented is taken.

5.2.1. WimaxNetDevice

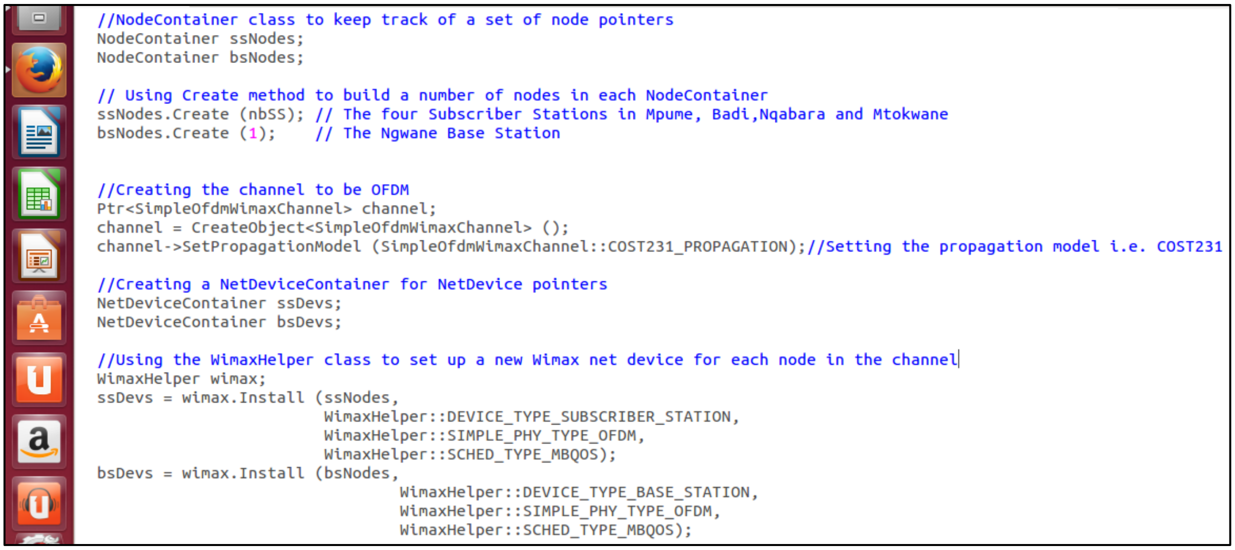
The WimaxNetDevice holds together numerous WiMAX-related class objects in a **NetDevice**, a network layer to device interface. These WiMAX classes include the class **Node**, **Packet**, **TraceContext,TraceResolver**, **Channel**, **WimaxChannel**, **PacketBurst**, **BurstProfileManager**, **ConnectionManager**, **ServiceFlowManager**, **BandwidthManager,** and **UplinkScheduler**(*NS-3: WiMAX models*, n.d). Generally, the WimaxNetDevice requires the "**ns3/wimax-module.h**" to be included and the following statement to activate its log components:



```
cmd.AddValue ("verbose", "turn on all WimaxNetDevice log components", verbose);
cmd.Parse (argc, argv);
```

Figure 5.1.WimaxNetDevice Declaration Statement to Turn on all Log Components

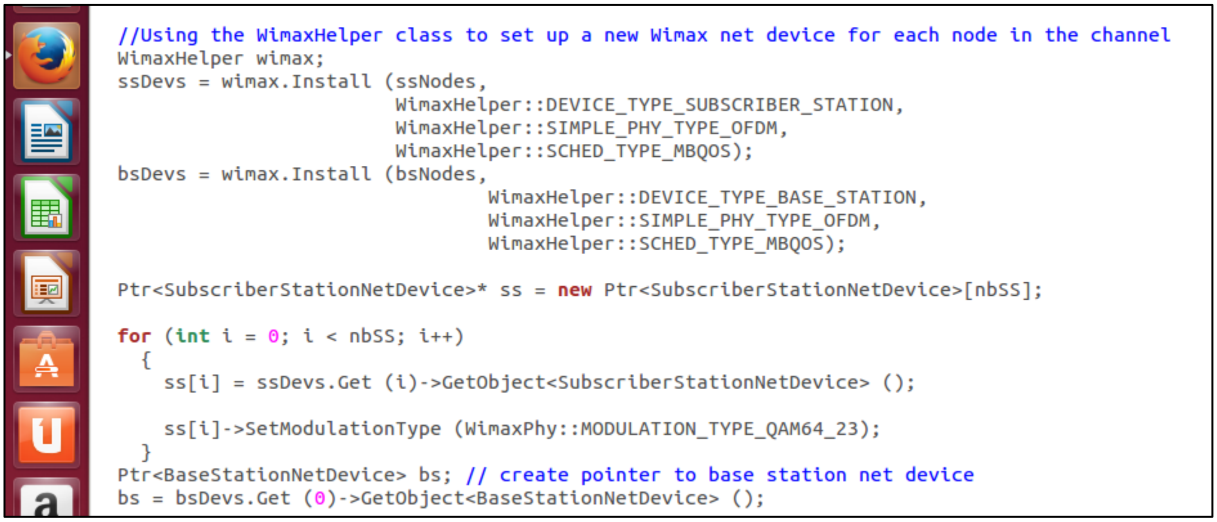
Following this declaration, WiMAX-related objects such as **NodeContainers**, channels, NetDeviceContainer, **SubscriberStationNetDevice,** and **BaseStationNetDevice** were implemented in the program as depicted below:



```
//NodeContainer class to keep track of a set of node pointers
NodeContainer ssNodes;
NodeContainer bsNodes;

// Using Create method to build a number of nodes in each NodeContainer
ssNodes.Create (nbSS); // The four Subscriber Stations in Mpume, Badi,Nqabara and Mtokwane
bsNodes.Create (1);    // The Ngwane Base Station


//Creating the channel to be OFDM
Ptr<SimpleOfdmWimaxChannel> channel;
channel = CreateObject<SimpleOfdmWimaxChannel> ();
channel->SetPropagationModel (SimpleOfdmWimaxChannel::COST231_PROPAGATION);//Setting the propagation model i.e. COST231

//Creating a NetDeviceContainer for NetDevice pointers
NetDeviceContainer ssDevs;
NetDeviceContainer bsDevs;

//Using the WimaxHelper class to set up a new Wimax net device for each node in the channel
WimaxHelper wimax;
ssDevs = wimax.Install (ssNodes,
                        WimaxHelper::DEVICE_TYPE_SUBSCRIBER_STATION,
                        WimaxHelper::SIMPLE_PHY_TYPE_OFDM,
                        WimaxHelper::SCHED_TYPE_MBQOS);
bsDevs = wimax.Install (bsNodes,
                                WimaxHelper::DEVICE_TYPE_BASE_STATION,
                                WimaxHelper::SIMPLE_PHY_TYPE_OFDM,
                                WimaxHelper::SCHED_TYPE_MBQOS);
```

Figure 5.2.NodeContainer, Channel and NetDeviceContainer Classes Implementation

Figure 5.2 above depicts the implementation of the **NodeContainer** class which enabled the implementation process to create and keep track of the network **ssNodes** and a **bsNodes**. Additionally, the channel class enabled the implementation process to create the OFDM WiMAX channels between the nodes in order to provide a media for network data transmission. Furthermore, the **WimaxNetDevice** also informed the implementation of the **SubscriberStationNetDevice** and **BaseStationNetDevice** classes as depicted in Figure 5.3 below:

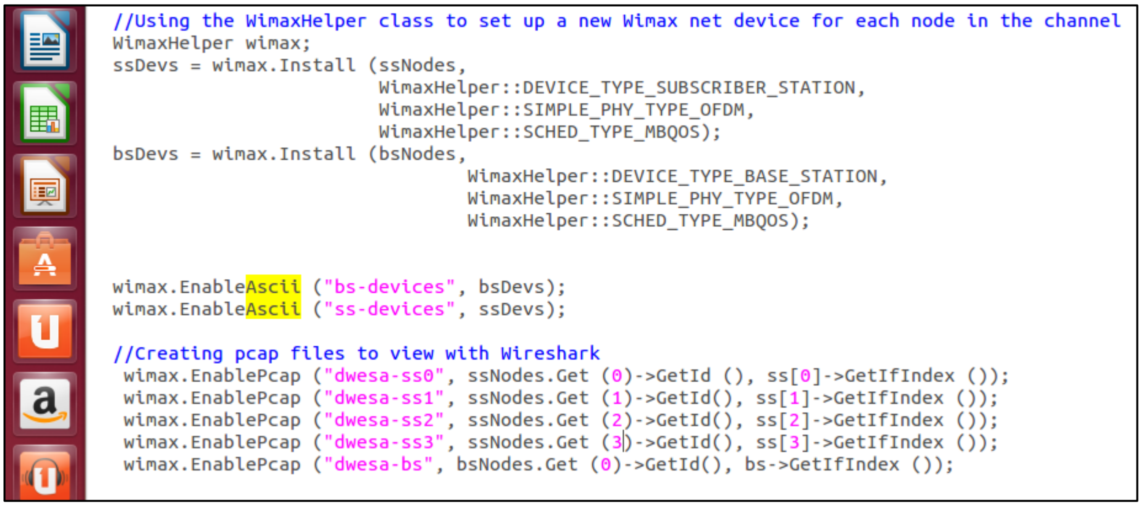


```
//Using the WimaxHelper class to set up a new Wimax net device for each node in the channel
WimaxHelper wimax;
ssDevs = wimax.Install (ssNodes,
                        WimaxHelper::DEVICE_TYPE_SUBSCRIBER_STATION,
                        WimaxHelper::SIMPLE_PHY_TYPE_OFDM,
                        WimaxHelper::SCHED_TYPE_MBQOS);
bsDevs = wimax.Install (bsNodes,
                                WimaxHelper::DEVICE_TYPE_BASE_STATION,
                                WimaxHelper::SIMPLE_PHY_TYPE_OFDM,
                                WimaxHelper::SCHED_TYPE_MBQOS);

Ptr<SubscriberStationNetDevice>* ss = new Ptr<SubscriberStationNetDevice>[nbSS];

for (int i = 0; i < nbSS; i++)
  {
    ss[i] = ssDevs.Get (i)->GetObject<SubscriberStationNetDevice> ();

    ss[i]->SetModulationType (WimaxPhy::MODULATION_TYPE_QAM64_23);
  }
Ptr<BaseStationNetDevice> bs; // create pointer to base station net device
bs = bsDevs.Get (0)->GetObject<BaseStationNetDevice> ();
```

Figure 5.3.SubscriberStationNetDevice and BaseStationNetDevice Classes Implementation

5.2.2. WiMAX Tracing

As the packets were transmitted between the nodes, the **WimaxHelper** was also implemented to trace their movement. This helped the implementation process to "access the low level trace sources that exist in the WiMAX physical layer, net device, and queue models” (“NS-3”, 2015).Figure 5.4 below presents the implementation of this helper in the program:



```
//Using the WimaxHelper class to set up a new Wimax net device for each node in the channel
WimaxHelper wimax;
ssDevs = wimax.Install (ssNodes,
                        WimaxHelper::DEVICE_TYPE_SUBSCRIBER_STATION,
                        WimaxHelper::SIMPLE_PHY_TYPE_OFDM,
                        WimaxHelper::SCHED_TYPE_MBQOS);
bsDevs = wimax.Install (bsNodes,
                                WimaxHelper::DEVICE_TYPE_BASE_STATION,
                                WimaxHelper::SIMPLE_PHY_TYPE_OFDM,
                                WimaxHelper::SCHED_TYPE_MBQOS);


wimax.EnableAscii ("bs-devices", bsDevs);
wimax.EnableAscii ("ss-devices", ssDevs);

//Creating pcap files to view with Wireshark
 wimax.EnablePcap ("dwesa-ss0", ssNodes.Get (0)->GetId (), ss[0]->GetIfIndex ());
 wimax.EnablePcap ("dwesa-ss1", ssNodes.Get (1)->GetId(), ss[1]->GetIfIndex ());
 wimax.EnablePcap ("dwesa-ss2", ssNodes.Get (2)->GetId(), ss[2]->GetIfIndex ());
 wimax.EnablePcap ("dwesa-ss3", ssNodes.Get (3)->GetId(), ss[3]->GetIfIndex ());
 wimax.EnablePcap ("dwesa-bs", bsNodes.Get (0)->GetId(), bs->GetIfIndex ());
```

Figure 5.4. The WiMAX Tracing Implementation

Figure 5.4 above depicts **WimaxHelper** class implementation for setting up a new WiMAX net device for each node in the channel. The **WimaxHelper**has built-in Pcap or Ascii tracing APIs which were called in the form of **EnableAscii** and **EnablePcap** to trace packet movement between the nodes. EnablePcap method enabled pcap individual node tracing such that packet capturing of each node of the network was achieved.

5.2.3. Class of Services (CoS)

As introduced in chapter 2, the WiMAX module supports the four CoSes defined by the 802.16-2004 standard, namely, the:

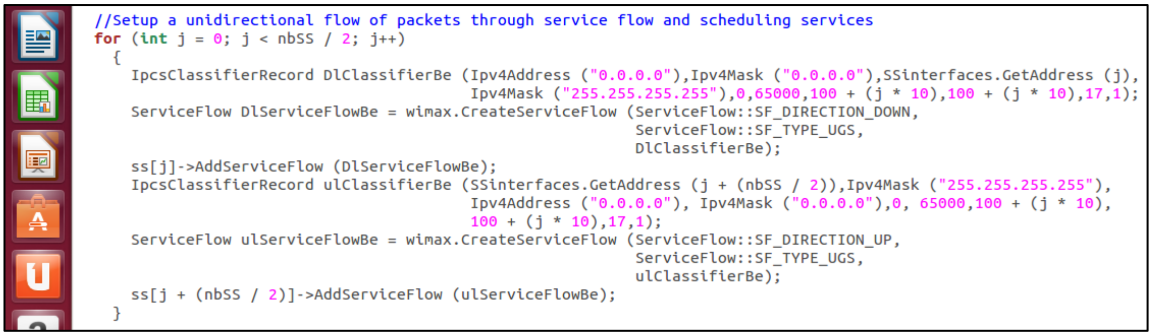
- Unsolicited Grant Service (UGS)
- Real-Time Polling Services (rtPS)
- Non Real-Time Polling Services (nrtPS)
- Best Effort (BE)

The manner in which these CoSes behave differs at varying degrees as it relates to how they request for bandwidth allocation and how it is subsequently granted to them. Thus, each service flow is associated to exactly one CoS at a time. This association provides a better view for future analysis of each CoS special effects on the overall network performance. The implementations of these

CoSes are carried out together with the WiMAX uplink and downlink primary scheduler types, namely, the:

- SIMPLE: a simple priority based FCFS scheduler
- RTPS: a real-time polling service (rtPS) scheduler
- MBQOS: a migration-based uplink scheduler

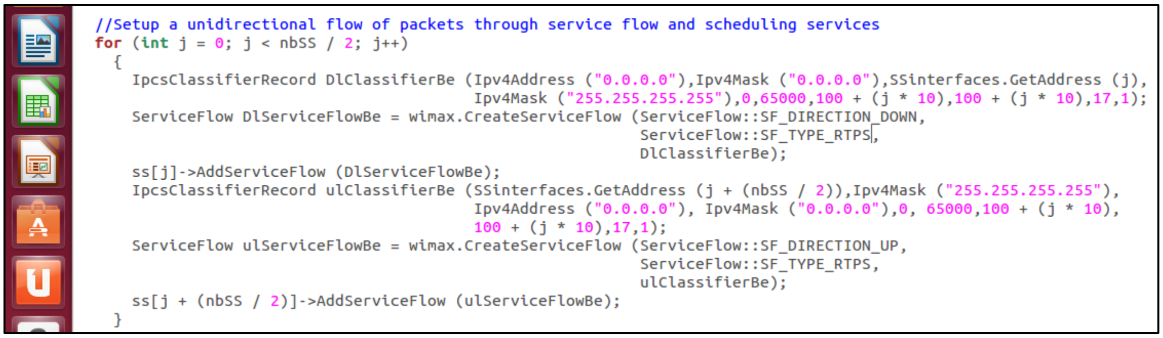
This understanding necessitated that twelve (12) separate scenarios of each CoS and each scheduler type be implemented to grasp an uninfluenced view of how bandwidth would be requested, granted and utilized given the underlying QoS parameters that each CoS defines. This means, each one (1) of the four (4) CoSes was implemented together with each one (1) of the three (3) scheduler types. The implementation of each CoS is presented in the following figures, 5.5, 5.6, 5.7 and 5.8. To start with, the UGS CoS implementation is presented in Figure 5.5 below:



```
//Setup a unidirectional flow of packets through service flow and scheduling services
for (int j = 0; j < nbSS / 2; j++)
  {
    IpcsClassifierRecord DlClassifierBe (Ipv4Address ("0.0.0.0"),Ipv4Mask ("0.0.0.0"),SSinterfaces.GetAddress (j),
                                         Ipv4Mask ("255.255.255.255"),0,65000,100 + (j * 10),100 + (j * 10),17,1);
    ServiceFlow DlServiceFlowBe = wimax.CreateServiceFlow (ServiceFlow::SF_DIRECTION_DOWN,
                                                           ServiceFlow::SF_TYPE_UGS,
                                                           DlClassifierBe);
    ss[j]->AddServiceFlow (DlServiceFlowBe);
    IpcsClassifierRecord ulClassifierBe (SSinterfaces.GetAddress (j + (nbSS / 2)),Ipv4Mask ("255.255.255.255"),
                                         Ipv4Address ("0.0.0.0"), Ipv4Mask ("0.0.0.0"),0, 65000,100 + (j * 10),
                                         100 + (j * 10),17,1);
    ServiceFlow ulServiceFlowBe = wimax.CreateServiceFlow (ServiceFlow::SF_DIRECTION_UP,
                                                           ServiceFlow::SF_TYPE_UGS,
                                                           ulClassifierBe);
    ss[j + (nbSS / 2)]->AddServiceFlow (ulServiceFlowBe);
  }
```

Figure 5.5 Unsolicited Grant Service (UGS) Implementation

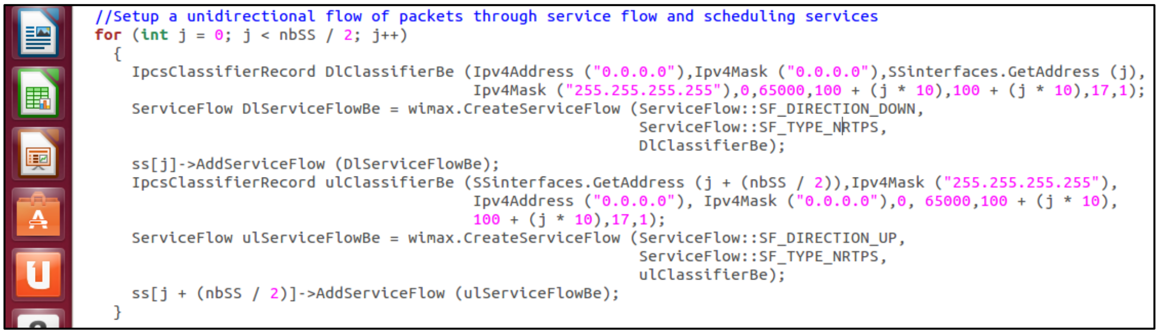
Figure 5.5 depicts a setup of a unidirectional flow of packets through a service flow type **UGS** for both the SS DL and BS UP service flows. The setup is implemented across all the network nodes i.e. SSs and BS, with the aid of WiMAX classes such as **IpcsClassifierRecord** which creates a classifier records and sets its parameters (**srcAddress, srcMask, dstAddress**, and protocol, amongst others) and **ServiceFlow** which is responsible for the creation of service flows. In a similar method, the rtPS CoS was implemented as presented in Figure 5.6 below:



```
//Setup a unidirectional flow of packets through service flow and scheduling services
for (int j = 0; j < nbSS / 2; j++)
  {
    IpcsClassifierRecord DlClassifierBe (Ipv4Address ("0.0.0.0"),Ipv4Mask ("0.0.0.0"),SSinterfaces.GetAddress (j),
                                         Ipv4Mask ("255.255.255.255"),0,65000,100 + (j * 10),100 + (j * 10),17,1);
    ServiceFlow DlServiceFlowBe = wimax.CreateServiceFlow (ServiceFlow::SF_DIRECTION_DOWN,
                                                           ServiceFlow::SF_TYPE_RTPS,
                                                           DlClassifierBe);
    ss[j]->AddServiceFlow (DlServiceFlowBe);
    IpcsClassifierRecord ulClassifierBe (SSinterfaces.GetAddress (j + (nbSS / 2)),Ipv4Mask ("255.255.255.255"),
                                         Ipv4Address ("0.0.0.0"), Ipv4Mask ("0.0.0.0"),0, 65000,100 + (j * 10),
                                         100 + (j * 10),17,1);
    ServiceFlow ulServiceFlowBe = wimax.CreateServiceFlow (ServiceFlow::SF_DIRECTION_UP,
                                                           ServiceFlow::SF_TYPE_RTPS,
                                                           ulClassifierBe);
    ss[j + (nbSS / 2)]->AddServiceFlow (ulServiceFlowBe);
  }
```

Figure 5.6 Real-Time Polling Service (rtPS) Implementation

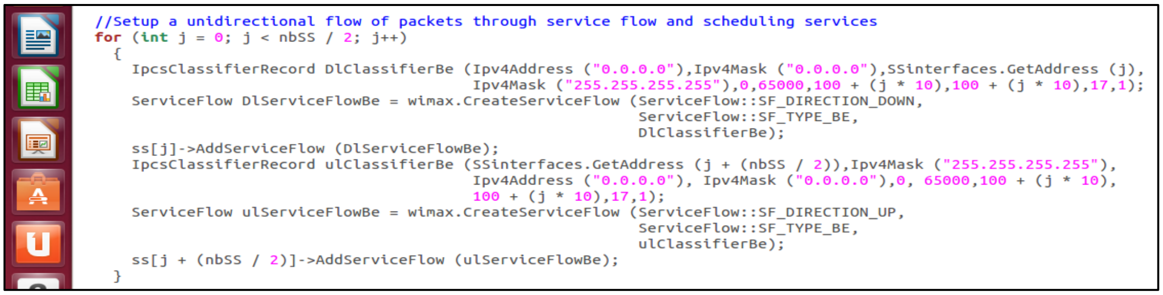
Figure 5.6 also depicts a setup of a unidirectional flow of packets through a service flow type **rtPS** for both the SS DL and BS UP service flows. Similarly, the setup is implemented across all the network nodes i.e. SSs and BS, with the aid of WiMAX classes such as **IpcsClassifierRecord** and **ServiceFlow** was presented under the previous figure, 5.7. In a similar method to the implementation of **UGS** and **rtPS**, the **nrtPS** CoS was implemented as presented in Figure 5.7 below:



```
//Setup a unidirectional flow of packets through service flow and scheduling services
for (int j = 0; j < nbSS / 2; j++)
  {
    IpcsClassifierRecord DlClassifierBe (Ipv4Address ("0.0.0.0"),Ipv4Mask ("0.0.0.0"),SSinterfaces.GetAddress (j),
                                         Ipv4Mask ("255.255.255.255"),0,65000,100 + (j * 10),100 + (j * 10),17,1);
    ServiceFlow DlServiceFlowBe = wimax.CreateServiceFlow (ServiceFlow::SF_DIRECTION_DOWN,
                                                           ServiceFlow::SF_TYPE_NRTPS,
                                                           DlClassifierBe);
    ss[j]->AddServiceFlow (DlServiceFlowBe);
    IpcsClassifierRecord ulClassifierBe (SSinterfaces.GetAddress (j + (nbSS / 2)),Ipv4Mask ("255.255.255.255"),
                                         Ipv4Address ("0.0.0.0"), Ipv4Mask ("0.0.0.0"),0, 65000,100 + (j * 10),
                                         100 + (j * 10),17,1);
    ServiceFlow ulServiceFlowBe = wimax.CreateServiceFlow (ServiceFlow::SF_DIRECTION_UP,
                                                           ServiceFlow::SF_TYPE_NRTPS,
                                                           ulClassifierBe);
    ss[j + (nbSS / 2)]->AddServiceFlow (ulServiceFlowBe);
  }
```

Figure 5.7 Non Real-Time Polling Service (nrtPS) Implementation

Also**,** Figure 5.7 depicts a setup of a unidirectional flow of packets through a service flow type **nrtPS** for both the SS DL and BS UP service flows. Similarly, the setup is implemented across all the network nodes with the aid of WiMAX classes such as **IpcsClassifierRecord** and **ServiceFlow** was presented under the previous figure, 5.7. In a similar method to the implementation of **UGS**, **rtPS** and **nrtPS**, the **BE** CoS was implemented as presented in Figure 5.8 below



```
//Setup a unidirectional flow of packets through service flow and scheduling services
for (int j = 0; j < nbSS / 2; j++)
  {
    IpcsClassifierRecord DlClassifierBe (Ipv4Address ("0.0.0.0"),Ipv4Mask ("0.0.0.0"),SSinterfaces.GetAddress (j),
                                         Ipv4Mask ("255.255.255.255"),0,65000,100 + (j * 10),100 + (j * 10),17,1);
    ServiceFlow DlServiceFlowBe = wimax.CreateServiceFlow (ServiceFlow::SF_DIRECTION_DOWN,
                                                           ServiceFlow::SF_TYPE_BE,
                                                           DlClassifierBe);
    ss[j]->AddServiceFlow (DlServiceFlowBe);
    IpcsClassifierRecord ulClassifierBe (SSinterfaces.GetAddress (j + (nbSS / 2)),Ipv4Mask ("255.255.255.255"),
                                         Ipv4Address ("0.0.0.0"), Ipv4Mask ("0.0.0.0"),0, 65000,100 + (j * 10),
                                         100 + (j * 10),17,1);
    ServiceFlow ulServiceFlowBe = wimax.CreateServiceFlow (ServiceFlow::SF_DIRECTION_UP,
                                                           ServiceFlow::SF_TYPE_BE,
                                                           ulClassifierBe);
    ss[j + (nbSS / 2)]->AddServiceFlow (ulServiceFlowBe);
  }
```

Figure 5.8 Best Effort (BE) Implementation

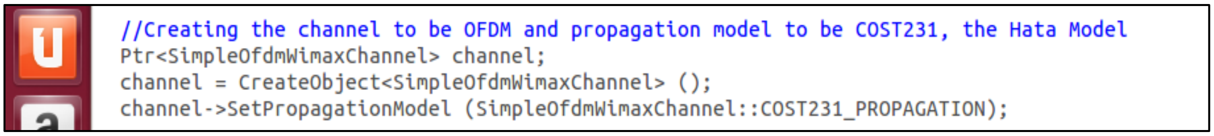
Lastly**,** Figure 5.8 depicts a setup of a unidirectional flow of packets through a service flow type **BE** for both the SS DL and BS UP service flows. Similarly to the implementation of **UGS**, **rtPS** and **nrtPS**, the setup is implemented across all the network nodes with the aid of WiMAX classes such as **IpcsClassifierRecord** and **ServiceFlow** was presented under the previous Figure, 5.8.

5.2.4. Propagation Model

Although **Cost231PropagationLossModel** is said to be applicable to urban areas, it is also said that this model is meant to further evaluate path loss in Suburban or Rural Quasi-Open/Open Area ("Nsnam", 2015) and Dwesa community, as described under the research site/location can be regarded as a Rural Quasi-Open Area. Therefore, this propagation model was implemented for its propagation capabilities of:

- Frequency: 1500 MHz to 2000 MHz
- Mobile Station Antenna Height: 1 up to 10m
- Base station Antenna Height: 30m to 200m
- Link Distance: up to 20 km

Figure 5.9 below presents the implementation of the **Cost231PropagationLossModel**as a set propagation model for the WiMAX channel created using the **SimpleOfdmWimaxChannel** class:



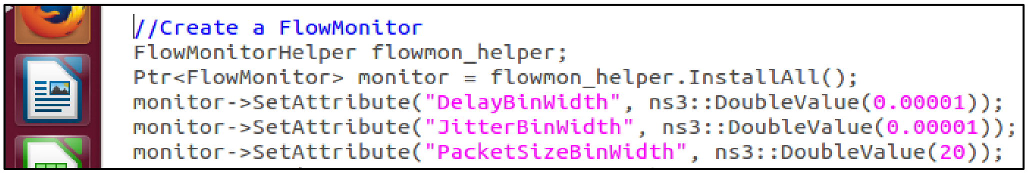
```
//Creating the channel to be OFDM and propagation model to be COST231, the Hata Model
Ptr<SimpleOfdmWimaxChannel> channel;
channel = CreateObject<SimpleOfdmWimaxChannel> ();
channel->SetPropagationModel (SimpleOfdmWimaxChannel::COST231_PROPAGATION);
```

Figure 5.9 Implementation of the SimpleOfdmWimaxChannel and Cost231PropagationLossModel classes

The creation of a channel and its propagation model as presented in Figure 5.9 above paved way for the attachment of the channel to the PHY layer of the devices **ssDevs** and **bsDevs**.

5.2.6. Data Collection and Flow Monitor

The need to collect and store performance data from a simulation emerged and the **FlowMonitor** class was implemented to monitor and report back packet flows and **FlowMonitorHelper** to enable IP flow monitoring on a set of nodes as observed during simulation time ("Nsnam,2015). Figure 5.11 presents the implementation process of the **FlowMonitor** and **FlowMonitorHelper** classes wherein parameters such as **DelayBinWidth,JitterBinWidth** and **PacketSizeBinWidth** were set:



```
//Create a FlowMonitor
FlowMonitorHelper flowmon_helper;
Ptr<FlowMonitor> monitor = flowmon_helper.InstallAll();
monitor->SetAttribute("DelayBinWidth", ns3::DoubleValue(0.00001));
monitor->SetAttribute("JitterBinWidth", ns3::DoubleValue(0.00001));
monitor->SetAttribute("PacketSizeBinWidth", ns3::DoubleValue(20));
```

Figure 5.11 The FlowMonitor and FlowMonitorHelper classes Implementation

The implementation process as depicted in Figure 5.11 above sets the width for **Delay**, **Jitter** and **Packet Size** histograms. The flow monitoring implementation continues and metrics such as Average Delay, Average Jitter, Actual Packets Loss and Throughput, amongst others, presents the network performance as shown in Figure 5.12 below:

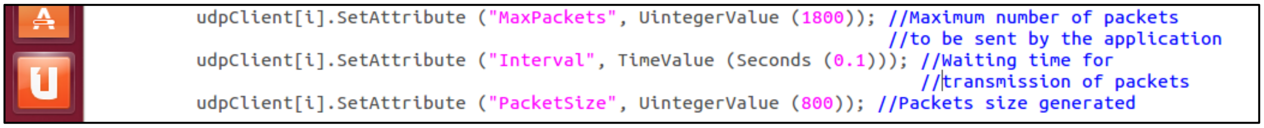
```
 monitor->SerializeToXmlFile("dwesa_results.xml", true, true);
 Ptr<Ipv4FlowClassifier> classifier = DynamicCast<Ipv4FlowClassifier>(flowmon_helper.GetClassifier ());
 std::map<FlowId, FlowMonitor::FlowStats> stats = monitor->GetFlowStats();

 for (std::map<FlowId, FlowMonitor::FlowStats>::const_iterator i =stats.begin (); i != stats.end (); i++)
 {
   Ipv4FlowClassifier::FiveTuple t = classifier->FindFlow (i->first);
   std::cout << "Flow " << i->first << " ( " << t.sourceAddress << " ->" << t.destinationAddress << " )\n";
   std::cout << "Tx Packets: " << i->second.txPackets << "\n";
   std::cout << "Rx Packets: " << i->second.rxPackets << "\n";
   std::cout << "Tx Bytes: " << i->second.txBytes << "\n";
   std::cout << "Rx Bytes: " << i->second.rxBytes << "\n";
   std::cout << "Time first Rx: " << (i->second.timeFirstRxPacket) << "\n";
   std::cout << "Time last Rx: " << (i->second.timeLastRxPacket)<< "\n";
   std::cout << "Port " << i->first << " ( " << t.sourcePort << " ->" << t.destinationPort << ")\n";
   std::cout << "Throughput:    " << (i->second.rxPackets*8.0 / 10.0 / 1024 / 1024)/(double((i->second.timeLastRxPacket - i-
second.timeFirstRxPacket).GetSeconds())) << " Mbps\n";
   std::cout << "Average delay:  " << (i->second.delaySum).GetSeconds()/(double (i->second.rxPackets)) << " s\n";
   std::cout << "Average jitter:  " << (i->second.jitterSum).GetSeconds()/(double (i->second.rxPackets - 1)) <<" s\n";
   std::cout << "Average received packet size:  " << (i->second.rxBytes)/(double (i->second.rxPackets)) << " byte\n";
   std::cout << "FlowMonitor Packets lost:  " << (i->second.lostPackets) << " packets\n";
   std::cout << "Actual Packets lost:  " << (i->second.txPackets -i->second.rxPackets) << " packets\n";
   std::cout << "Actual Packet loss:  " << (i->second.txPackets - i->second.rxPackets)/(double (i->second.txPackets)) << "\n\n";
 }
```

Figure 5.12 FlowMonitor Class Metrics Implementation

The implementation is such that, as depicted in Figure 5.12, the flow monitoring process can identify a flow by an ID (flow), the transmitted number of packets (Tx packets), the received number of packets (Rx Packets), the amount of bytes transmitted (Tx Bytes), the amount of bytes received (Rx Bytes), the transmission time, the port number (Port), throughput (Throughput), Average Jitter and Delay, amongst others.

5.3. Results Presentation and Analysis

This section presents the results generated as a result of several scenarios, up to 12 scenarios that were created, with each scenario examining the WiMAX network performance of a combination of a certain CoS (UGS, rtPS. nrtPS or BE) and WiMAX UL and DL primary scheduler (SIMPLE, RTPS or MBQOS). The UGS combinations with each of the WiMAX UL and DL primary schedulers are presented first, followed by rtPS, nrtPS and BE accordingly. Across all the 12 scenarios, the amount of bandwidth allocated and size of packets for transmission were consistent, as shown in Figure 5.13 below, to ensure that each scenario is implemented and tested on similar conditions and thus warrants a fair analysis of its performance:



```
udpClient[i].SetAttribute ("MaxPackets", UintegerValue (1800)); //Maximum number of packets
                                                                //to be sent by the application
udpClient[i].SetAttribute ("Interval", TimeValue (Seconds (0.1))); //Waiting time for
                                                                   //transmission of packets
udpClient[i].SetAttribute ("PacketSize", UintegerValue (800)); //Packets size generated
```

Figure 5.13 The Maximum Packets Sent and Packet Size Generated

Figure 5.13 above presents the implementation of a client application which sends UDP packets to a maximum of 1800, time to wait between packets being 0.1 seconds and packet size of 800 bytes. It is worth noting that the services provided to Dwesa community through the implementation of WiMAX network are mainly web-based applications which are suitable for TCP. However, the works of Khunjuzwa and Thinyane (2011) paved a way for VoIP application services for this community which can be best offered over UDP as underlying protocol. It is for ideal that UDP as underlying protocol was used to accommodate a worst case scenario of traffic congestion. This implementation was maintained for all the different scenarios and thus has ensured that the network traffic conditions are similar.

5.3.1. Scenario 1: UGS and MBQOS

As presented in Figure 5.13 above, large files were transmitted across the network and bandwidth was allocated for their transmission. Using UGS and MBQOS as the first scenario, the following results, presented in Figures 5.14, 5.15, 5.16 and 5.17 below, were obtained with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput. (Also see **Appendix** A)

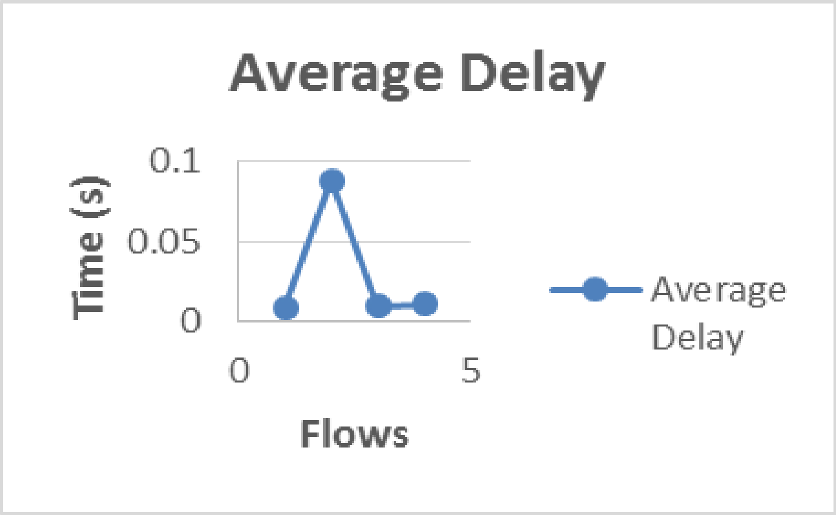


Figure 5.14: Average Delay of UGS and MBQOS Combination

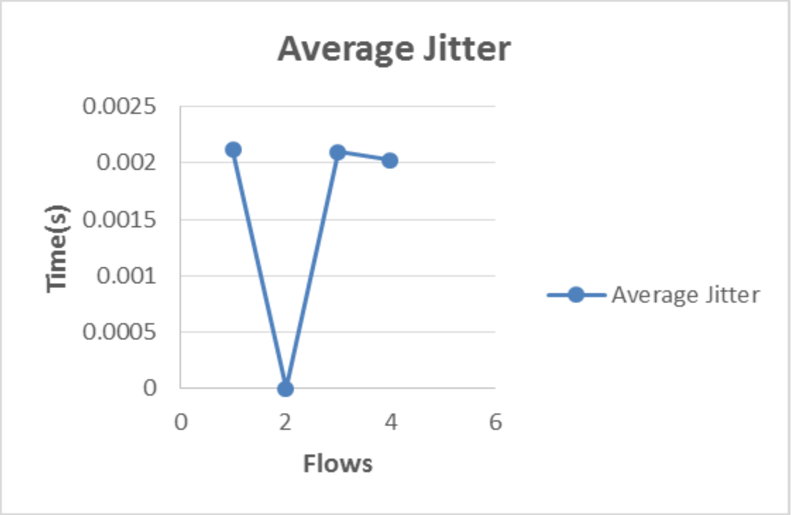


Figure 5.15: Average Jitter of UGS and MBQOS Combination

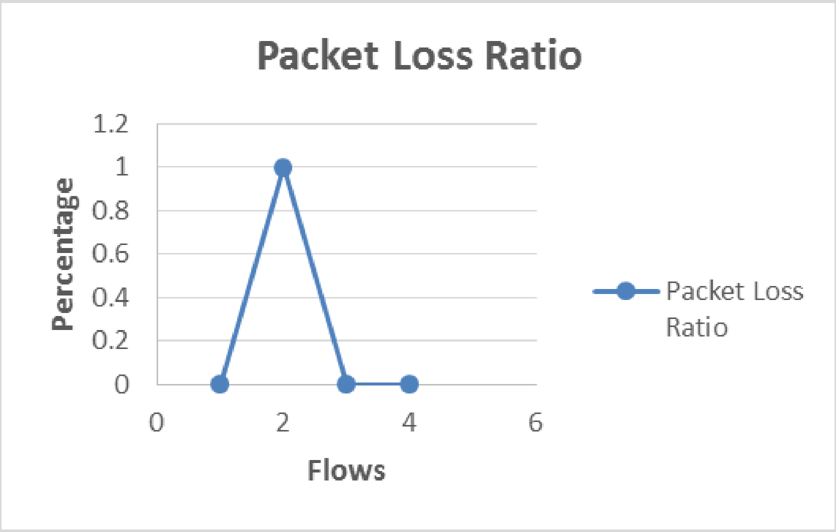


Figure 5.16: PLR of UGS and MBQOS Combination

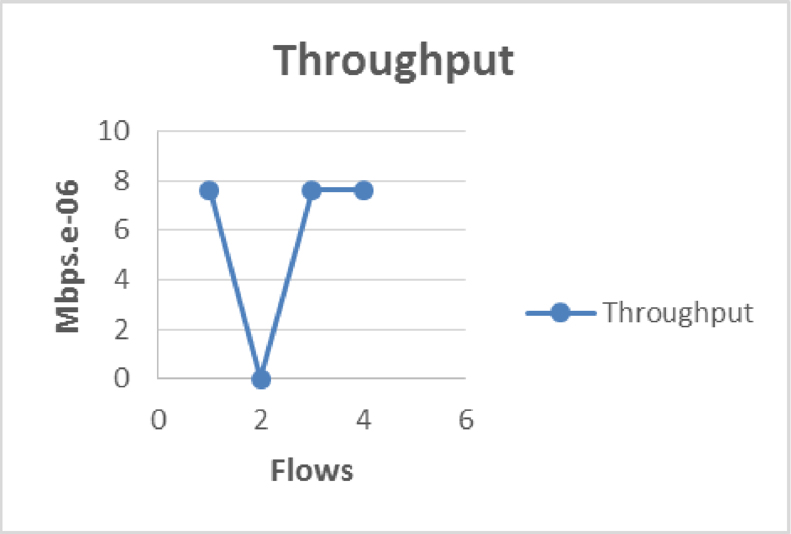


Figure 5.17: Throughput of UGS and MBQOS Combination

A closer look from Figures 5.14, 5.15, 5.16 and 5.17 above is hereby taken to consider the metrics put in place to measure QoS of this simulated WiMAX network. A consideration for all these metrics is given to each flow (1,2,3 and 4) of this scenario. For Scenario 1, the Average Delay generated was reported to have figures of **0.00865201s**, **0.0881707s**, **0.00880161s** and **0.0101704s** for flows 1, 2, 3 and 4, respectively. The smallest Average Delay being **0.00880161s** of flow 3 and the biggest Average Delay being **0.0881707s** of flow 2.

The Average Jitter generated was reported to have figures of **0.00211945s**, **-nan s** *(which in the graphs is represented by a zero (0))*, **0.00209911s**, and **0.00202856s** for flows 1, 2, 3 and 4 respectively. The smallest Average Jitter being **0.00202856s** of flow 4 and the biggest Average Delay being **–nan s** *(which in the graphs is represented by a zero (0))* of flow 2. For Packet Loss Ratio, the figure recorded for flows 1, 2, 3 and 4 were **0, 0.999932, 0** and **0**, respectively. Flows 1, 3, and 4 recorded the smallest PLR of 0 and flow 2 recorded the biggest PLR of 0.999932. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of **7.6299e-06**, **inf Mbps**, **7.6299e-06**, and **7.63043e-06**, respectively. Flows 1 and 3 recorded the lowest Throughput values whereas Flow 2 recorded the highest Throughput values.

5.3.2. Scenario 2: UGS and FCFS

The second scenario saw the implementation of UGS and FCFS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.18, 5.19, 5.20 and 5.21 below with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput.(Also see **Appendix** B)

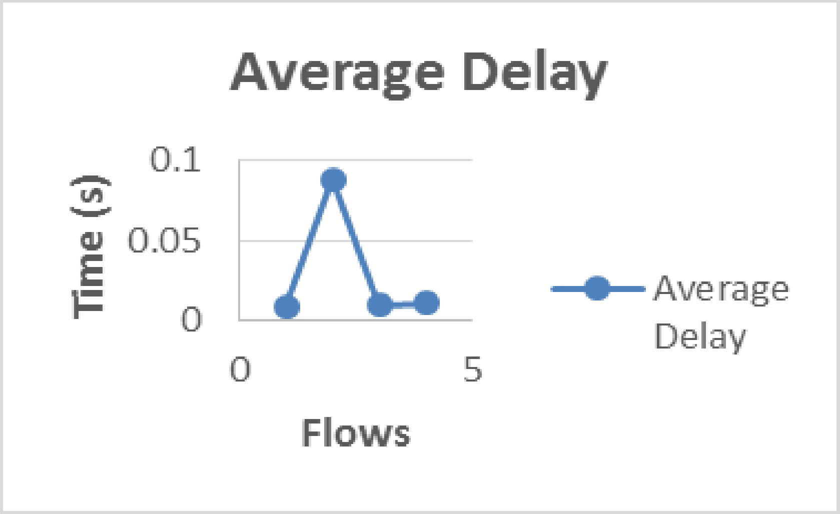


Figure 5.18: Average Delay of UGS and FCFS Combination

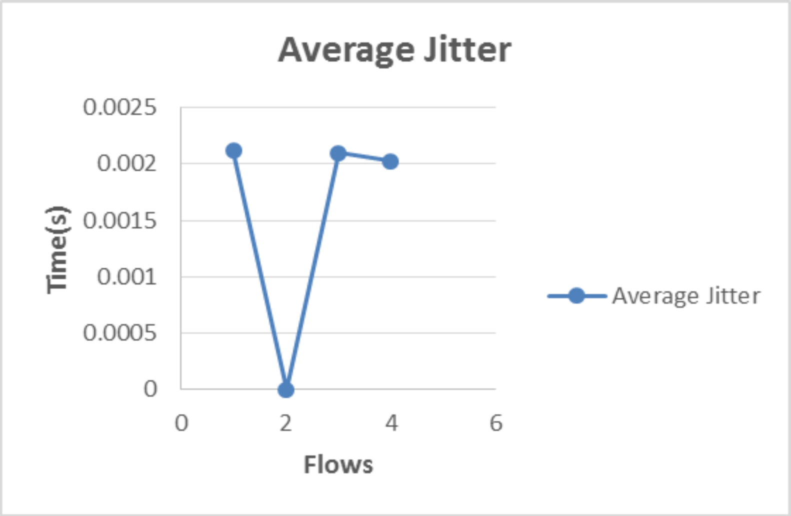


Figure 5.19: Average Jitter of UGS and FCFS Combination

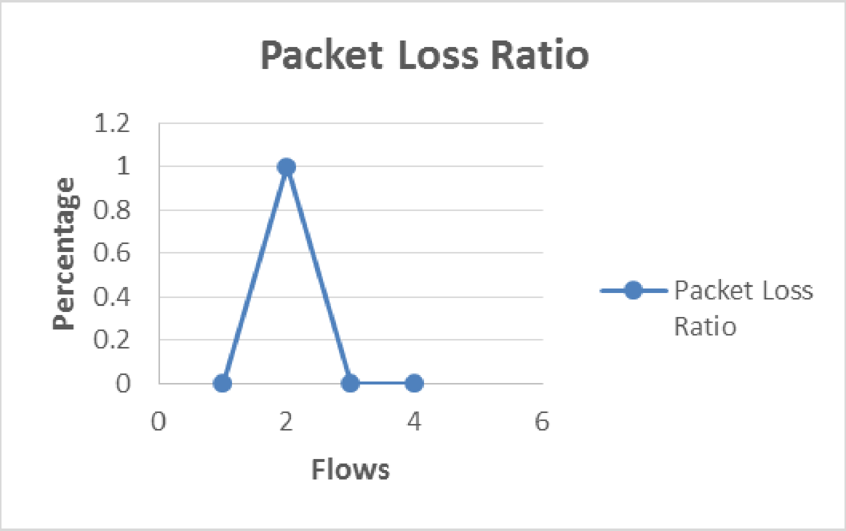


Figure 5.20: Packet Loss Ratio of UGS and FCFS Combination

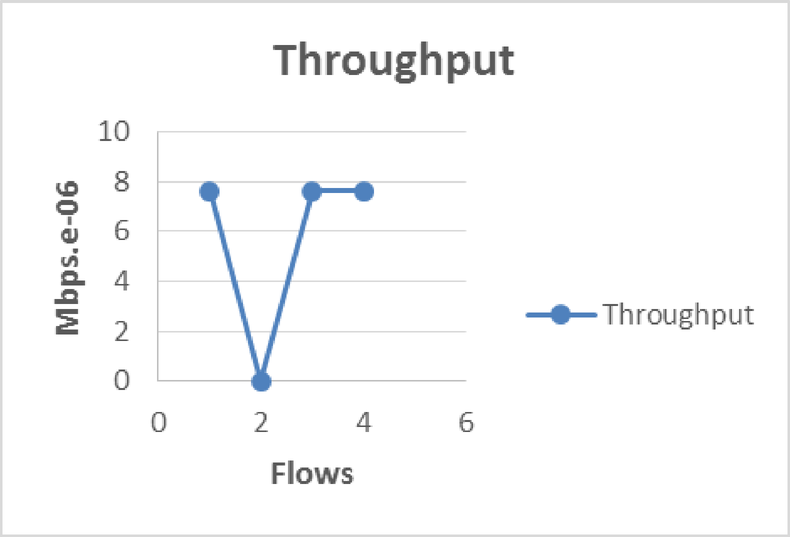


Figure 5.21: Throughput of UGS and FCFS Combination

A closer look from Figures 5.18, 5.19, 5.20 and 5.21 above is hereby taken to consider the metrics put in place to measure QoS of this simulated WiMAX network. A consideration for all these metrics is given to each flow (1, 2 ,3 and 4) of this scenario. For Scenario 1, the Average Delay generated was reported to have figures of **0.00835201s**, **0.0881707s**, **0.00880161s** and **0.0101704s** for flows 1, 2, 3 and 4, respectively. The smallest Average Delay being **0.00835201s** of flow 1 and the biggest Average Delay being **0.0881707s** of flow 2.

The Average Jitter generated was reported to have figures of **0.00211945s**, **-nan s**, **0.00209911s**, and **0.00202856s** for flows 1, 2, 3 and 4 respectively. The smallest Average Jitter being **0.00202856s** of flow 4 and the biggest Average Delay being **–nan s** of flow 2. For Packet Loss Ratio, the figure recorded for flows 1, 2, 3 and 4 were 0, 0.999932, 0 and 0 respectively. Flows 1, 3, and 4 recorded the smallest PLR of 0 and flow 2 recorded the biggest PLR of 0.999932. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of **7.6299e-06**, **inf Mbps**, **7.6299e-06**, and **7.63043e-06,** respectively. Flows 1 and 3 recorded the lowest Throughput values whereas Flow 2 recorded the highest Throughput values.

5.3.3. Scenario 3: UGS and RTPS

The third scenario saw the implementation of UGS and RTPS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.22, 5.23, 5.23 and 5.24 below with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput.(Also see **Appendix** C)

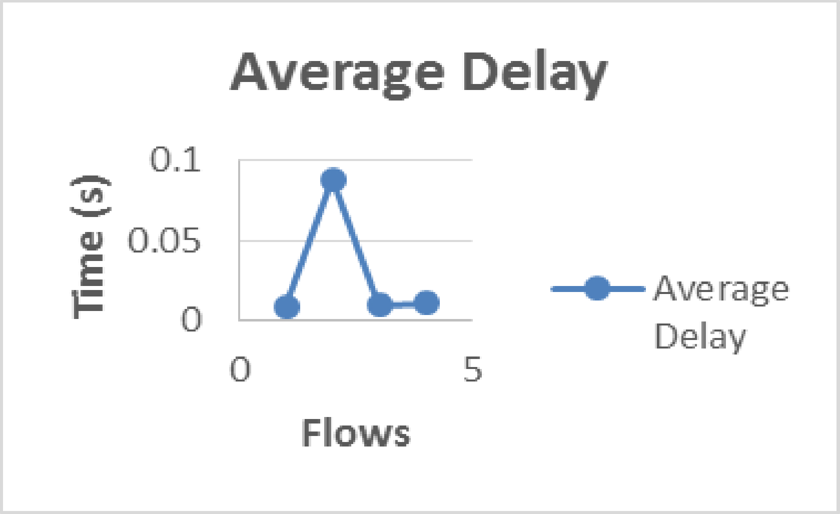


Figure 5.22: Average Delay of UGS and RTPS Combination

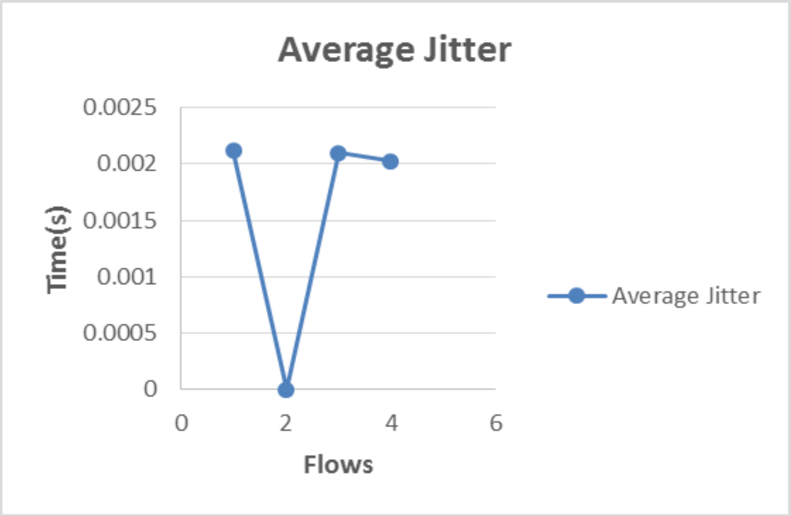


Figure 5.23: Average Jitter of UGS and RTPS Combination

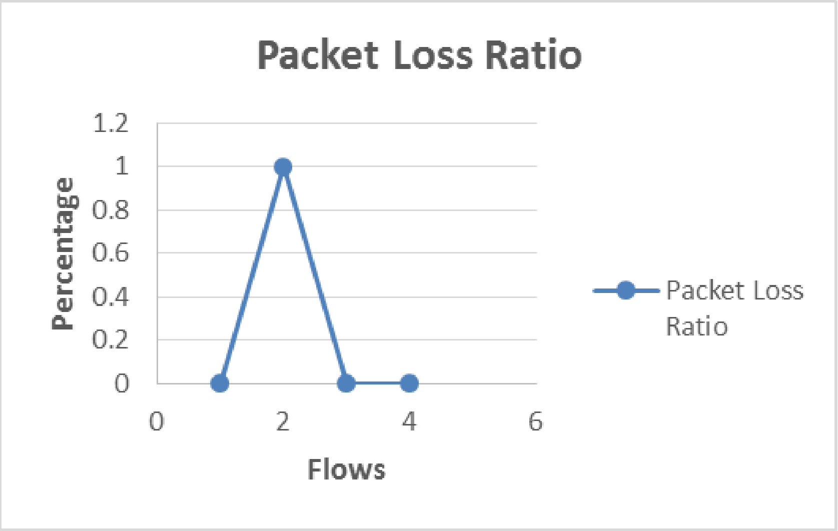


Figure 5.24: Packet Loss Ratio of UGS and RTPS Combination

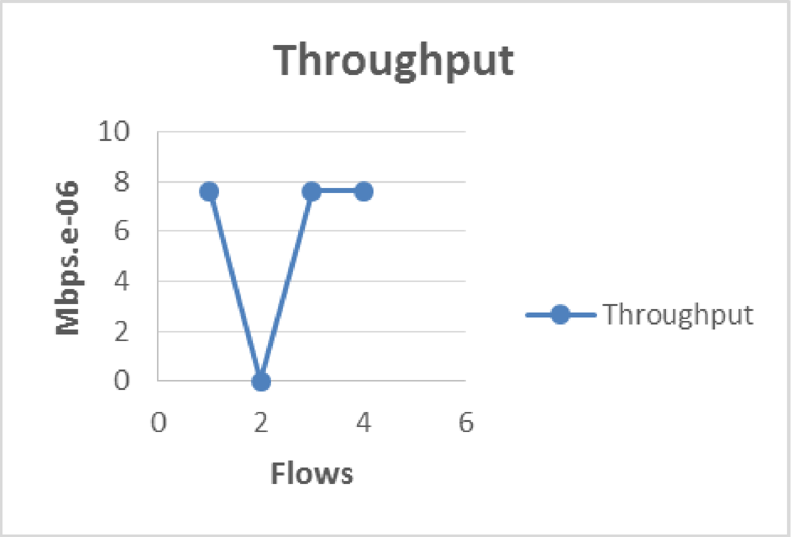


Figure 5.25: Throughput of UGS and RTPS Combination

A closer look from Figures 5.22, 5.23, 5.23 and 5.24 above is hereby taken to consider the metrics put in place to measure QoS of this simulated WiMAX network. A consideration for all these metrics is given to each flow (1,2,3 and 4) of this scenario. For Scenario 1, the Average Delay generated was reported to have figures of **0.00865201s**, **0.0881707s**, **0.00880161s** and **0.0101704s** for flows 1, 2, 3 and 4, respectively. The smallest Average Delay being **0.00880161s** of flow 3 and the biggest Average Delay being **0.0881707s** of flow 2.

The Average Jitter generated was reported to have figures of **0.00211945s**, **-nan s**, **0.00209911s**, and **0.00202856s** for flows 1, 2, 3 and 4, respectively. The smallest Average Jitter being **0.00202856s** of flow 4 and the biggest Average Delay being –**nan s** of flow 2. For Packet Loss Ratio, the figure recorded for flows 1, 2, 3 and 4 were **0, 0.999932, 0** and **0**, respectively. Flows 1, 3, and 4 recorded the smallest PLR of 0 and flow 2 recorded the biggest PLR of 0.999932. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of **7.6299e-06**, **inf Mbps**, **7.6299e-06**, and **7.63043e-06** respectively. Flows 1 and 3 recorded the lowest Throughput values whereas Flow 2 recorded the highest Throughput values.

5.3.4. Scenario 4: RTPS and MBQOS

The fourth scenario saw the implementation of RTPS and MBQOS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.26, 5.27, 5.28 and 5.29 belowwith a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput. (Also see **Appendix** D)

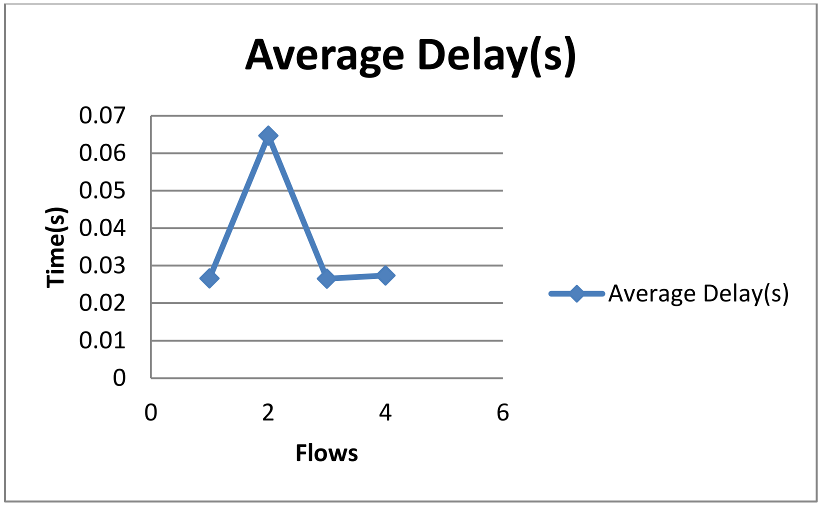


Figure5.26:Ave. Delay of RTPS and MBQOS Combination

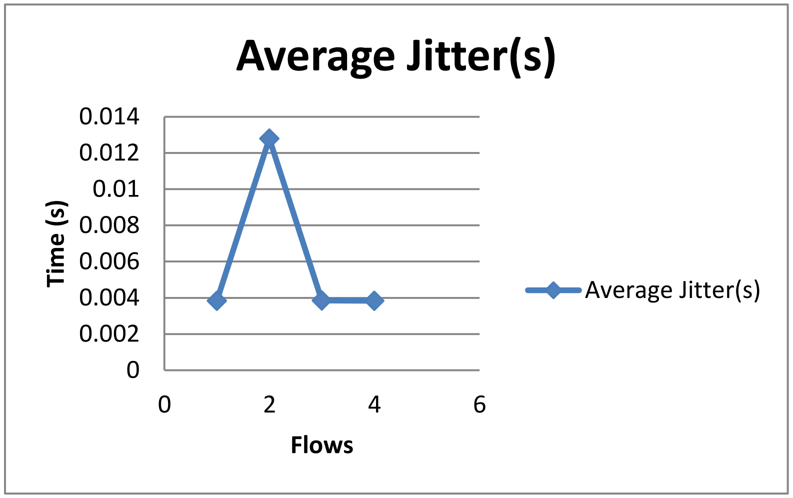


Figure 5.27: Ave. Jitter of RTPS and MBQOS Combination

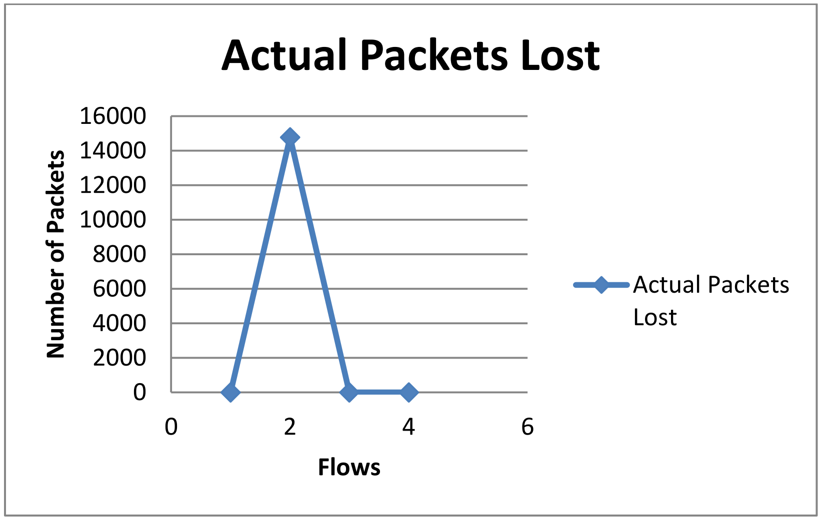


Figure5.28: PLR of RTPS and MBQOS Combination

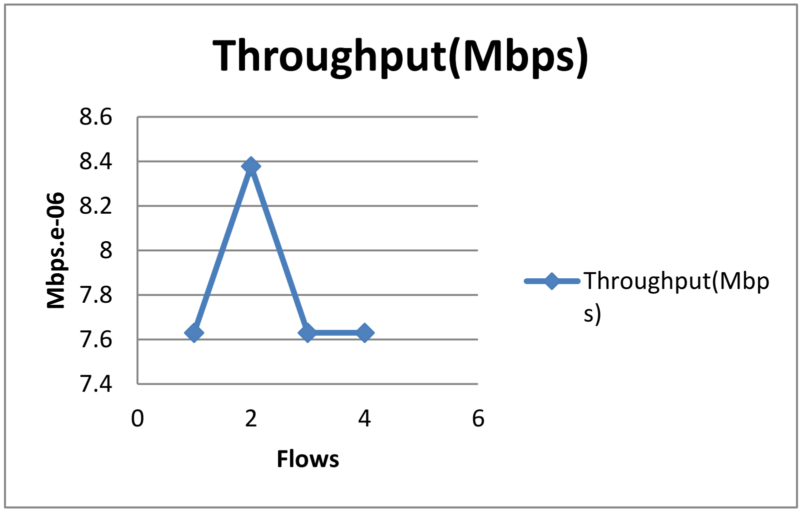


Figure5.29: Throughput of RTPS and MBQOS Combination

Figures 5.26, 5.27, 5.28 and 5.29 above details the unique network performances as a result of the fourth scenario underlying WiMAX network combination factors. Similarly, a consideration for all these metrics is given to each flow (1, 2, 3 and 4) of this scenario. For Scenario 4, the Average Delay generated was reported to have figures of **0.0266159s**, **0.064642s**, **0.0265192s** and **0.0273927s** for flows 1, 2, 3 and 4, respectively. The smallest Average Delay being **0.0265192s** of flow 3 and the biggest Average Delay being **0.064642s** of flow 2.

The Average Jitter generated was reported to have figures of **0.00383462s**, **0.0127833s**, **0.00385171s**, and **0.00382415s** for flows 1, 2, 3 and 4, respectively. The smallest Average Jitter being **0.00382415s** of flow 4 and the biggest Average Jitter being **0.0127833s** of flow 2. For Packet Loss Ratio, the figures recorded were **0**, **0.998649**, **0** and **0** for flows 1, 2, 3 and 4, respectively. Flows 1, 3, and 4 recorded the least PLR of **0** while flow 2 recorded **0.998649** as the highest PLR. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of **7.62996e-06**, **8.3781e-06**, **7.62996e-06**, and **7.63011e-06,** respectively. Flows 1 and 3 recorded the lowest Throughput values of **7.62996e-06** each whereas Flow 2 recorded the highest Throughput value of **8.3781e-06**.

5.3.5. Scenario 5: RTPS and FCFS

The fifth scenario saw the implementation of RTPS and FCFS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.30, 5.31, 5.32 and 5.33 below with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput.(Also see **Appendix** E)

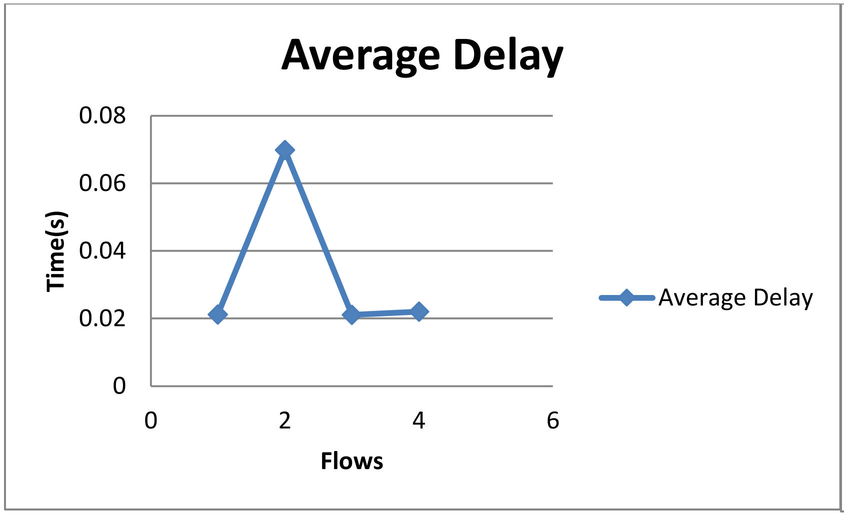


Figure5.30: Average Delay of RTPS and FCFS Combination

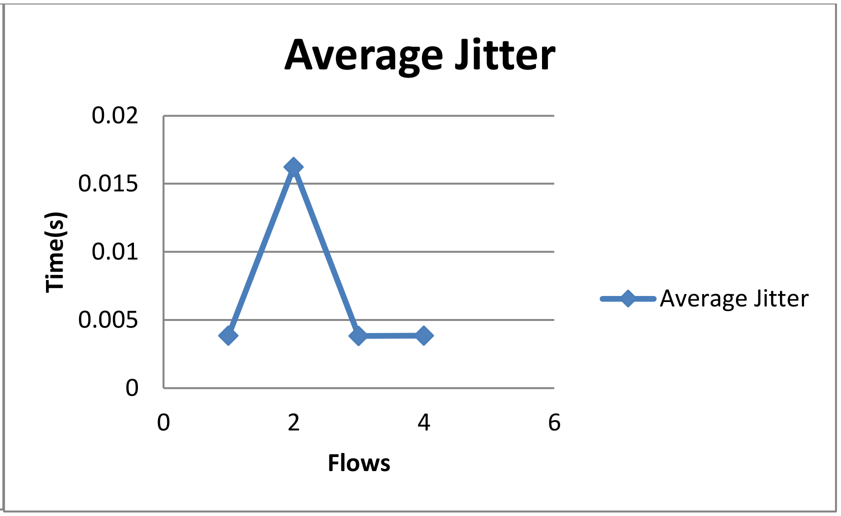


Figure 5.31: Average Jitter of RTPS and FCFS Combination

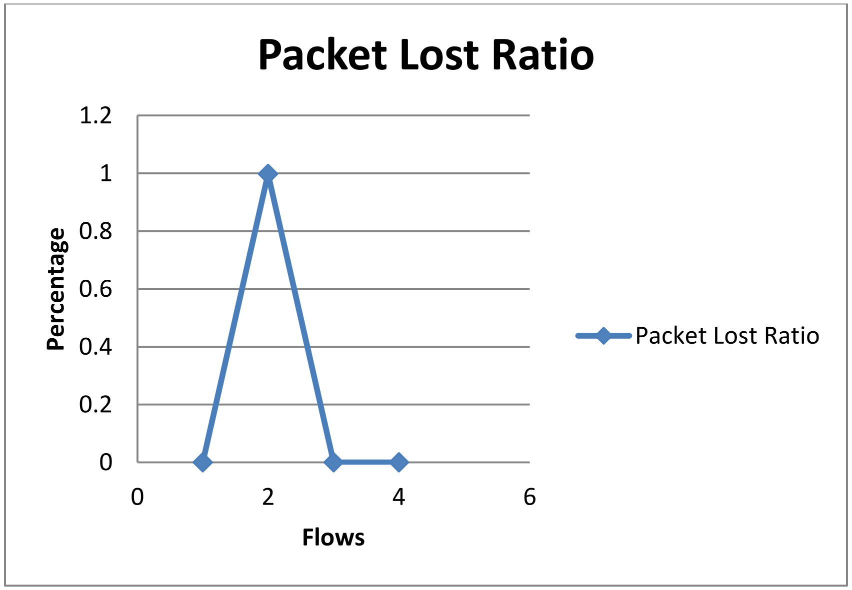


Figure5.32: PLR of RTPS and FCFS Combination

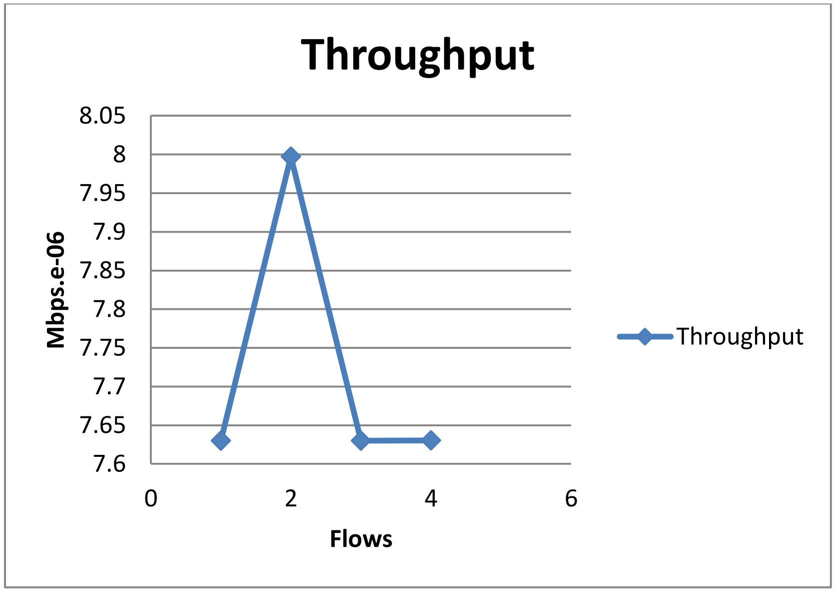


Figure5.33: Throughput of RTPS and FCFS Combination

Using a similar approach to analyze Figures 5.30, 5.31, 5.32 and 5.33 above, a consideration for all the metrics is given to each flow (1, 2, 3 and 4) of this scenario. For this scenario, the Average Delay generated was reported to have figures of **0.021204s**, **0.0698529s**, **0.0210928s** and **0.0219935s** for flows 1, 2, 3 and 4, respectively. The smallest Average Delay being **0.0210928s** of flow 3 and the biggest Average Delay being **0.0698529s** of flow 2.

The Average Jitter generated was reported to have figures of **0.00385276s**, **0.0162272s**, **0.00383745s**, and **0.00384151s** for flows 1, 2, 3 and 4, respectively. The smallest Average Jitter being **0.00383745s** of flow 3 and the biggest Average Jitter being **0.0162272s** of flow 2. For Packet Loss Ratio, the figures recorded were **0**, **0.998581**, **0** and **0** for flows 1, 2, 3 and 4, respectively. Flows 1, 3, and 4 recorded the least PLR of **0** while flow 2 recorded **0.998581** as the highest PLR. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of **7.62991e-06**, **7.99736e-06**, **7.629961e-06**, and **7.63042e-06**, respectively. Flows 1 and 3 recorded the lowest Throughput values of **7.62991e-06** each whereas Flow 2 recorded the highest Throughput value of **7.99736e-06**.

5.3.6. Scenario 6: RTPS and RTPS

The sixth scenario saw the implementation of RTPS and RTPS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.34, 5.35, 5.36 and 5.37 belowwith a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput. (Also see **Appendix** F)

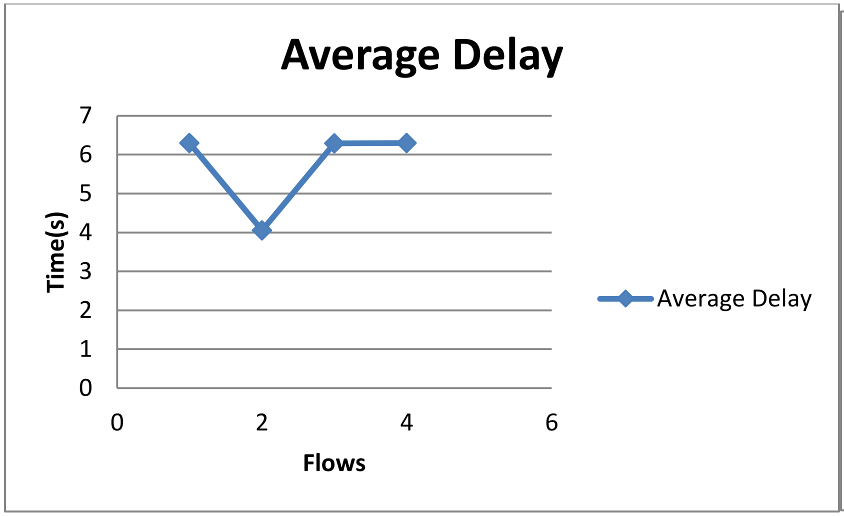


Figure5.34: Average Delay of RTPS and RTPS Combination

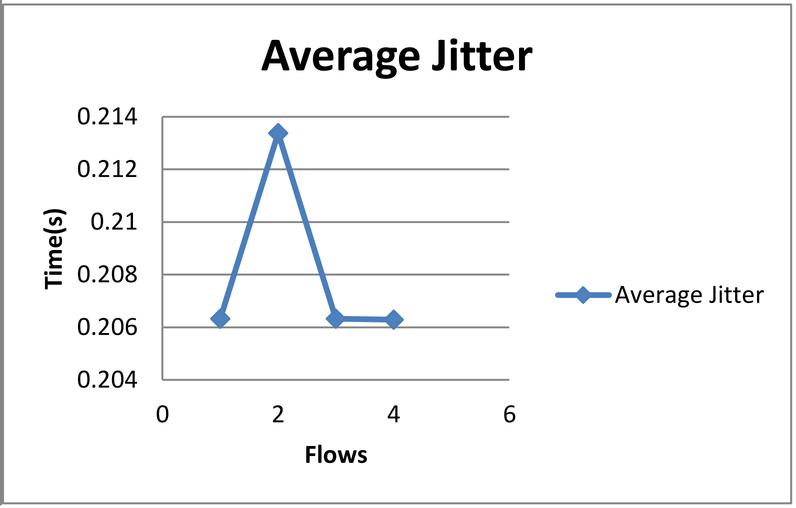


Figure 5.35: Average Jitter of RTPS and RTPS Combination

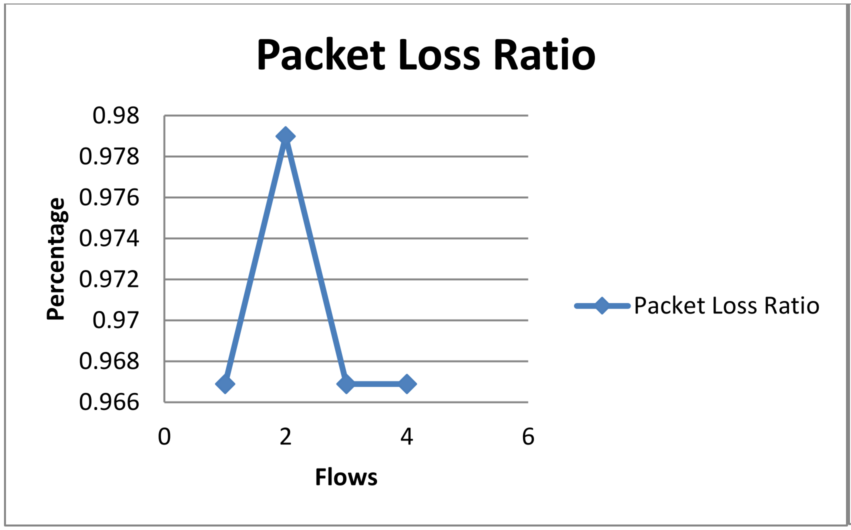


Figure5.36: PLR of RTPS and RTPS Combination

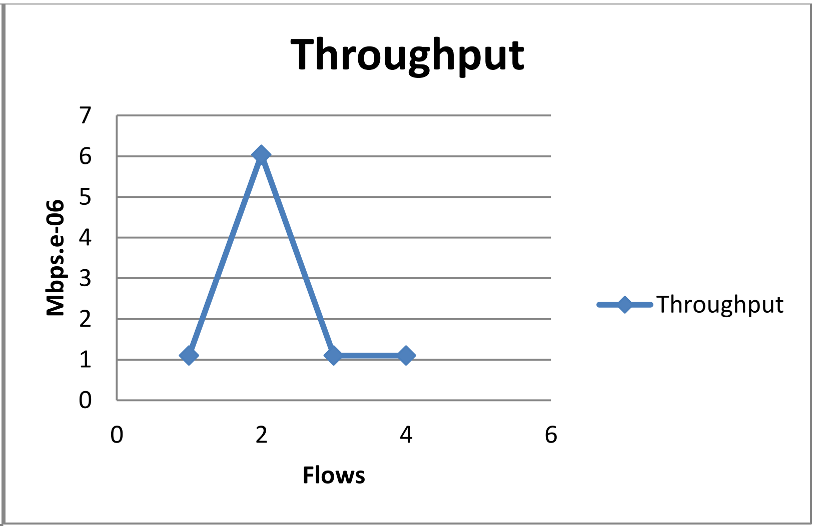


Figure5.37: Throughput of RTPS and RTPS Combination

A similar approach for analyzing Figures 5.34, 5.35, 5.36 and 5.37 above, a consideration for all the metrics is given to each flow (1, 2, 3 and 4) of this scenario. For this scenario, the Average Delay generated was reported to have figures of **6.29477s**, **4.05401s**, **6.29365s** and **6.29594s** for flows 1, 2, 3 and 4, respectively. The smallest Average Delay being **4.05401s**of flow 2 and the biggest Average Delay being **6.29594s** of flow 4.

The Average Jitter generated was reported to have figures of **0.00385276s**, **0.0162272s**, **0.00383745s**, and **0.00384151s** for flows 1, 2, 3 and 4, respectively. The smallest Average Jitter being **0.00383745s** of flow 3 and the biggest Average Jitter being **0.0162272s** of flow 2. For Packet Loss Ratio, the figures recorded were **0**, **0.998581**, **0** and **0** for flows 1, 2, 3 and 4, respectively. Flows 1, 3, and 4 recorded the least PLR of **0** while flow 2 recorded **0.998581** as the highest PLR. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of **7.62991e-06**, **7.99736e-06**, **7.629961e-06**, and **7.63042e-06,** respectively. Flows 1 and 3 recorded the lowest Throughput values of **7.62991e-06** each whereas Flow 2 recorded the highest Throughput value of **7.99736e-06**.

5.3.7. Scenario 7: NRTPS and MBQOS

The seventh scenario saw the implementation of NRTPS and MBQOS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.38, 5.39, 5.40 and 5.41 below with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput.(Also see **Appendix** G)

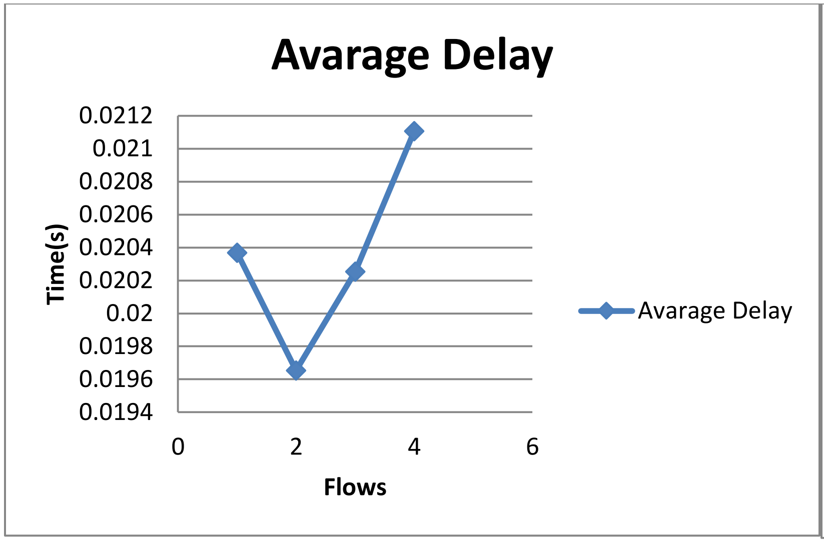


Figure 5.38: Ave. Delay of NRTPS and MBQOS Combination

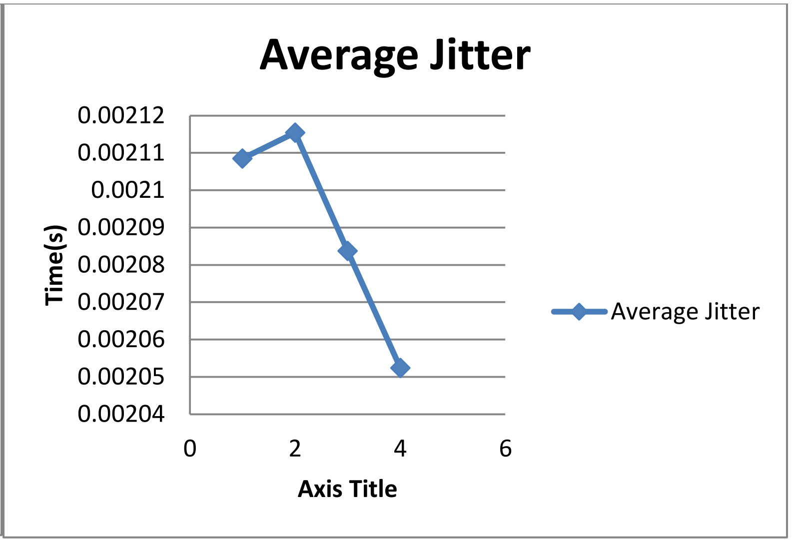


Figure 5.39: Ave. Jitter of NRTPS & MBQOS Combination

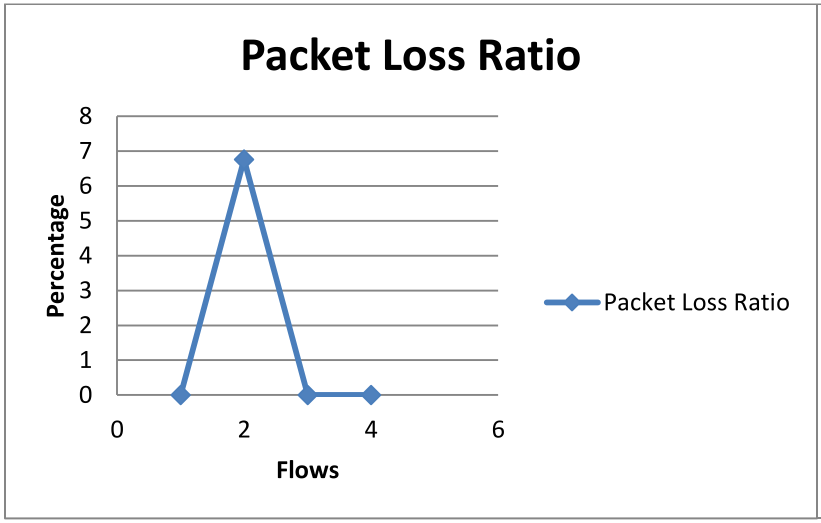


Figure 5.40: PLR of NRTPS and MBQOS Combination

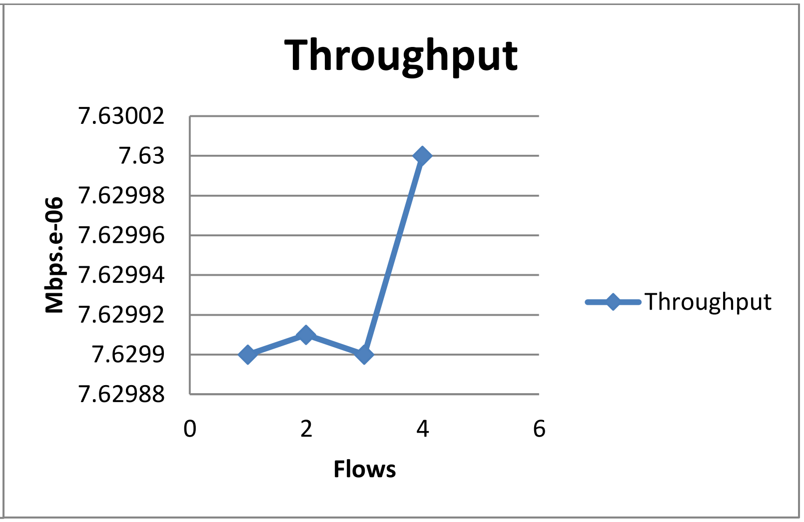


Figure5.41: Throughput of NRTPS and MBQOS Combination

Once more, a similar approach for analyzing Figures 5.38, 5.39, 5.40 and 5.41 above, a consideration for all the metrics is given to each flow (1, 2, 3 and 4) of Scenario 7. Scenario 7, recorded the figures of 0.0203682s, 0.0196527s, 0.0202536s and 0.0211068 for flows 1, 2, 3 and 4, respectively for Average Delay. The smallest Average Delay being 0.0196527s of flow 2 and the biggest Average Delay being 0.0211068s of flow 4.

For the Average Jitter, figures 0.00210847s, 0.0021154s, 0.0020837s, and 0.00205239s for flows 1, 2, 3 and 4, respectively were generated. The smallest Average Jitter being 0.00205239s of flow 4 and the biggest Average Jitter being 0.0021154s of flow 2. The Packet Loss Ratio, recorded figures of 0, 6.75676e-06, 0 and 0 for flows 1, 2, 3 and 4, respectively. Flows 1, 3, and 4 recorded the least PLR of 0 while flow 2 recorded 6.7567e06 as the highest PLR. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of 7.6299e-06, 7.62991e-06, 7.6299e-06, and 7.63e-06, respectively. Flows 1 and 3 recorded the lowest Throughput values of 7.62991e-06 each whereas Flow 4 recorded the highest Throughput value of 7.63e-06.

5.3.8. Scenario 8: NRTPS and FCFS

The eighth scenario saw the implementation of NRTPS and FCFS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.42, 5.43, 5.44 and 5.45 below with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput.(Also see **Appendix** H)

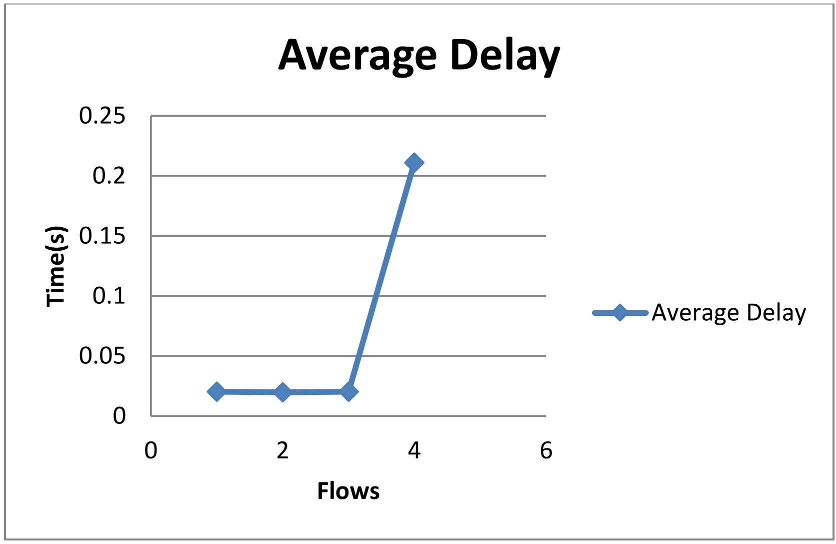


Figure5.42: Average Delay of NRTPS and FCFS Combination

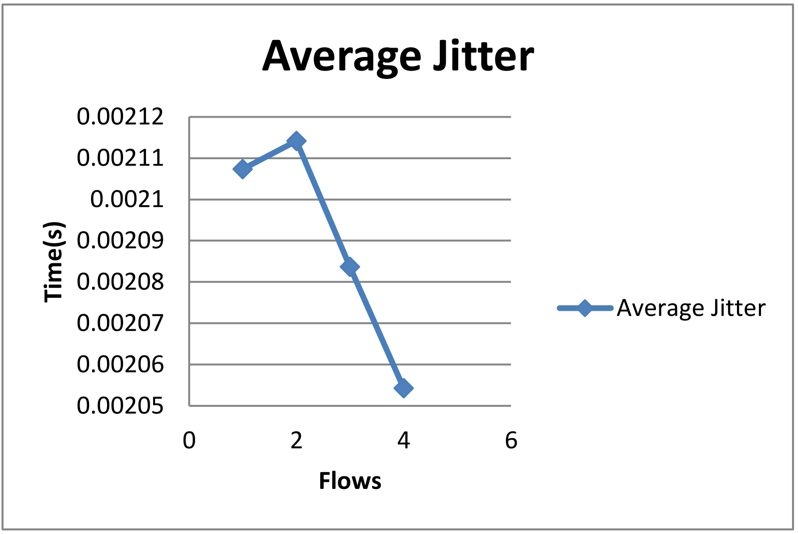


Figure 5.43: Average Jitter of NRTPS and FCFS Combination

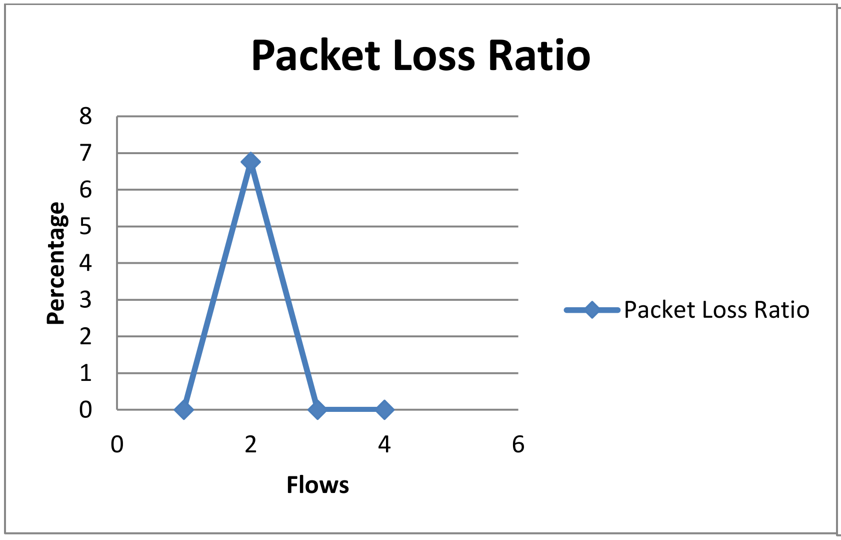


Figure 5.44: PLR of NRTPS and FCFS Combination

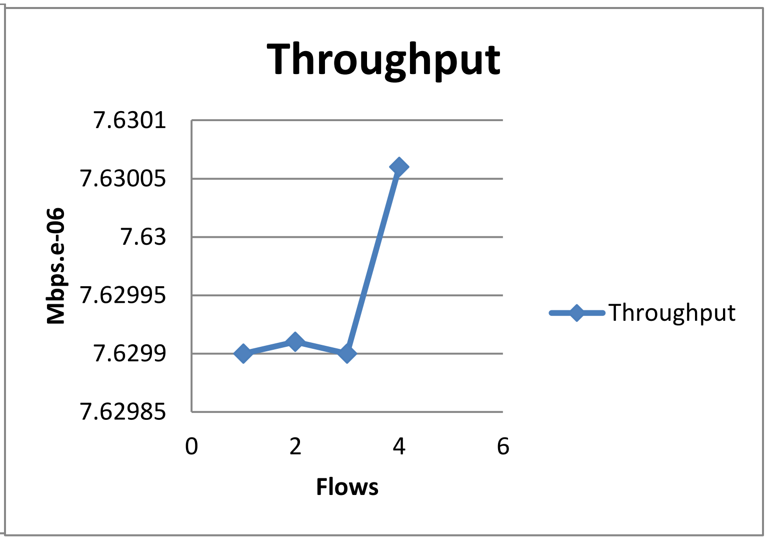


Figure 5.45: Throughput of NRTPS and FCFS Combination

Similarly, in analyzing Figures 5.42, 5.43, 5.44 and 5.45 above, a consideration for all the metrics is given to each flow (1, 2 ,3 and 4) of Scenario 8. For this scenario, figures of **0.0203652s, 0.0196505s, 0.020252s and 0.0211053** for flows 1, 2, 3 and 4, respectively were recorded for Average Delay.The smallest Average Delay being **0.0196505s** of flow 2 and the biggest Average Delay being **0.0211053s** of flow 4**.**

The Average Jitter generated was reported to have figures of **0.00210732s, 0.00211417s, 0.0020837s,** and **0.00205429s** for flows 1, 2, 3 and 4, respectively. The smallest Average Jitter being 0.00205429s of flow **4** and the biggest Average Jitter being 0.**00**2**11417**s of flow 2.For Packet Loss Ratio, the figures recorded were **0**, **6.75676e-05**, **0** and **0** for flows 1, 2, 3 and 4, respectively. Flows 1, 3, and 4 recorded the least PLR of **0** while flow 2 recorded **6.75676e-05** as the highest PLR. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of **7.6299e-06**, **7.62991e-06**, **7.6299e-06**, and **7.63006e-06,** respectively. Flows 1 and 3 recorded the lowest Throughput values of **7.6299e-06** each whereas Flow 4 recorded the highest Throughput value of **7.63006e-06**.

5.3.9. Scenario 9: NRTPS and RTPS

The ninth scenario saw the implementation of NRTPS and RTPS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.46, 5.47, 5.48 and 5.49 below with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput.(Also see **Appendix** I)

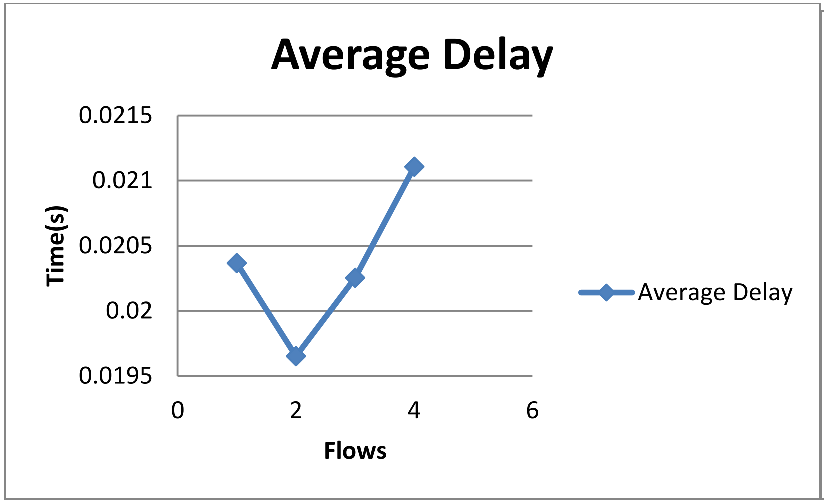


Figure 5.46: Ave. Delay of NRTPS and RTPS Combination

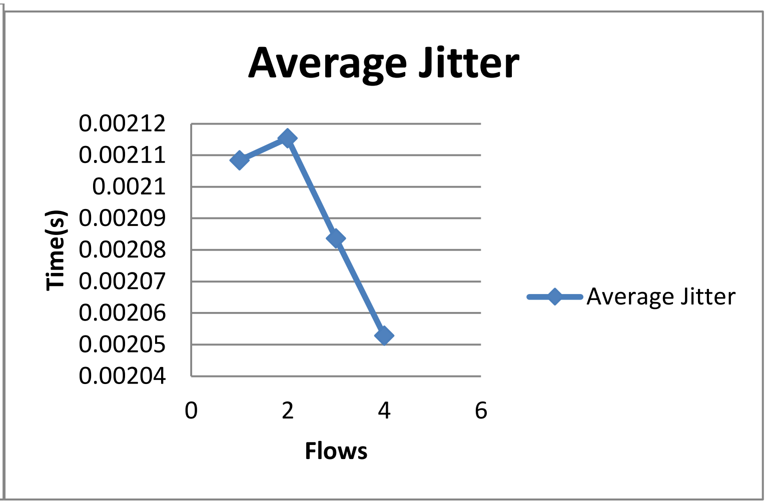


Figure 5.47: Average Jitter of NRTPS and RTPS Combination

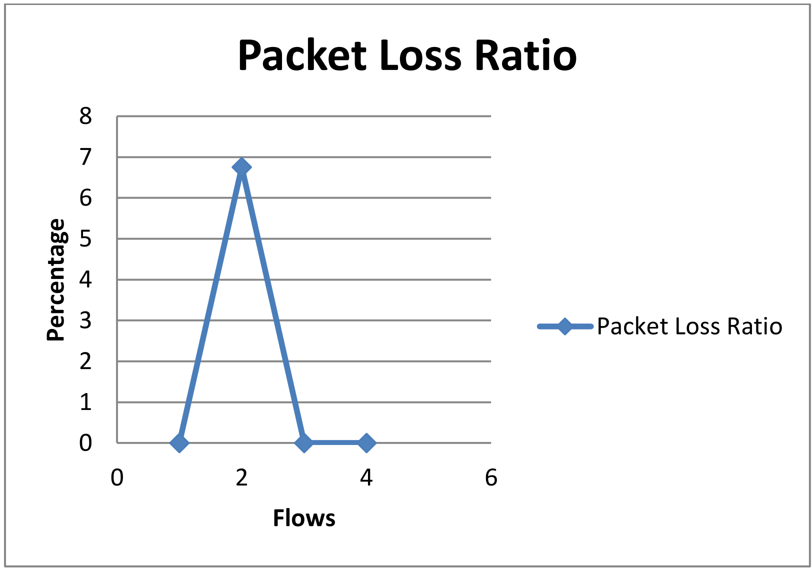


Figure 5.48: PLR of NRTPS and RTPS Combination

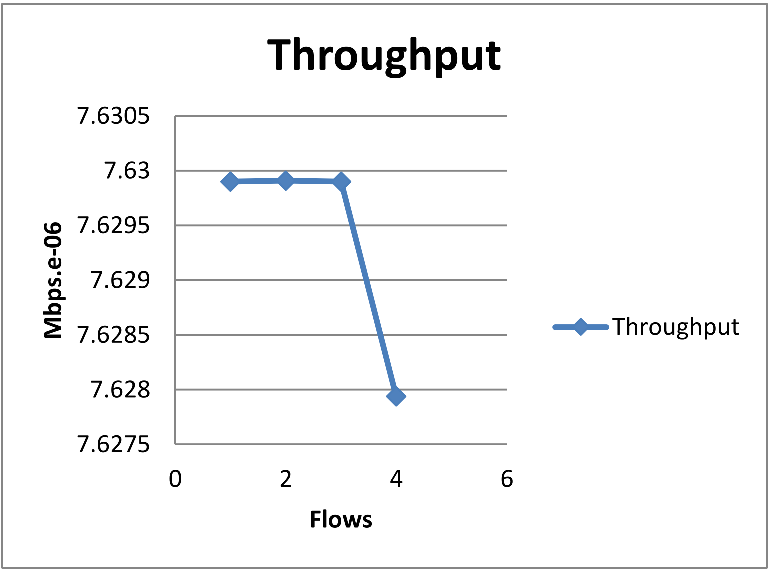


Figure 5.49: Throughput of NRTPS and RTPS Combination

Again, a similar approach for analyzing Figures 5.46, 5.47, 5.48 and 5.49 above, and consideration for all the metrics are given to each flow (1, 2, 3 and 4) of Scenario 9. This scenario recorded the figures of 0.0203674s, 0.019652s, 0.0202528s and 0.0211052 for flows 1, 2, 3 and 4, respectively for Average Delay. The smallest Average Delay being 0.019652s of flow 2 and the biggest Average Delay being 0.0211052s of flow 4.

For the Average Jitter, figures 0.00210845s, 0.00211537s, 0.0020837s, and 0.00205281s for flows 1, 2, 3 and 4, respectively we generated. The smallest Average Jitter being 0.00205281s of flow 4 and the biggest Average Jitter being 0.00211537s of flow 2. The Packet Loss Ratio, recorded figures of 0, 6.75676e-06, 0 and 0.00027027 for flows 1, 2, 3 and 4, respectively. Flows 1 and 3 recorded the least PLR of 0 while flow 2 recorded 6.7567e06 as the highest PLR. It is vitally important to note that the PLR recorded for flow 4 under this scenario was no longer Zero (0) as it was the case previously. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of 7.6299e-06, 7.62991e-06, 7.6299e-06, and 7.63e-06, respectively. Flows 1 and 3 recorded the lowest Throughput values of 7.62991e-06 each whereas Flow 4 recorded the highest Throughput value of 7.63e-06.

5.3.10. *Scenario* 10: BE and MBQO*S*

The tenth scenario saw the implementation of BE and MBQOS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.50, 5.51, 5.52 and 5.53 below with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput.(Also see **Appendix** J)

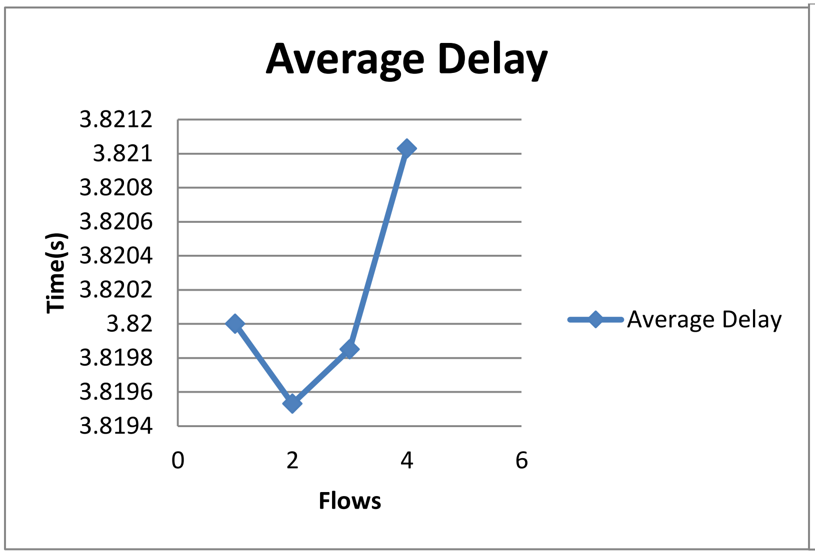


Figure 5.50: Ave. Delay of BE and MBQOS Combination

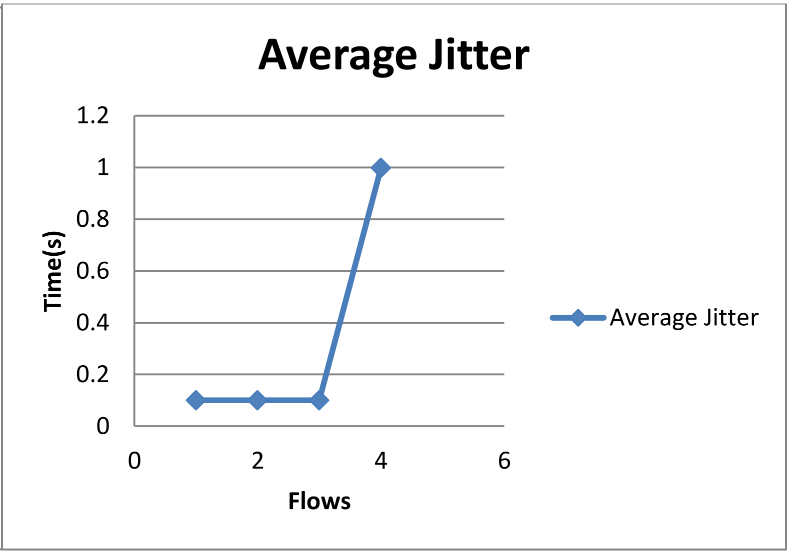


Figure 5.51: Average Jitter of BE and MBQOS Combination

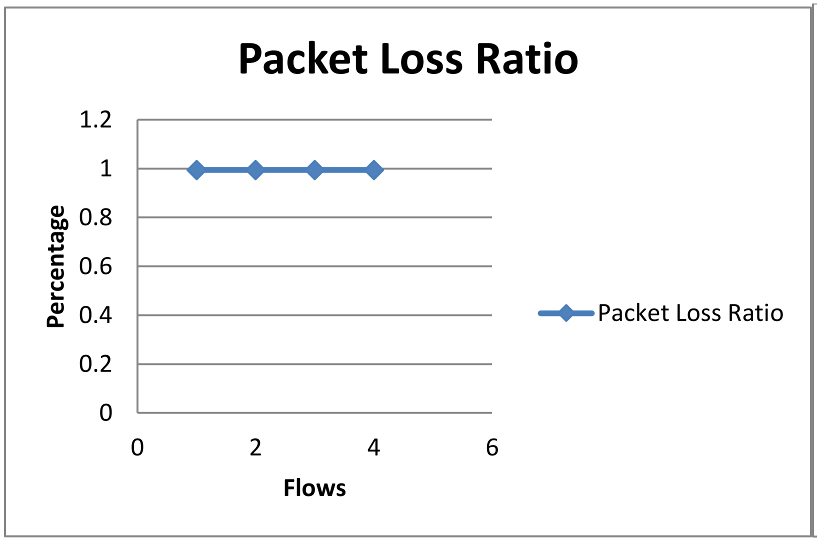


Figure 5.52: Average Jitter of BE and MBQOS Combination

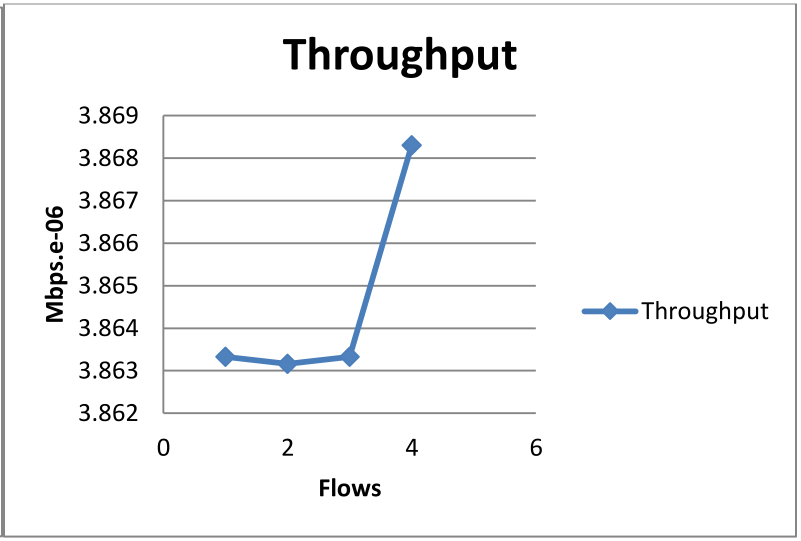


Figure 5.53: Throughput of BE and MBQOS Combination

Analyzing Figures 5.50, 5.51, 5.52 and 5.53 above, a similar approach was taken and a consideration for all the metrics is given to each flow (1, 2, 3 and 4) of Scenario 10. This scenario recorded the figures of 3.8s, 3.81953s, 3.81985s and 3.82103s for flows 1, 2, 3 and 4, respectively for Average Delay. The smallest Average Delay being 3.8s of flow 1 and the biggest Average Delay being 3.82103s of flow 4.

For the Average Jitter, figures 0.100081s, 0.10009s, 0.100081s, and 0.0998238s for flows 1, 2, 3 and 4, respectively, were generated. The smallest Average Jitter being 0.0998238s of flow 4 and the biggest Average Jitter being 0.10009s of flow 2. The Packet Loss Ratio, recorded figures of 0.994797, 0.994797, 0.994797 and 0.994797 for flows 1, 2, 3 and 4, respectively. For this scenario, there was no smallest or biggest PLR recorded as the PLR was consistent at 0.994797 in all the flows. Lastly, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of 3.86333e-06, 3.86316e-06, 3.86333e-06, and 3.8683e-06, respectively. Flow2 recorded the lowest Throughput values of 3.86316e-06 each whereas Flow 4 recorded the highest Throughput value of 3.8683e-06.

5.3.11. Scenario 11: BE and FCFS

The eleventh scenario saw the implementation of BE and FCFS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.54, 5.55, 5.56 and 5.57 below with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput. (Also see **Appendix** K)

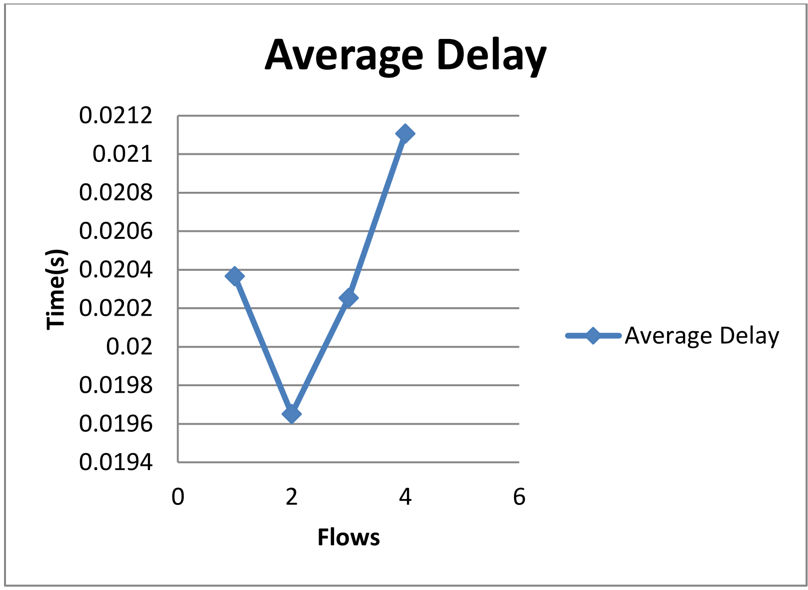


Figure 5.54: Ave. Delay of BE and FCFS Combination

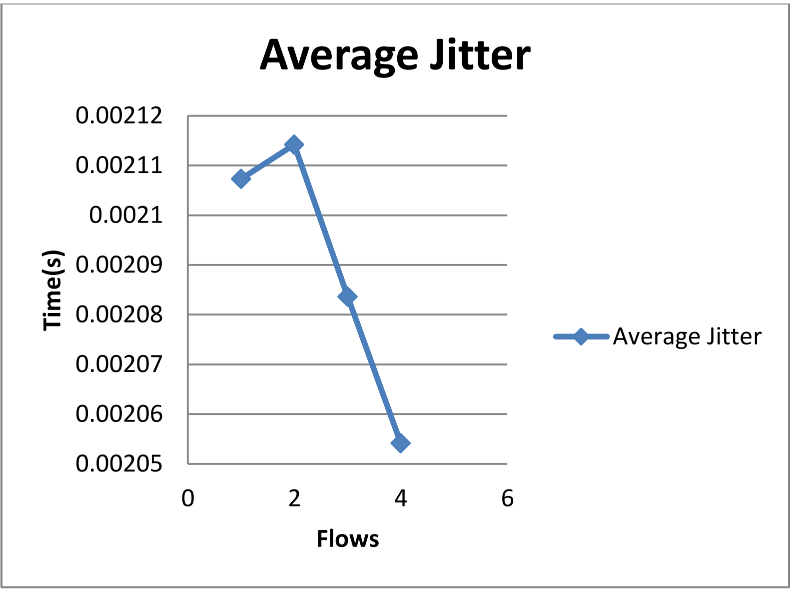


Figure 5.55: Average Jitter of BE and FCFS Combination

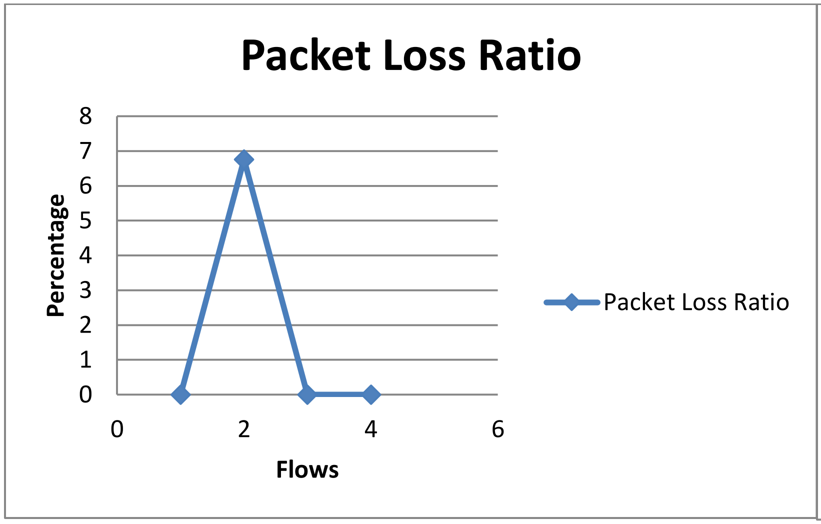


Figure 5.56: Average Jitter of BE and FCFS Combination

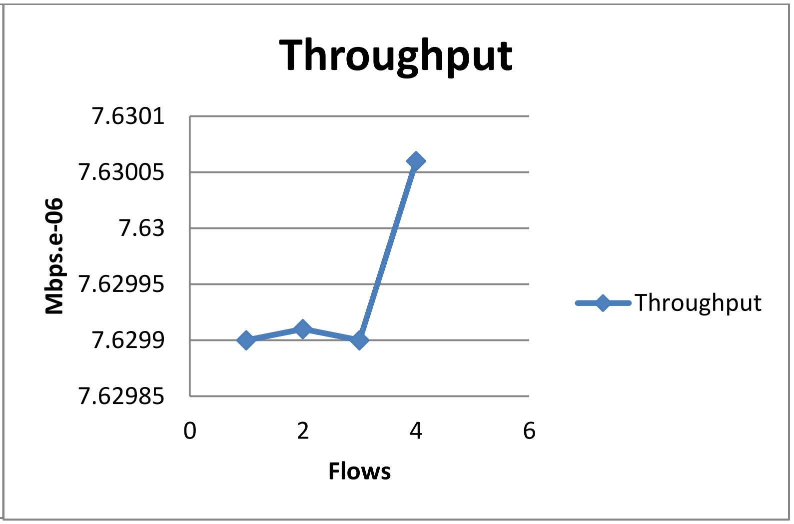


Figure 5.57: Throughput of BE and FCFS Combination

Yet again, a similar approach for analyzing Figures 5.54, 5.55, 5.56 and 5.57 above, a consideration for all the metrics is given to each flow (1, 2, 3 and 4) of Scenario 11. For Average Delay, this scenario recorded the figures of 0.0203661s, 0.0196514s, 0.0202529s and 0.0211061 for flows 1, 2, 3 and 4, respectively. The smallest Average Delay being 0.0196514s of flow 2 and the biggest Average Delay being 0.0211061 of flow 4.

For the Average Jitter, figures 0.0021073s, 0.00211418s, 0.00208365s, and 0.00205418s for flows 1, 2, 3 and 4, respectively, were generated. The smallest Average Jitter being 0.00205418s of flow 4 and the biggest Average Jitter being 0.00211418s of flow 2. The Packet Loss Ratio, recorded figures of 0, 6.75676e-05, 0 and 0 for flows 1, 2, 3 and 4, respectively. Flows 1, 3 and 4 recorded the least PLR of 0 whereas flow 2 recorded 6.7567e05 as the highest PLR. Lastly on Scenario 11, the Throughput values for flows 1, 2, 3, and 4 were recorded with figures of 7.6299e-06, 7.62991e-06, 7.6299e-06, and 7.63006e-06 respectively. Flows 1 and 3 recorded the lowest Throughput values of 7.6299e-06 each whereas Flow 4 recorded the highest Throughput value of 7.63006e-06.

5.3.12. Scenario 12: BE and RTPS

The twelfth scenario saw the implementation of BE and RTPS as the underlying WiMAX network combination factors and the results of their resultant network performance are presented in Figures 5.58, 5.59, 5.60 and 5.61 below with a specific focus on the QoS metrics such as Average Delay, Average Jitter, Packet Loss Ratio and Throughput.(Also see **Appendix** L)

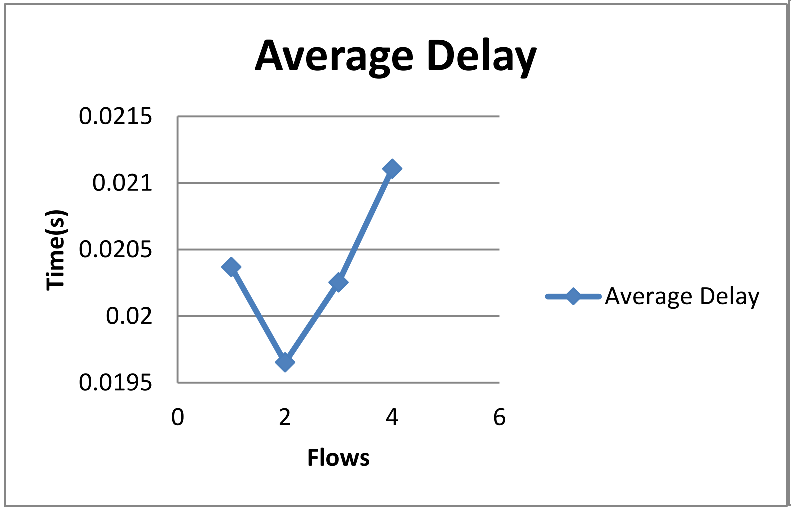


Figure 5.58: Ave. Delay of BE and FCFS Combination

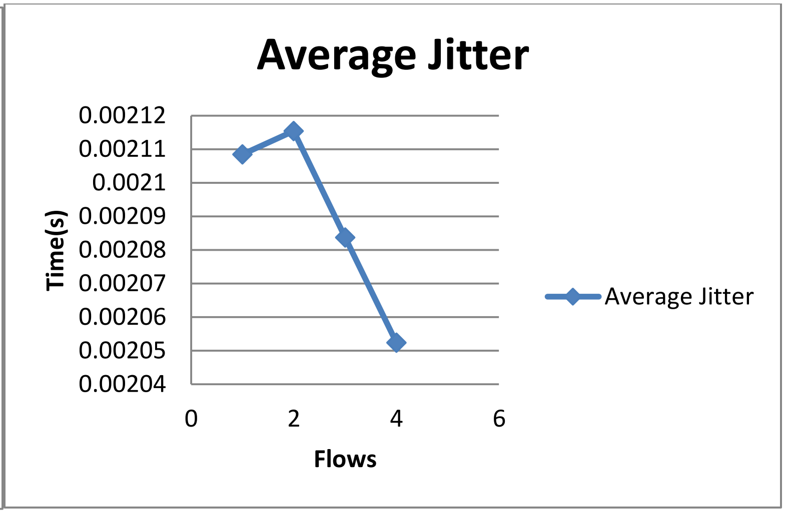


Figure 5.59: Average Jitter of BE and FCFS Combination

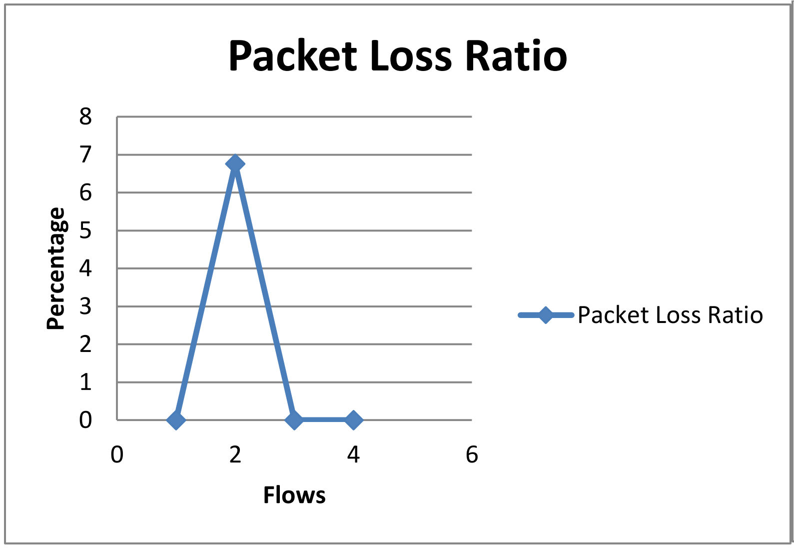


Figure 5.60: Average Jitter of BE and FCFS Combination

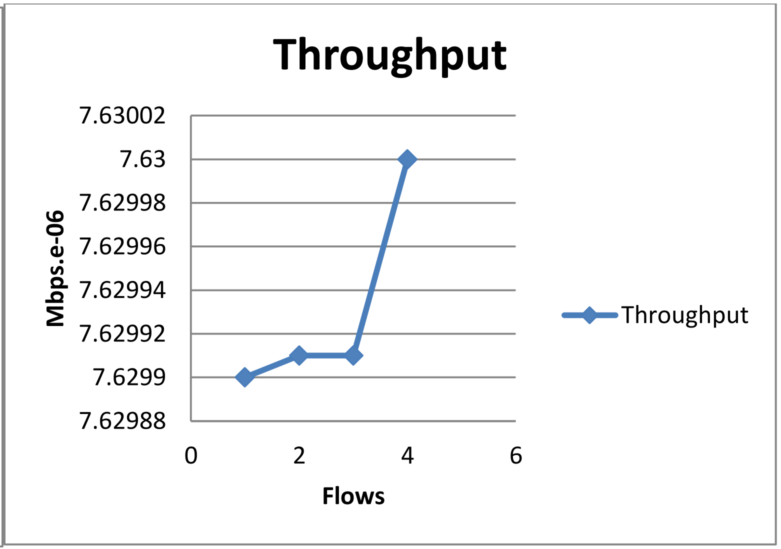


Figure 5.61: Throughput of BE and FCFS Combination

Last but not least, analyzing Figures 5.58, 5.59, 5.60 and 5.61 above, a similar approach was taken and a consideration for all the metrics is given to each flow (1, 2, 3 and 4) of Scenario 12. For Average Delay, this scenario recorded the figures of 0.0203682s, 0.0196527s, 0.0202536s and 0.0211068s for flows 1, 2, 3 and 4, respectively, the smallest. The smallest Average Delay being 0.0196527s of flow 2 and the biggest Average Delay being 0.0211068s of flow 4.

For the Average Jitter, figures 0.00210846s, 0.0021154s, 0.00208369s, and 0.00205239s for flows 1, 2, 3 and 4, respectively, were generated. The smallest Average Jitter being 0.00205239s of flow 4 and the biggest Average Jitter being 0.0021154s of flow 2. The Packet Loss Ratio, recorded figures of 0, 6.75676e-05, 0 and 0 for flows 1, 2, 3 and 4 respectively. Flows 1, 3 and 4 recorded the smallest PLR values of 0 whilst flow 2 recorded the biggest PLR of 6.75676e-05. Finally, the Throughput for flows 1, 2, 3, and 4 were recorded with figures of 7.6299e-06, 7.62991e-06, 7.6299e-06, and 7.63e-06, respectively. Flows 1 and 3 recorded the lowest Throughput values of 7.6299e-06 each whereas Flow 4 recorded the highest Throughput value of 7.63e-06.

5.4. Results Implication and Assessment

This section leads from the results presentations tendered in Section 5.3 to present the conclusive implications and assessments of the observations made from them. In chapter 5, the results were presented from an umbrella point of view wherein the WiMAX metrics observations were clustered together based on a certain scenario created. In this section, the results presented by the four WiMAX metrics of this research project are hereby analysed independently to assess how all combinations of CoS's with SchedType contributed towards the performance of the network. In this case, a certain metric is identified and the network performance is collected from all the twelve (12) scenarios created using it. To this effect, the process begins with assessing the Average Delay, followed by the Average Jitter, Throughput and Packet Loss Ratio as the last assessment.

5.4.1. Average Delay Implication

The Average Delay of each flow was monitored and recorded. Twelve Scenarios were created in which each CoS was individually combined with each SchedType. The WiMAX network Average Delay was recorded as presented in Figure 5.62below (Also See **Appendix** M)

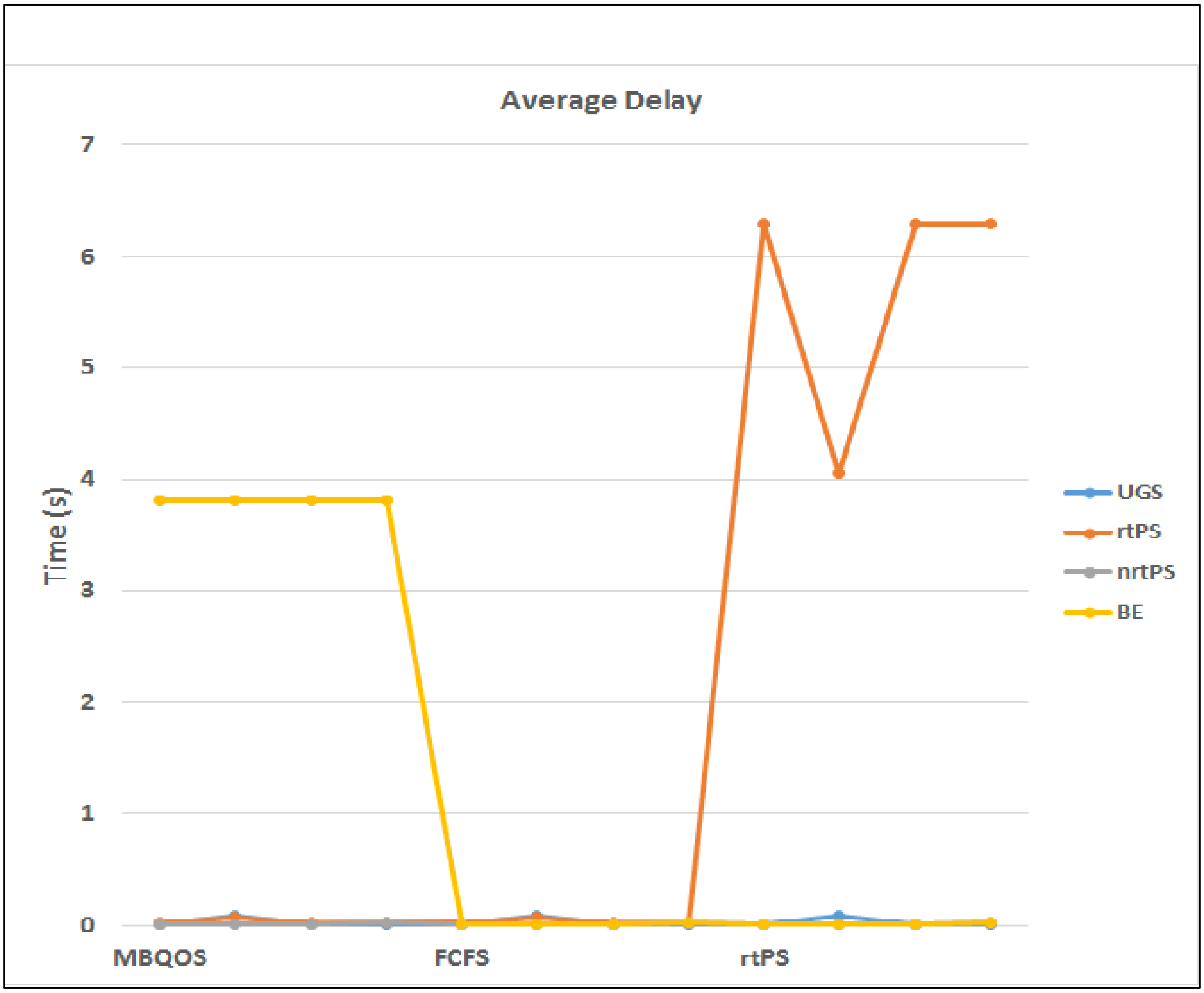


Figure 5.62: CoSs vs SchedTypes WiMAX Average Delay

As a result of different combinations of the CoSs and SchedTypes, Figure 5.62 above depicts a simulated WiMAX network Average Delay that was generated. The Average Delay is presented in terms of a SchedType performance on four CoSs or the CoSs performance on a particular SchedType. To this effect, the clustered four bars, which each represented flows 1, 2, 3, and 4, represent the CoSs and the first four clusters (with each cluster representing a single flow) are observations over MBQOS. The next four clustered bars are an observation of WiMAX Average Delay over FCFS. Lastly, the last four clustered bars are an observation of WiMAX Delay over rtPS SchedType.

A closer look at Figure 5.62 shows how each combination of a CoS and SchedType has caused WiMAX Average Delays of each network flow. It shows instances of Average Delay variations from low-to-high. It further presents how certain combinations can be consistent and how other CoSs and SchedTypes can perform poorly when combined together. Furthermore, it presents a picture of how a particular CoS can poorly perform when combined with a particular SchedType but perform relatively better when combined with the rest of SchedTypes.

A. ***MBQOS vs UGS, rtPS, nrtPS and BE- Flow 1:***

The MBQOS combination with the CoSs for flow 1 produced WiMAX Average Delays of **0.00865201s**, **0.0266159s** and **0.0203682sand 3.82s** for UGS, rtPS, nrtPS and BE, respectively. However, MBQOS produced a relatively high WiMAX Average Delay when combined with BE at ***3.82s*** and low Average Delay of **0.00865201s** when combined with UGS.

B. ***MBQOS vs UGS, rtPS, nrtPS and BE- Flow 2:***

For this combination, again MBQOS produced WiMAX Average Delay that is characterized by levels of inconsistency when combined with different CoSs. For UGS, the Average Delay was **0.0881707s**, **0.064642s**, **0.0196527s** and **3.81953s** for UGS, rtPS, nrtPS and BE, respectively. Again MBQOS produced a relatively high WiMAX Average Delay when combined with BE at **3.81953s** and low Average Delay of **0.0196527s** when combined with nrtPS.

C. ***MBQOS vs UGS, rtPS, nrtPS and BE- Flow 3:***

For flow 3, MBQOS combinations produced WiMAX Average Delays of **0.00880161s**, **0.0265192s**, **0.0202536s**, and **3.81985s** for UGS, rtPS, nrtPS and BE, respectively. Combined with BE, MBQOS continued to produce a relatively high WiMAX Average Delay at **3.81985s** and low Average Delay of **0.00880161s** when combined with UGS.

D. ***MBQOS vs UGS, rtPS, nrtPS and BE- Flow 4:***

For flow 4, the MBQOS combinations produced WiMAX Average Delays of **0.0101704s**, **0.0273927s**, **0.0211068s** and **3.82103s** for UGS, rtPS, nrtPS and BE respectively. Combined with BE, MBQOS continued to produce a relatively high WiMAX Average Delay of **3.82103s** and low Average Delay of **0.0101704s** when combined with UGS.

E. ***FCFS vs UGS, rtPS, nrtPS and BE- Flow 1:***

The FCFS combinations with all CoSs has consistently produced low Average Delays for flow 1. The WiMAX Average Delays were recorded as **0.00865201s**, **0.021204s, 0.0203652s** and **0.0203661s** for UGS, rtPS, nrtPS and BE, respectively. Combined with rtPS, FCFS produced a relatively high WiMAX Average Delay of **0.021204s** and low Average Delay of **0.00865201s** when combined with UGS.

F. ***FCFS vs UGS, rtPS, nrtPS and BE- Flow 2:***

The WiMAX Average Delays continued to be low for this combination in flow 2, although slightly different from flow 1. The WiMAX Average Delays were recorded as **0.0881707s**, **0.0698529s**, **0.0196505s** and **0.0196514s** for UGS, rtPS, nrtPS and BE. Combined with UGS, FCFS produced a relatively high WiMAX Average Delay of **0.0881707s** and low Average Delay of **0.0196505s** when combined with nrtPS.

G. ***FCFS vs UGS, rtPS, nrtPS and BE- Flow 3:***

For flow 3, the FCFS combinations with all CoSs continued to produce low WiMAX Average Delays. The WiMAX Average Delays were recorded as **0.00880161s**, **0.0210928s, 0.02052s** and **0.0202529s** for UGS, rtPS, nrtPS and BE, respectively. Combined with rtPS, FCFS produced a relatively high WiMAX Average Delay of **0.0210928s** and low Average Delay of **0.00880161s** when combined with UGS.

H. ***FCFS vs UGS, rtPS, nrtPS and BE- Flow 4:***

The FCFS combinations with all CoSs continued to produce low WiMAX Average Delays in flow 4. The WiMAX Average Delays were recorded as **0.0101704s**, **0.0219935s, 0.0211053s** and **0.211061s** for UGS, rtPS, nrtPS and BE, respectively. Combined with rtPS, FCFS continued to produce a relatively high WiMAX Average Delay of **0.0219935s** and low Average Delay of **0.0101704s** when combined with UGS.

I. ***rtPS vs UGS, rtPS, nrtPS and BE- Flow 1:***

For flow 1, the rtPS combinations produced a WiMAX Average Delay of **0.00865201s**, **6.29477s, 0.0203674s** and **0.0203682s** for UGS, rtPS, nrtPS and BE respectively. It is worth noting that this combination has produced the second highest WiMAX Average Delay of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS with **6.29477s**. Combined with UGS, rtPS produced the lowest WiMAX Average Delay of **0.00865201s.**

J. ***rtPS vs UGS, rtPS, nrtPS and BE- Flow 2:***

The rtPS combinations with all CoSs continued to produce low WiMAX Average Delays in flow 2. The WiMAX Average Delays were recorded as**0.0881707s**, **4.05401s, 0.019652s** and **0.0196527s** for UGS, rtPS, nrtPS and BE respectively. Combined with rtPS, rtPS continued to produce a relatively high WiMAX Average Delay of **4.05401s** and low Average Delay of **0.019652s** when combined with nrtPS.

K. ***rtPS vs UGS, rtPS, nrtPS and BE- Flow 3:***

Similarly for flow 1, the rtPS combinations produced a WiMAX Average Delay of **0.00880161s**, **6.29365s, 0.0202528s** and **0.0202536s** for UGS, rtPS, nrtPS and BE, respectively. Equally, it is worth noting that this combination, indifferently from flow 1 of rtPS combinations, has produced the third highest WiMAX Average Delay of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS with **6.29365s**. Combined with UGS, rtPS produced the lowest WiMAX Average Delay of **0.00880161s.**

L. ***rtPS vs UGS, rtPS, nrtPS and BE- Flow 4:***

Similarly to flows 1 and 3, the rtPS combinations produced a WiMAX Average Delay of **0.0101704s**, **6.29594s, 0.2111052s** and **0.0211068** for UGS, rtPS, nrtPS and BE, respectively for flow 4. It is worth noting that this combination has produced the highest WiMAX Average Delay of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS with **6.29594s**.Combined with UGS, rtPS produced the lowest WiMAX Average Delay of **0.0101704s.**

5.4.2. Average Jitter Implication

Similarly to the Average Delay, various combinations of each flow were monitored and recorded for Average Jitter. Twelve Scenarios were also created in which each CoS was individually combined with each SchedType. The WiMAX network Average Jitter was recorded as presented in Figure 5.63(Also See **Appendix** N)

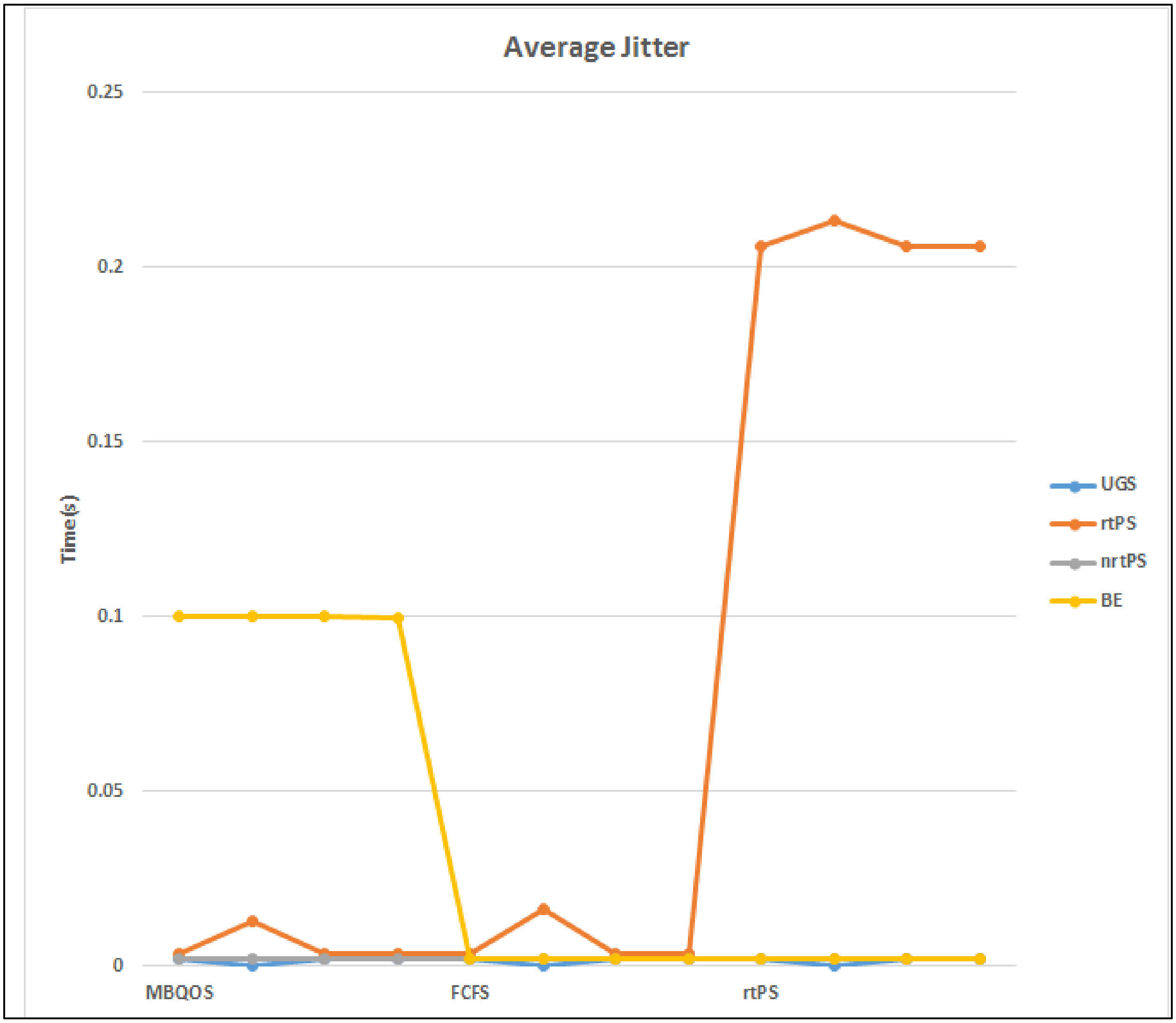


Figure 5.63: CoSs vs SchedTypes WiMAX Average Jitter

As different combinations of the CoSs and SchedTypes were tested, Figure 5.63 above depicts a simulated WiMAX network Average Jitter that was generated thereof. In a similar approach to the Average Delay, the Average Jitter is presented in terms of a SchedType performance on four CoSs or the CoSs performance on a particular SchedType. To this effect, the clustered four bars, which each represented flows 1, 2, 3, and 4, represent the CoSs and the first four clusters (with each cluster representing a single flow) are observations over MBQOS. The next four clustered bars are an observation of WiMAX Average Jitter over FCFS. Lastly, the last four clustered bars are an observation of WiMAX Average Jitter over rtPS SchedType. Again, The WiMAX network Average Jitter is further presented in the following figure to show performance and trends generated

Taking a closer look at Figure 5.63 shows how each combination of a CoS and SchedType has caused WiMAX Average Jitter of each network flow. It shows instances of Average Jitter variations from low-to-high. As it was the case with the Average Delay, Figure 5.63 further presents how certain combinations can be consistent and how other CoSs and SchedTypes can perform poorly when combined together. Furthermore, it presents a picture of how a particular CoS can poorly perform when combined with a particular SchedType but perform relatively better when combined with the rest of SchedTypes.

A. *MBQOS vs UGS, rtPS, nrtPS and BE- Flow 1:*

The MBQOS combination with the CoSs for flow 1 produced WiMAX Average Jitter records of **0.00211945s**, **0.00383462s, 0.00210847s** and **0.100081s** for UGS, rtPS, nrtPS and BE, respectively. However, MBQOS produced a relatively high WiMAX Average Jitter when combined with BE at **0.100081s** and low Average Jitter of **0.00210847s** when combined with nrtPS.

B. *MBQOS vs UGS, rtPS, nrtPS and BE- Flow 2:*

For this combination, again MBQOS produced WiMAX Average Jitter that is characterized by levels of inconsistency when combined with different CoSs. For UGS, the Average Jitter was **-nan s**, **0.0127833s**, **0.0021154s** and **0.10009s** for UGS, rtPS, nrtPS and BE, respectively. Again MBQOS produced a relatively high WiMAX Average Jitter when combined with BE at **0.10009s,** and low Average Jitter of **–nan s** when combined with nrtPS.

C. *MBQOS vs UGS, rtPS, nrtPS and BE- Flow 3:*

For flow 3, MBQOS combinations produced WiMAX Average Jitter of **0.00209911s**, **0.00385171s**, **0.0020837s**, and **0.100081s** for UGS, rtPS, nrtPS and BE, respectively. Combined with BE, MBQOS continued to produce a relatively high WiMAX Average Jitter at **0.100081s** and low Average Jitter of **0.0020837s** when combined with nrtPS.

D. *MBQOS vs UGS, rtPS, nrtPS and BE- Flow 4:*

For flow 4, the MBQOS combinations produced WiMAX Average Jitter of **0.00202856s**, **0.00382415s**, **0.00205239s** and **0.0998238s** for UGS, rtPS, nrtPS and BE, respectively. Combined with BE, MBQOS continued to produce a relatively high WiMAX Average Jitter of **0.0998238s** and low Average Jitter of **0.00202856s** when combined with UGS.

E. *FCFS vs UGS, rtPS, nrtPS and BE- Flow 1:*

The FCFS combinations with all CoSs has consistently produced low Average Jitter records for flow 1. The WiMAX Average Jitter were recorded as **0.00211945s**, **0.00385276s, 0.00210732s** and **0.0021073s** for UGS, rtPS, nrtPS and BE respectively. Combined with rtPS, FCFS produced a relatively high WiMAX Average Jitter of **0.00385276s** and low Average Delay of **0.0021073s** when combined with BE.

F. *FCFS vs UGS, rtPS, nrtPS and BE- Flow 2:*

The WiMAX Average Jitter continued to be low for this combination in flow 2, although slightly different from flow 1. The WiMAX Average Jitter were recorded as **–nan s**, **0.0162272s**, **0.00211417s** and **0.00211418s** for UGS, rtPS, nrtPS and BE. Combined with UGS, FCFS produced a relatively high WiMAX Average Jitter of **0.0162272s** and low Average Jitter of **–nan s** when combined with UGS.

G. *FCFS vs UGS, rtPS, nrtPS and BE- Flow 3:*

For flow 3, the FCFS combinations with all CoSs continued to produce low WiMAX Average Jitter. The WiMAX Average Jitter values were recorded as **0.00209911s**, **0.00383745s, 0.0020837s** and **0.00208365s** for UGS, rtPS, nrtPS and BE respectively. Combined with rtPS, FCFS produced a relatively high WiMAX Average Jitter of **0.00383745s** and low Average Jitter of **0.00208365s** when combined with BE.

H. *FCFS vs UGS, rtPS, nrtPS and BE- Flow 4:*

The FCFS combinations with all CoSs continued to produce low WiMAX Average Jitter in flow 4. The WiMAX Average Jitter values were recorded as**0.00202856s**, **0.00384151s, 0.00205429s and 0.00205418** for UGS, rtPS, nrtPS and BE, respectively. Combined with rtPS, FCFS continued to produce a relatively high WiMAX Average Jitter of **0.00384151s** and low Average Jitter of **0.00202856s** when combined with UGS.

I. *rtPS vs UGS, rtPS, nrtPS and BE- Flow 1:*

For flow 1, the rtPS combinations produced a WiMAX Average Jitter of **0.00211945s**, **0.206328s, 0.00210845s** and **0.00210846s** for UGS, rtPS, nrtPS and BE, respectively. It is worth noting that this combination has produced the second highest WiMAX Average Jitter of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS with **0.206328s**. Combined with nrtPS, rtPS produced the lowest WiMAX Average Jitter of **0.00210845s.**

J. *rtPS vs UGS, rtPS, nrtPS and BE- Flow 2:*

The rtPS combinations with all CoSs continued to produce low WiMAX Average Jitter in flow 2. The WiMAX Average Jitter were recorded as–**nan s**, **0.213374s, 0.00211537s** and **0.0021154s** for UGS, rtPS, nrtPS and BE, respectively. Combined with rtPS, rtPS continued to produce a relatively high WiMAX Average Jitter of **0.213374s**and low Average Jitter of **–nan s** when combined with UGS.

K. *rtPS vs UGS, rtPS, nrtPS and BE- Flow 3:*

Similarly for flow 1, the rtPS combinations produced a WiMAX Average Jitter of **0.00209911s**, **0.206321s, 0.0020837s** and **0.00208369s** for UGS, rtPS, nrtPS and BE, respectively. Equally, it is worth noting that this combination, indifferently from flow 1 of rtPS combinations, has produced the third highest WiMAX Average Jitter of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS with **0.206321s**. Combined with BE, rtPS produced the lowest WiMAX Average Jitter of **0.00208369s.**

L. *rtPS vs UGS, rtPS, nrtPS and BE- Flow 4:*

Similarly to flows 1 and 3, the rtPS combinations produced a WiMAX Average Jitter of **0.00202856s**, **0.206294s, 0.00205281s** and **0.00205239s** for UGS, rtPS, nrtPS and BE, respectively for flow 4. It is worth noting that this combination has produced the highest WiMAX Average Jitter of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS with **0.206294s**. Combined with UGS, rtPS produced the lowest WiMAX Average Jitter of **0.00202856s.**

5.4.3. Throughput Implication

The Throughput of each flow was monitored and recorded. Twelve Scenarios were created in which each CoS was individually combined with each SchedType. The WiMAX network Throughput was recorded as presented in Figure 5.64 below (Also See **Appendix** O)

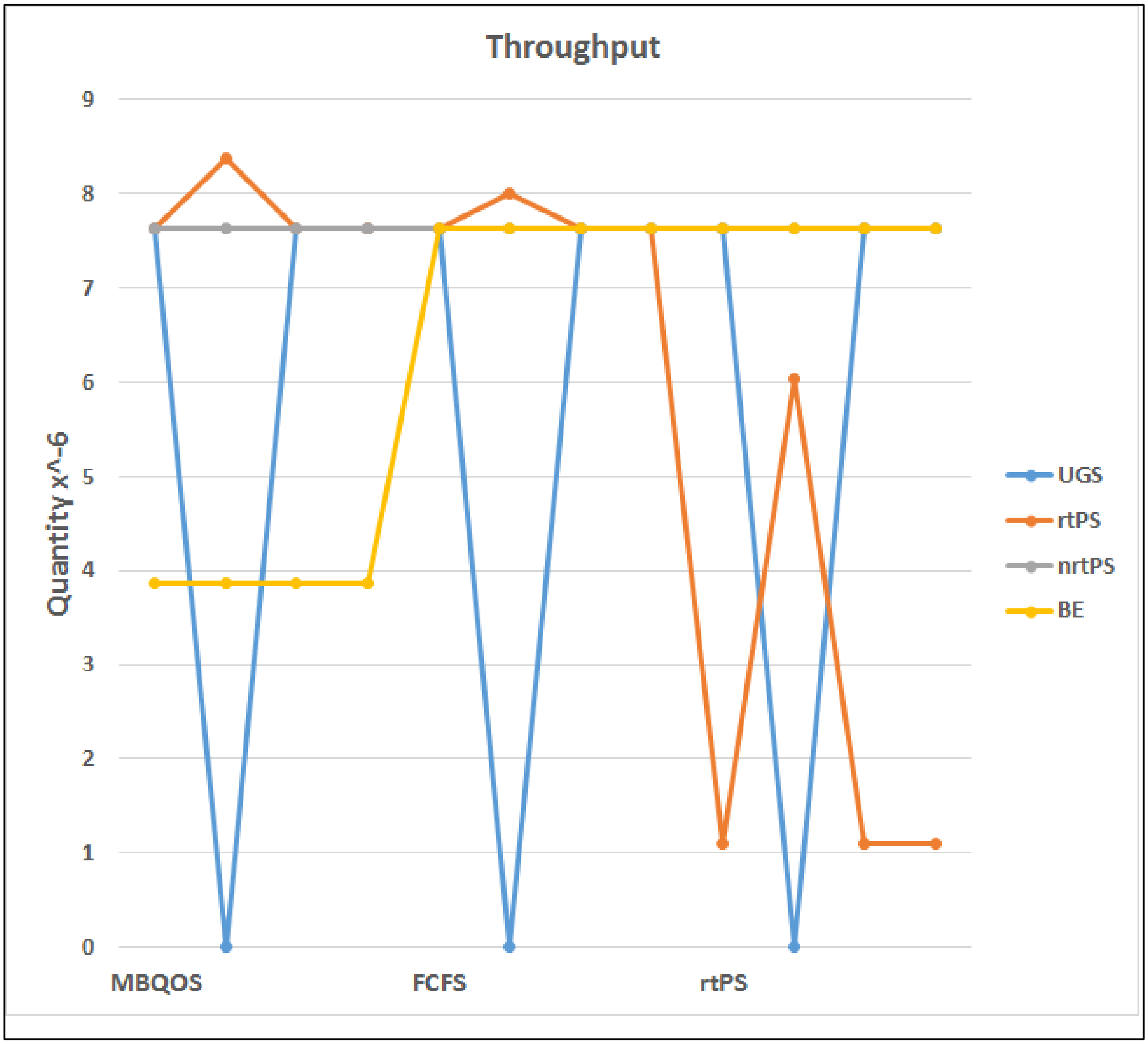


Figure 5.64: CoSs vs SchedTypes WiMAX Throughput

Figure 5.64 above presents a simulated WiMAX network Throughput that was generated as a result of different combinations of the CoSs and SchedTypes. The Throughput is presented in terms of a SchedType performance on four CoSs or the CoSs performance on a particular SchedType. To this effect, the clustered four bars, which each represented flows 1, 2, 3, and 4, represent the CoSs and the first four clusters (with each cluster representing a single flow) are observations over MBQOS. The next four clustered bars are an observation of WiMAX Throughput over FCFS. Lastly, the last four clustered bars are an observation of WiMAX Throughput over rtPS SchedType. In instances where there are just three instead of four bars, it means that the Throughput was infinite and because infinite is not a numerical value, it could not be represented as a numerical value. The WiMAX network Throughput is further presented in the following figure to show performance and trends generated:

A closer look at Figure 5.64 shows how each combination of a CoS and SchedType has contributed to the WiMAX Throughput of each network flow. It shows instances of Throughput variations from low-to-high. It further presents how certain combinations can be consistent and how other CoSs and SchedTypes can perform poorly when combined together. Furthermore, it presents a picture of how a particular CoS can poorly perform when combined with a particular SchedType but perform relatively better when combined with the rest of SchedTypes.

M. *MBQOS vs UGS, rtPS, nrtPS and BE- Flow 1:*

The MBQOS combination with the CoSs for flow 1 produced a consistent WiMAX Throughput of **7.6299e-06Mbps**, **7.62996e-06Mbps** and **7.6299e-06Mbps** for UGS, rtPS and nrtPS, respectively. However, MBQOS produced a relatively low WiMAX Throughput when combined with BE at **3.8633e-06Mbps**

N. *MBQOS vs US, rtPS, nrtPS and BE- Flow 2:*

For this combination, again MBQOS produced WiMAX Throughput that is characterized by levels of inconsistency when combined with different CoSs. For UGS, the Throughput was Inf. Mbps, **8.3781e-06Mbps** for rtPS (the highest Throughput of all combinations), **7.62991e-06Mbps** for nrtPS and **3.86316e-06Mbps** for BE.

O. *MBQOS vs UGS, rtPS, nrtPS and BE- Flow 3:*

Similarly to flow 1, MBQOS combinations produced a WiMAX Throughput of **7.6299e-06Mbps**, **7.62996e-06Mbps** and **7.6299e-06Mbps** for UGS, rtPS and nrtPS, respectively. Combined with BE, MBQOS continued to produce a relatively poor WiMAX Throughput of **3.8633eMbps.**

P. *MBQOS vs UGS, rtPS, nrtPS and BE- Flow 4:*

Similarly to flows 1 and 3, with a much improved performance MBQOS combinations produced a WiMAX Throughput of **7.63043e-06Mbps**, **7.63011e-06Mbps** and **7.63-06Mbps** for UGS, rtPS and nrtPS, respectively. Combined with BE, although with an improved performance than in flows 1, 2 and 3, MBQOS continued to produce a relatively poor WiMAX Throughput of **3.8683eMbps**.

Q. ***FCFS vs UGS, rtPS, nrtPS and BE- Flow 1:***

The FCFS combinations with all CoSs have proved to be consistent in flow 1. The WiMAX Throughput is more or less the same with any CoS in this flow whereby **7.6299e-06Mbps**, **7.62991e-06Mbps, 7.6299e-06Mbps** and **7.6299e-06Mbps** produced for UGS, rtPS, nrtPS and BE, respectively.

R. ***FCFS vs UGS, rtPS, nrtPS and BE- Flow 2:***

For this combination, again FCFS produced WiMAX Throughput that is characterized by levels of inconsistency when combined with different CoSs. For UGS, the Throughput was **Inf. Mbps**, **7.99736e-06Mbps** for rtPS (the highest Throughput of FCFS combinations), **7.62991e-06Mbps** for nrtPS and **7.62991e-06Mbps** for BE.

S. ***FCFS vs UGS, rtPS, nrtPS and BE- Flow 3:***

Similarly to flow 1, the FCFS combinations with all CoSs have proved to be consistent in this flow. The WiMAX Throughput is more or less the same with any CoS in this flow whereby **7.6299e-06Mbps**, **7.62991e-06Mbps, 7.6299e-06Mbps** and **7.6299e-06Mbps** produced for UGS, rtPS, nrtPS and BE, respectively.

T. ***FCFS vs UGS, rtPS, nrtPS and BE- Flow 4:***

Similarly to flows 1 and 3, the FCFS combinations with all CoSs have proved again to be consistent in flow 4. The produced WiMAX Throughput is more or less the same with any CoS in this flow whereby **7.63043e-06Mbps**, **7.630421e-06Mbps, 7.63006e-06Mbps** and **7.63006e-06Mbps** produced for UGS, rtPS, nrtPS and BE, respectively. WiMAX Throughput produced was slightly improved in this flow as opposed to other flows under this combination.

U. ***rtPS vs UGS, rtPS, nrtPS and BE- Flow 1:***

For flow 1, the rtPS combinations produced a WiMAX Throughput of **7.6299e-06Mbps**, **1.10158e-06Mbps, 7.6299e-06Mbps** and **7.6299e-06Mbps** for UGS, rtPS, nrtPS and BE, respectively. It is worth noting that this combination has produced the lowest WiMAX Throughput of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS.

V. ***rtPS vs UGS, rtPS, nrtPS and BE- Flow 2:***

In this flow 1, the rtPS combinations produced a WiMAX Throughput of **Inf. Mbps**, **6.03608e-06Mbps, 7.62991e-06Mbps** and **7.62991e-06Mbps** for UGS, rtPS, nrtPS and BE, respectively. For this combination, Flow 2 recorded the highest WiMAX Throughput with the combination of rtPS as a SchedType and rtPS as a CoS.

W. ***rtPS vs UGS, rtPS, nrtPS and BE- Flow 3:***

Similarly for flow 1, the rtPS combinations produced a WiMAX Throughput of **7.6304e-06Mbps**, **1.10158e-06Mbps, 7.6299e-06Mbps** and **7.6299e-06Mbps** for UGS, rtPS, nrtPS and BE, respectively. Equally, it is worth noting that this combination, indifferently from flow 1 of rtPS combinations, has produced the lowest WiMAX Throughput of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS. Also, it is worth noting that rtPS performed much better in this flow when combined with UGS than any other CoS.

X. ***rtPS vs UGS, rtPS, nrtPS and BE- Flow 4:***

Similarly to flows 1 and 3, the rtPS combinations produced a WiMAX Throughput of **7.6299e-06Mbps**, **1.10158e-06Mbps, 7.6299e-06Mbps** and **7.6299e-06Mbps** for UGS, rtPS, nrtPS and BE, respectively for flow 4. It is worth noting that this combination has produced the lowest WiMAX Throughput of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS. Equally, it is worth noting that this combination, indifferently from flows 1 and 3of rtPS combinations, has produced the lowest WiMAX Throughput of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS.

5.4.4. Packet Loss Ratio Implication

A Similar approach applied to the WiMAX Average Delay, Average Jitter and Throughput, each flow was monitored and recorded for Packet Loss Ratio. Consequently, twelve Scenarios were also created in which each CoS was individually combined with each SchedType. The WiMAX network Packet Loss Ratio was recorded as presented in Figure 5.65 below (Also See **Appendix** P)

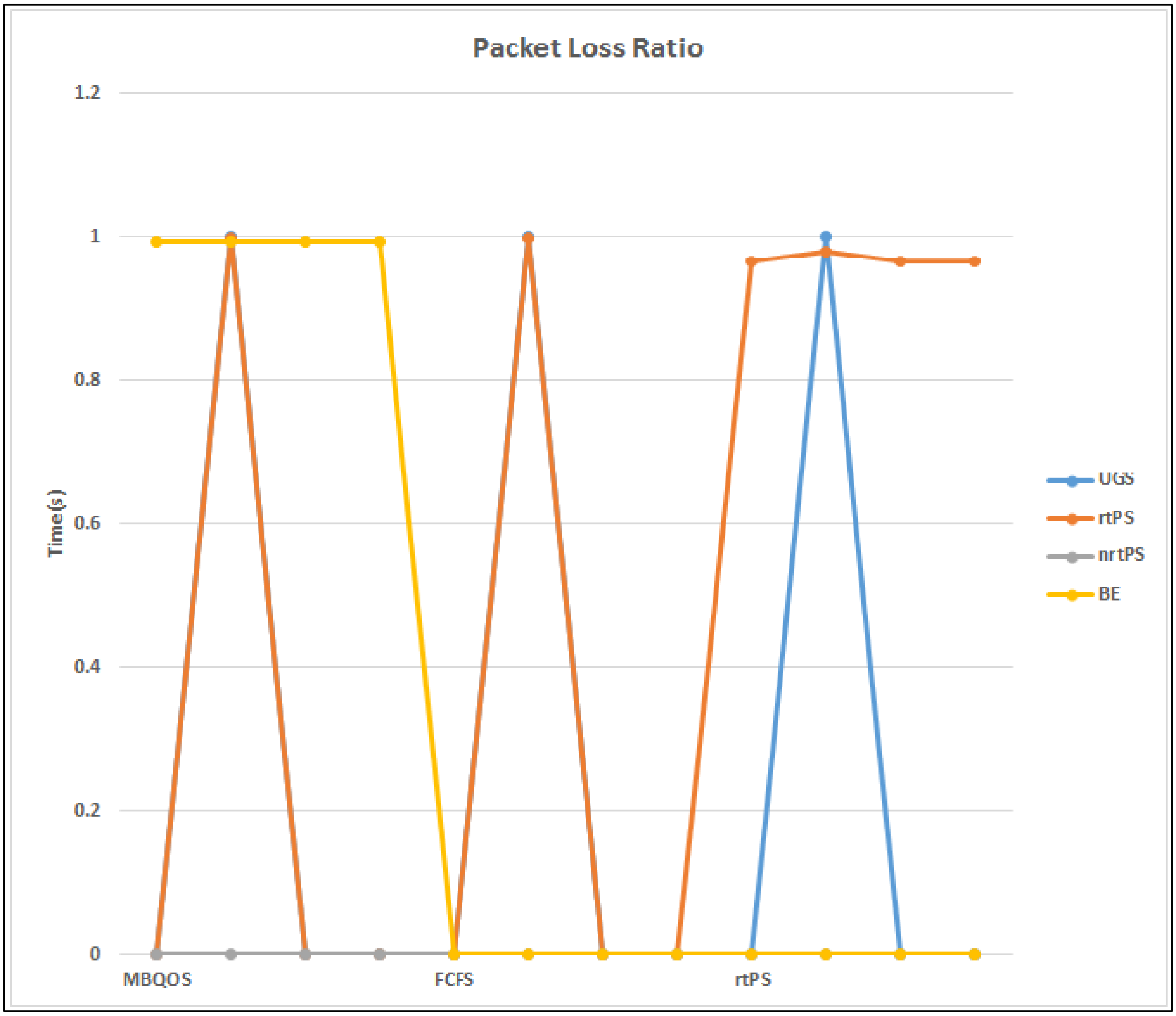


Figure 5.65: CoSs vs SchedTypes WiMAX Packet Loss Ratio

Following the testing process of different combinations of the CoSs and SchedTypes, Figure 5.65 above depicts a simulated WiMAX network Packet Loss Ratio that was generated thereof. In a similar approach to the Average Delay, Average Jitter and Throughput, the Packet Loss Ratio is presented in terms of a SchedType performance on four CoSs or the CoSs performance on a particular SchedType. To this effect, the clustered four bars, which each represented flows 1, 2, 3, and 4, represent the CoSs and the first four clusters (with each cluster representing a single flow) are observations over MBQOS. The next four clustered bars are an observation of WiMAX Packet Loss Ratio over FCFS. Lastly, the last four clustered bars are an observation of WiMAX Packet Loss Ratio over rtPS SchedType. Again, The WiMAX network Packet Loss Ratio is further presented in the following figure to show performance and trends generated:

A closer look at Figure 5.65 shows how each combination of a CoS and SchedType has contributed to the WiMAX Packet Loss Ratio of each network flow. It shows instances of Packet Loss Ratio variations from low-to-high. It further presents how certain combinations can be consistent and how other CoSs and SchedTypes can perform poorly when combined together. Furthermore, it presents a picture of how a particular CoS can poorly perform when combined with a particular SchedType but perform relatively better when combined with the rest of SchedTypes.

Y. ***MBQOS vs UGS, rtPS, nrtPS and BE- Flow 1:***

The MBQOS combination with the CoSs for flow 1 produced WiMAX Packet Loss Ratioof **0**, **0,0and 0.994797** for UGS, rtPS, nrtPS and BE, respectively. However, MBQOS produced a relatively high WiMAX Packet Loss Ratio when combined with BE at **0.994797**and low Packet Loss Ratio of **0** when combined with UGS, rtPS and nrtPS.

Z. ***MBQOS vs UGS, rtPS, nrtPS and BE- Flow 2:***

For this combination, again MBQOS produced WiMAX Packet Loss Ratio that is characterized by levels of inconsistency when combined with different CoSs. For UGS, the Packet Loss Ratiowas **0.999932**, **0.999932**, **6.75676e-05** and **0.994797**for UGS, rtPS, nrtPS and BE, respectively. Again MBQOS produced a relatively high WiMAX Packet Loss Ratio when combined with nrtPS at **6.75676e-05** and low Packet Loss Ratioof **0.994797**when combined with BE.

AA. ***MBQOS vs UGS, rtPS, nrtPS and BE- Flow 3:***

For flow 3, MBQOS combinations produced WiMAX Packet Loss Ratioof **0**, **0**,**0**, and **0.994797** for UGS, rtPS, nrtPS and BE respectively. Combined with BE, MBQOS continued to produce a relatively high WiMAX Packet Loss Ratioat**0.994797** and low Packet Loss Ratio of **0** when combined with UGS, rtPS and nrtPS.

BB. ***MBQOS vs UGS, rtPS, nrtPS and BE- Flow 4:***

For flow 4, the MBQOS combinations produced WiMAX Packet Loss Ratio of **0**, **0**, **0** and **0.994797** for UGS, rtPS, nrtPS and BE, respectively. Combined with BE, MBQOS continued to produce a relatively high WiMAX Packet Loss Ratioof **0.994797** and low Packet Loss Ratio of **0** when combined with UGS, rtPS and nrtPS.

CC. *FCFS vs UGS, rtPS, nrtPS and BE- Flow 1:*

The FCFS combinations with all CoSs has consistently produced low Packet Loss Ratio for flow 1. The WiMAX Packet Loss Ratio were recorded as **0**, **0, 0** and **0** for UGS, rtPS, nrtPS and BE, respectively. Combined with UGS, rtPS, nrtPS and BE, FCFS produced a consistent WiMAX Packet Loss Ratio of **0**.

DD. *FCFS vs UGS, rtPS, nrtPS and BE- Flow 2:*

The WiMAX Packet Loss Ratio proved to be higher for this combination in flow 2, depicting a great change from flow 1 . The WiMAX Packet Loss Ratio were recorded as **0.999932**, **0.998581**, **6.75676e-05** and **6.75676e-05** for UGS, rtPS, nrtPS and BE, respectively. Combined with nrtPS and BE, FCFS produced a relatively high WiMAX Packet Loss Ratio of **6.75676e-05** and low Packet Loss Ratio of **0.998581** when combined with rtPS.

EE. *FCFS vs UGS, rtPS, nrtPS and BE- Flow 3:*

For flow 3, the FCFS combinations with all CoSs continued to produce low WiMAX Packet Loss Ratio. The WiMAX Packet Loss Ratio were recorded as **0**, **0, 0** and **0** for UGS, rtPS, nrtPS and BE, respectively. Combined with UGS, rtPS, nrtPS and BE, FCFS produced a consistent WiMAX Packet Loss Ratio of **0**.

FF.*FCFS vs UGS, rtPS, nrtPS and BE- Flow 4:*

The FCFS combinations with all CoSs continued to produce low WiMAX Packet Loss Ratio in flow 4. The WiMAX Packet Loss Ratio were recorded as **0**, **0, 0** and **0** for UGS, rtPS, nrtPS and BE, respectively. Once more, when combined with UGS, rtPS, nrtPS and BE, FCFS produced a consistent WiMAX Packet Loss Ratio of **0**.

GG. *rtPS vs UGS, rtPS, nrtPS and BE- Flow 1:*

For flow 1, the rtPS combinations produced a WiMAX Packet Loss Ratio of **0**, **0.966892, 0** and **0** for UGS, rtPS, nrtPS and BE, respectively. Combined with rtPS as a CoS, rtPS produced a relatively high WiMAX Packet Loss Ratio of **0.966892** and low Packet Loss Ratio of **0** when combined with UGS, nrtPS and BE.

HH. *rtPS vs UGS, rtPS, nrtPS and BE- Flow 2:*

The rtPS combinations with all CoSs continued to produce low WiMAX Packet Loss Ratio in flow 2. The WiMAX Packet Loss Ratio were recorded as **0.999932**, **0.978986, 6.75676e-05** and **6.75676e-05** for UGS, rtPS, nrtPS and BE, respectively. Combined with nrtPS and BE, rtPS continued to produce a relatively high WiMAX Packet Loss Ratio of **6.75676e-05** and low Packet Loss Ratio of **0.978986** when combined with rtPS.

II. *rtPS vs UGS, rtPS, nrtPS and BE- Flow 3:*

Similarly for flow 1, the rtPS combinations produced a WiMAX Packet Loss Ratio of **0**, **0.966892, 0** and **0** for UGS, rtPS, nrtPS and BE, respectively. Equally, it is worth noting that this combination, indifferently from flow 1 of rtPS combinations, has produced the third highest WiMAX Packet Loss Ratio of all combinations with the combination of rtPS as a SchedType and rtPS as a CoS with **0.966892** and low Packet Loss Ratio of **0** when combined with UGS, nrtPS and BE.

JJ. *rtPS vs UGS, rtPS, nrtPS and BE- Flow 4:*

Similarly to flows 1 and 3, the rtPS combinations produced a WiMAX Packet Loss Ratio of **0**, **0.966892, 0.00027027** and **0** for UGS, rtPS, nrtPS and BE, respectively, for flow 4.Combined with rtPS, rtPS produced a relatively high WiMAX Packet Loss Ratio of **0.966892** and low Packet Loss Ratio of **0** when combined with UGS and BE.

5.5. Discussion

Several instances of inference can be drawn from observing how WiMAX average Delay, Average Jitter, Throughput and Packet Loss Ratio have been affected by different combinations of CoSs and SchedTypes. For each WiMAX metric i.e. Average Delay, Average Jitter, Throughput and Packet Loss Ratio, instances of inference will be drawn independently. To this effect, the following performance analysis is given with respect to the four QoS parameters under this study, namely, Average Delay, Average Jitter, Throughput and Packet Loss Ratio.

Considering the Average Delay, the first inference is informed by the increased amount of delay observed against the SchedTypes, MBQOS and rtPS when combined with BE and rtPS respectively. As this can be clearly viewed from Figure 5.62 above, (also see **Appendix** M), the

BE CoS caused significant amount of delays across all the four flows when combined with MBQOS. This means that during this time, connections may stay longer than expected and no service can be guaranteed. With such a performance, the measure of success of a SS or ssNodes in accessing a network to complete a connectivity during this time is not guaranteed and thus this combination compromises both GoS and QoS. This performance, where BE flows under MBQOS proved to have experienced increased amounts of delays as opposed to other CoSs, was expected. According to Chauhan *et al*, (2013) MBQOS gives more QoS prioritization to UGS, rtPS and nrtPS flows than it does to the BE flows. In MBQOS, periodic grants to request bandwidth are provided for UGS, rtPS and nrtPS flows and the UL scheduler has a responsibility of ensuring that resources are allocated to these flows, assert Chauhan et al, (2013). They also assert that in MBQOS the UL scheduler guarantees that delay and bandwidth requirements are at all times met for these three flows (UGS, rtPS and nrtPS). Therefore, with less prioritization of BE flows under MBQOS, it was expected that the high delays would be experienced.

However, a sudden drop of delay can be observed across all the remaining four flows when BE was combined with FCFS and rtPS alike. During this time, there was less or no delay caused by a BE combination and thus the GoS and QoS were not compromised from this perspective. This turn of events can be better explained by the fact that packets receive equal treatment as they are all placed in a single queue and serviced in the very same order they were lined up in the queue as articulated by Sin-seok *et al.* (2011) and Saravanaselvi and Latha (2012). As it relates to rtPS as a CoS, an inverse performance to BE is observed wherein rtPS caused less or no delay when combined with MBQOS and FCFS and thus ensured guaranteed QoS and GoS. However, when combined with the SchedType rtPS, a sudden rise in the delay is observed across all the four flows. Although it can be observed that in flow 2 the delay dropped to a certain extent, the delay itself remained very high as opposed to delays caused by other CoS's and SchedTypes. This can be explained by the fact that rtPS SchedType gives prioritization to USG>nrtPS>BE and does not include rtPS as a CoS, as explained by Chauhan et al, (2013). Therefore, UL scheduler has no responsibility to ensure that the delay and bandwidth requirement for rtPS CoS flows are met hence the increased amounts of delays are experienced in rtPS and rtPS combination. This delay has dire consequences on the network as it would require overhead of bandwidth request and polling latency. This inference informs an undertaking that the combinations of MBQOS and BE and rtPS and rtPS are not ideal for implementation on a WiMAX network as the delay they cause implores significant consequences on the network as aforementioned and discussed. However, against the FCFS SchedType, the observation suggests that there was less or no delay caused and this was true regardless of the CoS implemented. Effectively, this observation suggests that, as there was no overhead required and no need to use polling amongst others, this CoS and SchedType ensured for the guaranteed QoS and GoS.

Considering the Average Jitter, the observations, as viewed from Figures 5.63 above, (also see **Appendix** N), suggest that a significant amount of jitter was caused by the CoS's, BE and rtPS across all flows when combined with MBQOS and rtPS, respectively. The manner in which MBQOS and rtPS gives prioritization of resource allocation to CoSs as explained by Chauhan *et al*. (2013), Sin-seok*et al*. (2011) and Saravanaselvi and Latha (2012) and presented above in the average delay inference might give pointers to the significant amount of jitter caused by BE and rtPS. Additionally, the observations suggests that to a certain extent, the CoS rtPS also caused jitter across all flows when combined with MBQOS and FCFS. This implies that, jitter is likely to occur when implementing rtPS as a CoS regardless of the SchedType implemented. However, the difference lies in the varying degrees of jitter occurrences, ranging from low to high. Although the observations suggest that the jitter was fairly caused in all flows by all combinations, nrtPS remains as the least likely cause for network jitter as this can be better viewed from Figure 5.63. This means QoS and GoS can be better ensured for non-real time traffic with minimum reserved rate across all combinations through implementing nrtPS. However, QoS and GoS can be ensured if BE is not implemented with MBQOS as it can be observed that with this combination, a significant amount of jitter occurred across all the four flows. This dropped significantly when BE was combined with SchedTypes, FCFS and rtPS, and thus ensured GoS and QoS provisioning for real-time services. On the other hand, more jitter occurred when the CoS rtPS and SchedType rtPS were combined and implemented. It can also be observed that this combination yielded the highest jitter occurrences as opposed to other combinations implemented in this research project. Effectively, this can be interpreted as an occurrence that compromised the QoS and GoS when the four flows sought to transmit all the queued packets (the real-time services) using the available bandwidth. Lastly on this inference, UGS and nrtPS proved to be robust against jitter occurrences across all flows in all combinations. This is a similar experience observed by Sharma and Chawla (2014) and Anouari and Haqiq (2012) where in their studies the jitter value was very small for the UGS class. This can be explained by the assertion made by So-In *et al.* (2010) that, amongst other things, UGS was designed to strictly meet the jitter constraints. This implies that less jitter occurred on real-time services provided through UGS and thus ensured provisioning of guaranteed QoS and GoS alike. Also, guaranteed QoS and GoS were ensured for the non-real-time services provisioned for using nrtPS.

As it relates to the throughput presented in Figure 5.64, (also see **Appendix** O), the first inference to draw is informed by the observation of rtPS as a CoS. When rtPS as a CoS is combined with MBQOS and FCFS as SchedTypes, the WiMAX Throughput produced proved to be very high for both SchedTypes. Additionally, the highest Throughput simulated was as a result of rtPS and MBQOS combination. This outcome was expected as MBQOS allows rtPS and nrtPS bandwidth requests to migrate from the intermediate priority queue to the high priority queue and thus ensure their QoS requirements are met, as articulated in (“Nsnam”, 2015). However, the rtPS WiMAX Throughput as a CoS started to drop when combined with the rtPS as SchedType. This suggests the combination of rtPS and rtPS as a CoS and SchedType, respectively, cannot be considered as either best or better for enhancing WiMAX Throughput whilst ensuring the guaranteed QoS is met.

Secondly, the BE combinations proved to be poor when combined with MBQOS but much better with the other SchedTypes. This was expected because in MBQOS scheduler the bandwidth requests for BE services are stored in a low priority queue as explained by Ismail *et al.* (2010). This suggests that the combination of BE and MBQOS as a CoS and SchedType, respectively, is also not a best option for WiMAX Throughput enhancement. A similar inference can be drawn for UGS as it was depicted that across all SchedTypes, the packet loss ratio would lead to lowest Throughput figures.

The last inference that can be drawn is that, nrtPS proved to produce better WiMAX Throughput outcomes in a manner that is very consistent across all the combinations and flows. This can be explained by the account made by Farooq and Turletti (2009) that nrtPS flows are polled when sufficient bandwidth is available. Meaning, the success rate of packet delivery, once get transmitted is guaranteed as the flows are only polled when there is enough bandwidth to transmit. Furthermore, this can be explained by the account made by Ismail *et al.* (2010) that nrtPS requests, similarly to rtPS, in MBQOS can migrate from the intermediate queue to the high priority queue to enhance the chances that their QoS requirements are met. This suggests that the QoS requirements such as minimum reserved rate, maximum sustained rate and traffic priority are guaranteed when combining nrtPS with different SchedType.

This implies that for RMAs such as Dwesa, where network users are not really engaged on applications such as VoIP, Video Streaming and Telemedicine, amongst others, the WiMAX network can be provided at a cost-effective manner but still with ensured guaranteed QoS. Bandwidth can be provided to meet the QoS requirement of nrtPS and thus cut the costs of providing bandwidth for other applications which the RMA users are not engaged on. The urgency of cutting cost to remain sustainable or to survive also relates to the provisioning of ICTs with added pressure on projects such as Dwesa which are sponsorship-reliant. Therefore, in such projects, Throughput should continue to remain as a top-priority whilst sustainability and service are ensured.

Lastly, considering the Packet Loss Ratio, the observation from the Figures 5.65, (also see **Appendix** P), suggests that, although some combinations rendered the network robust, they were all susceptible to packet loss but in varying degrees. SchedTypes combinations with BE and nrtPS proved to be the least susceptible combinations to packet loss. This implies that the network was more robust in providing non-real-time services than it was for real-time services except for a combination of BE with MBQOS. This was expected because in MBQOS scheduler the bandwidth requests for BE service are stored in a low priority queue as explained by Ismail *et al.* (2010). It is worth noting that in the combinations of BE and nrtPS the Packet Loss Ratio proved to be more consistent across all the flows. This also includes the combination of MBQOS with; although the Packet Loss Ratio proved to be very high there was still some consistency in the ratio. This provides positive implication in that with these types of combinations, it is much easier to predict the rate at which packets would be lost. This understanding helps to draw an inference to say, the Packet Loss Ratio proved to be very low with these two combinations, notwithstanding the exception of MBQOS combination with BE. What sets aside the CoS's, BE and nrtPS, from rtPS and UGS is that their ratio is characterized by lack or no variation than is the case with their real-time service-oriented CoS counterparts. With the observed variations, it becomes a more complex issue to draw an inference to either say, the Packet Loss Ratio in rtPS and UGS CoS's is relatively low or high when combined with SchedType(s) X, except for a scenario where rtPS is combined with SchedType rtPS. The observation suggests that with rtPS and UGS the variation is such that in a certain flow, at times, only one packet can be lost and in the next flow almost all packets would be lost. Therefore, these two combinations are risky and unreliable to use and further compromise the robustness of the network that its traffic and service requirements would be satisfied.

The overall assessment of the CoS, GoS and QoS of the WiMAX network observed through performance bandwidth is that:

- The CoS's can utilize the available bandwidth towards providing the guaranteed QoS and GoS over the WiMAX network differently and thus either enhance or compromise the very same objective of a better network performance.
- The CoS's can perform to their optimum best when combined with certain SchedTypes, e.g. BE with FCFS, BE with rtPS, rtPS with MBQOS, and rtPS with FCFS, amongst others. This attest to the assertion that CoS alone cannot guarantee QoS on WiMAX network but for various applications the guaranteed QoS can be improved through proper selection of scheduling algorithms, Khoei et al (2014).
- The CoS's can perform poorly when combined with certain SchedTypes e.g. rtPS with rtPS and BE with MBQOS.
- Towards the provisioning of a robust WiMAX network, nrtPS can be a more flexible CoS even when combined with rtPS, a real-time service-oriented SchedType combination with a non-real-time service-oriented CoS.

5.6. Conclusion

In this chapter, the implementation process, the models and systems used and results thereof were presented accordingly. The WimaxNetDevice, WiMAX attributes and tracing of flows, amongst others, were presented as the underpinning models and systems for the implementation process and discussed into their greater details. Last but not least, the results of this research were presented in the form in which the research work was carried out. Twelve (12) scenarios were presented, each representing a combination of a scheduling type and CoS. In the next Chapter, an in-depth analysis and conclusive discussion of the results presented under this chapter will be conducted. Additionally, the next chapter will also present a brief discussion on the future works from this research project.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1. Introduction

In this chapter, the conclusion of the whole research work is hereby drawn with emphasis given to the evaluation work presented in chapter 5 vis-à-vis much of the work presented in chapter 2 detailing the research background, research context, research problems and research objectives, amongst other aspects. Furthermore, chapter 6 presents the problems encountered in undertaking and concluding this research project and also presents the identified research areas under various fields related to this research as the future work.

6.2. Addressing the Research Questions

This section presents the research questions and how they were addressed through the finding of this study as follows:

- **What is the impact of the problems caused by the inconsistent availability of bandwidth on the QoS, CoS and GoS on the Dwesa WiMAX network?**
  The experiments carried out in this research project depicted that the guaranteed QoS and GoS can be compromised due to ShedType(s) and CoS(s) used for a particular flow or network as a whole. The impact is such that some bandwidth requests may receive minimum prioritization while others may receive maximum prioritization. The average delay and average jitter might be strongly felt or just be minor to even notice depending on the availability of bandwidth to carry through the transmission of data packets. Similarly, PLR and throughput might be greatly increased or decreased if there is sufficient bandwidth available or bandwidth is utilized to its maximum, respectively. This is due to the unique characteristics each SchedType and CoS possesses to meet the guaranteed QoS and GoS. The nature of the impact is not always the same, it varies either from bad to worse or good to better in terms of network performance. This lies at the combination of the SchedType and CoS employed as some SchedTypes tend to prioritize certain services over others which is the same account for the CoSs. To address this situation is through finding a balanced combination of the two (SchedTypes and CoSs) for a particular nature of the network traffic that will better accommodate the inconsistent availability of bandwidth.

- **What mechanisms could be employed in optimizing the Dwesa WiMAX network to improve user experience when traffic congestion is increased due to inhabitants' training activities?**

  Finding a best combination of SchedType and CoS has proved to be a preferable mechanism to employ in the Dwesa WiMAX network for performance optimization during traffic congestion. The experiments carried out depicted that given a certain amount of data packets to be transmitted over the network- increased throughput, decreased PLR, reduced average delay and average jitter can be achieved depending on the SchedType and CoS combination used. Such combinations include BE with FCFS, BE with rtPS, rtPS with MBQOS, rtPS with FCFS and nrtPS with rtPS. These combinations can be used as the required mechanisms for performance optimization of the Dwesa WiMAX network to accommodate increased traffic congestion.

- **What cost-effective approach, suitable for the Dwesa WiMAX network, could be developed with the aim to optimize the network?**

  The SchedTypes and CoSs forms part of the WiMAX package already deployed in Dwesa and do not require to be purchased again as opposed to the notion of costly availing more bandwidth for the network. It could be of advantage, in terms of cost, to utilize the available bandwidth through employing balanced combinations of the existing SchedTypes and CoSs.

- **What are tangible possibilities does the Dwesa network peculiar activities (ICT4D) and conditions have for a prolonged period of time to ensure guaranteed QoS and improved GoS?**

  The combinations of BE with FCFS, BE with rtPS, rtPS with MBQOS, rtPS with FCFS and nrtPS with rtPS proved to be the tangible possibilities that the Dwesa network peculiar activities (ICT4D) and conditions have in order to meet the guaranteed QoS and GoS between flows. The possibilities are such that for each flow, a particular combination can be implemented; flow 1 implement BE with FCFS, flow 2 implement BE with rtPS, flow 3 implement rtPS with MBQOS and flow 4 implement nrtPS with rtPS, for instance. From this scenario, a combination that yields better network performance against the rest can be adopted and implemented across the network flows.

6.3. Addressing the Research Objectives

In context to the Dwesa WiMAX network and against the research objectives of this research work, the assessments provided in the previous section of this chapter can be interpreted and understood in the following manner:

- **To assess the impact and extent of the problems caused by the inconsistent availability of bandwidth on the QoS, CoS and GoS on the WiMAX Network deployed at Dwesa.**

  The QoS and GoS currently provided should be observed from an understanding of the current combination of the CoS and SchedType against the available bandwidth provision. The CoS and SchedType combination has a fundamental role in the bandwidth allocation mechanism and the problems caused by the inconsistent availability of bandwidth can be tracked down from this point. The reduction of the network problems caused by inconsistent availability of bandwidth lies at the implementation of a more robust combination of a CoS and SchedType.

- **To provide a network optimization technique to accommodate inconsistent availability of bandwidth caused during traffic congestion as a result of user training activities.**

  The available bandwidth, as observed, appeared to be effectively utilized when implementing the combinations of SchedTypes with non-real-time service-oriented CoS's. The analysis of the Packet Loss Ratio suggests that BE and nrtPS caused the least packet losses except for a scenario when BE was combined with MBQOS, a scenario which can be circumvented by simple shying away from implementing it. This is contrary to the fact that bandwidth would still be utilized in real-time service-oriented CoS's, yet the Packet Loss Ratio would remain regrettably high. User training activities usually detail the processing of FTP, variable size data and web traffic, the non-real-time services. The best mechanism for this state remains as the implementation of nrtPS and BE. However, with the advancement of certain social networks, which now implement real-time video processing, a combination of nrtPS with rtPS SchedType or BE with rtPS would serve as the ideal mechanism. This would enable the network to give priority to redistribute the available bandwidth to such real-time videos and allocate the remaining bandwidth to other non-real-time service requests.

- **To recommend the suitable bandwidth cost-effective approach for the Dwesa WiMAX network on the basis of the research findings for the network optimization**. Understanding the computer literacy levels of the inhabitants and towards the provisioning of a robust WiMAX network; implementing the nrtPS CoS with rtPS SchedType can prove to be an ideal setup to expurgate bandwidth provisioning costs whilst living by the objectives of providing a better network performance.

- **To assess how the Dwesa WiMAX network peculiar activities (ICT4D) and conditions can be addressed in a prolonged period of time.**
  Understanding the peculiarity of the activities and conditions in Dwesa WiMAX network, various combinations, bias to the non-real-time service-oriented CoS's, can be implemented from time-to-time in order to meet the user's demands.

6.4. Problems Encountered

A number of problems were encountered during the implementation of this research project, ranging from general to technical. The general problems encountered included the travelling from the University to the research site for studying the Dwesa WiMAX network setup which would later be emulated in the laboratory environment. Travelling to the research site would require a lot of physical strength as the distance was very long since the research site was deeply located in RMA of Dwesa.

The lack of enough technical detail about the Dwesa WiMAX network setup proved to be a stumbling block in getting the project started at a faster pace and assurance. The researcher had to rely on verbal information or simply start from scratch and gather his own technical detail about the network itself. Technically, as the Dwesa WiMAX network is a live network environment on which the inhabitants rely, access to perform certain technical changes or adjustments was prohibited, living the researcher with only implementing all the tests from the laboratory environment. Also, the crashing and fixing process of the WiMAX BS of the network proved to be too lengthy and thus contributed to the late conclusion of this research project.

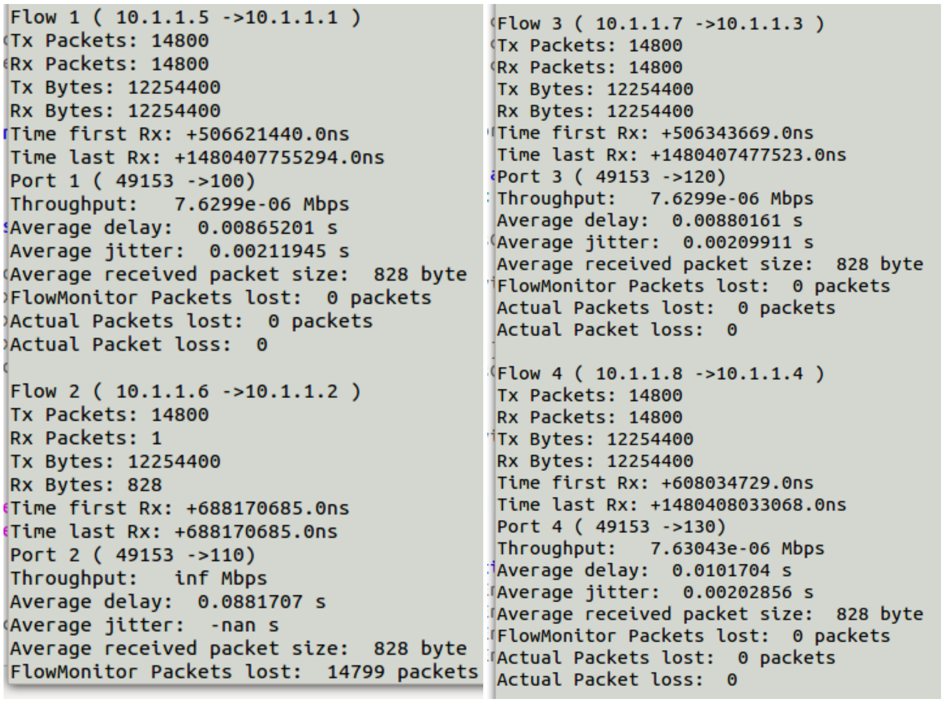
6.5. Future Work

Effectively, the implementation of the results and recommendations of this research project to the live Dwesa WiMAX network remains as the immediate future work. The results and recommendations of this research project can serve as an expert guidance on the idea to optimize the WiMAX network implemented down in Dwesa. Technically, this research work was based on the utilization of a fixed available bandwidth scenario in which a certain amount of data would be initialized for all the 12 scenarios implemented. Inversely, part of the future work can include a scenario in which packets could be transmitted with the amount of bandwidth they require and observe which CoS and SchedType combination would be cost-effective in terms of bandwidth. Another future work can include a study to determine the cost of each loss of the packet in the network as some bandwidth is consumed by the packets that eventually get discarded due to various factors. Finally, a study to enhance the provisioning of the QoS and GoS of the Dwesa WiMAX network using the IP and Ethernet efforts and mapping protocols in an expanded WiMAX network serves as a part of the future work.

6.6. Conclusion

This chapter began by providing a series of conclusive assessments and analysis of the twelve (12) scenarios created to monitor the WiMAX network performance using the Average Delay, Average Jitter, Throughput and Packet Loss Ratio as metrics. A further discussion on their implication against the guaranteed QoS and GoS provided by the network was presented. A reflection on the research objectives based on the outcomes of the results obtained was also presented. The chapter concluded by highlighting the future works as suggested by the results of this research project.

Appendix A

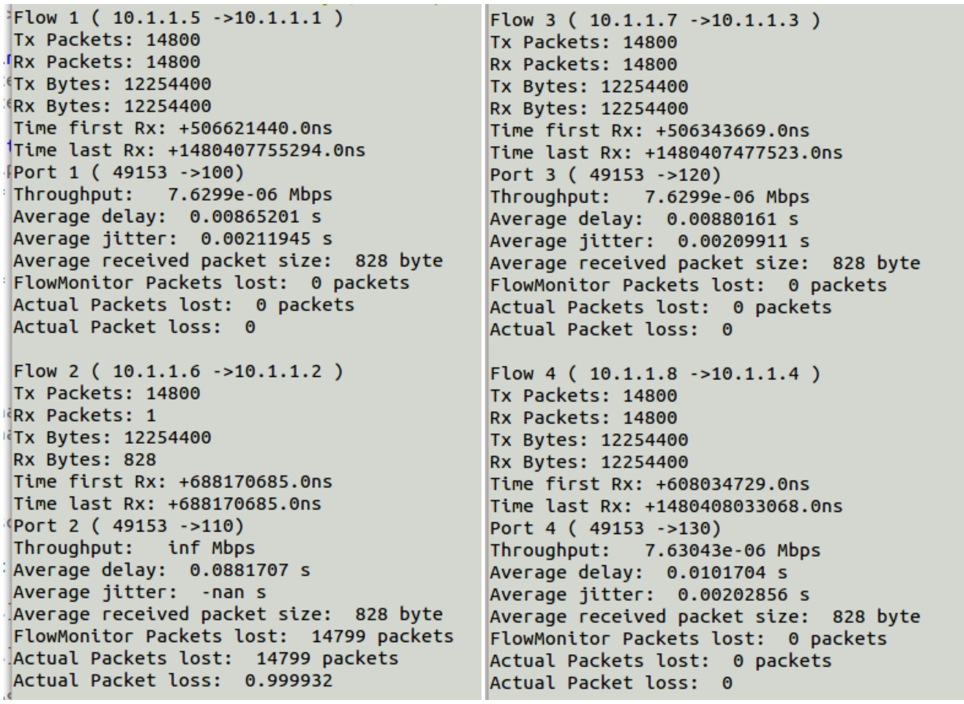


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +506621440.0ns
Time last Rx: +1480407755294.0ns
Port 1 ( 49153 ->100)
Throughput:   7.6299e-06 Mbps
Average delay:  0.00865201 s
Average jitter:  0.00211945 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 1
Tx Bytes: 12254400
Rx Bytes: 828
Time first Rx: +688170685.0ns
Time last Rx: +688170685.0ns
Port 2 ( 49153 ->110)
Throughput:   inf Mbps
Average delay:  0.0881707 s
Average jitter:  -nan s
Average received packet size:  828 byte
FlowMonitor Packets lost:  14799 packets
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +506343669.0ns
Time last Rx: +1480407477523.0ns
Port 3 ( 49153 ->120)
Throughput:   7.6299e-06 Mbps
Average delay:  0.00880161 s
Average jitter:  0.00209911 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +608034729.0ns
Time last Rx: +1480408033068.0ns
Port 4 ( 49153 ->130)
Throughput:   7.63043e-06 Mbps
Average delay:  0.0101704 s
Average jitter:  0.00202856 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0
```

The Results of UGS and MBQOS Scenario

Appendix B

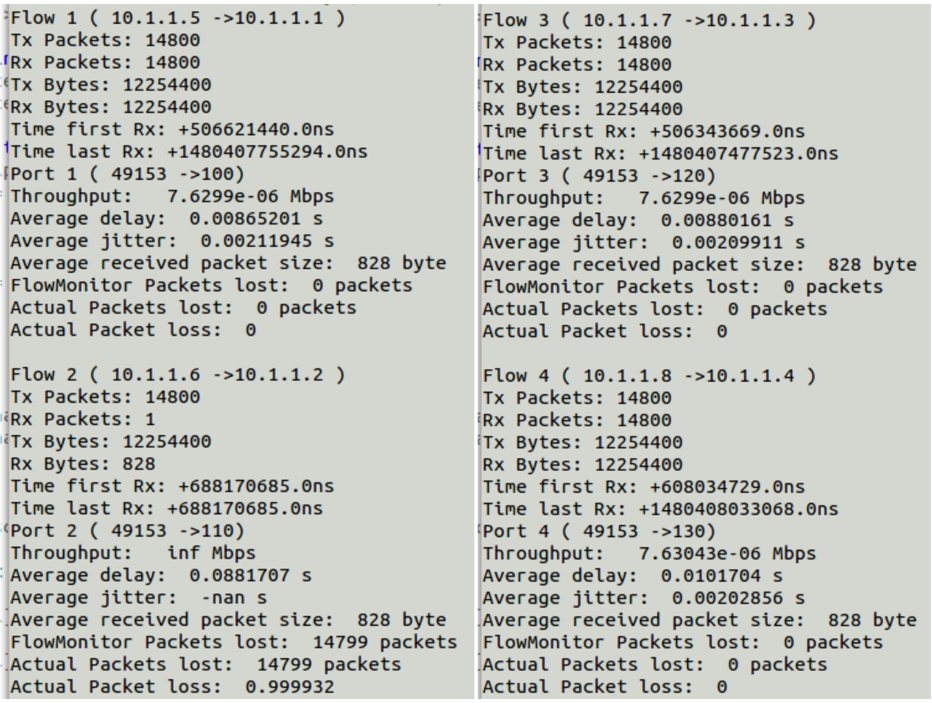


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +506621440.0ns
Time last Rx: +1480407755294.0ns
Port 1 ( 49153 ->100)
Throughput:   7.6299e-06 Mbps
Average delay:  0.00865201 s
Average jitter:  0.00211945 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 1
Tx Bytes: 12254400
Rx Bytes: 828
Time first Rx: +688170685.0ns
Time last Rx: +688170685.0ns
Port 2 ( 49153 ->110)
Throughput:   inf Mbps
Average delay:  0.0881707 s
Average jitter:  -nan s
Average received packet size:  828 byte
FlowMonitor Packets lost:  14799 packets
Actual Packets lost:  14799 packets
Actual Packet loss:  0.999932
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +506343669.0ns
Time last Rx: +1480407477523.0ns
Port 3 ( 49153 ->120)
Throughput:   7.6299e-06 Mbps
Average delay:  0.00880161 s
Average jitter:  0.00209911 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +608034729.0ns
Time last Rx: +1480408033068.0ns
Port 4 ( 49153 ->130)
Throughput:   7.63043e-06 Mbps
Average delay:  0.0101704 s
Average jitter:  0.00202856 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0
```

The Results of UGS and FCFS Scenario

Appendix C

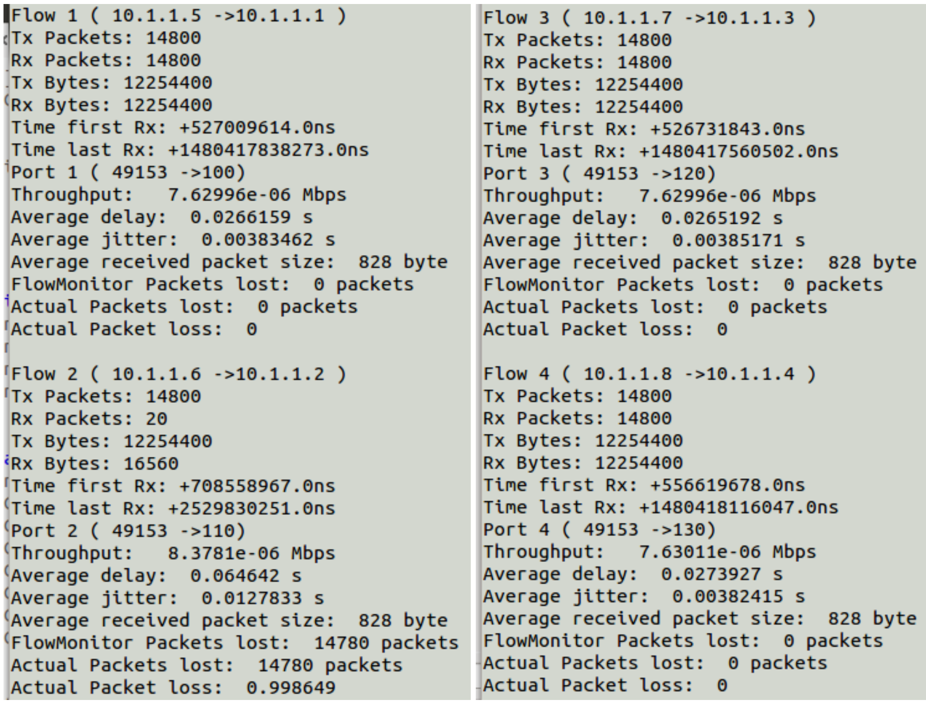


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +506621440.0ns
Time last Rx: +1480407755294.0ns
Port 1 ( 49153 ->100)
Throughput:   7.6299e-06 Mbps
Average delay:  0.00865201 s
Average jitter:  0.00211945 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 1
Tx Bytes: 12254400
Rx Bytes: 828
Time first Rx: +688170685.0ns
Time last Rx: +688170685.0ns
Port 2 ( 49153 ->110)
Throughput:   inf Mbps
Average delay:  0.0881707 s
Average jitter:  -nan s
Average received packet size:  828 byte
FlowMonitor Packets lost:  14799 packets
Actual Packets lost:  14799 packets
Actual Packet loss:  0.999932
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +506343669.0ns
Time last Rx: +1480407477523.0ns
Port 3 ( 49153 ->120)
Throughput:   7.6299e-06 Mbps
Average delay:  0.00880161 s
Average jitter:  0.00209911 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +608034729.0ns
Time last Rx: +1480408033068.0ns
Port 4 ( 49153 ->130)
Throughput:   7.63043e-06 Mbps
Average delay:  0.0101704 s
Average jitter:  0.00202856 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0
```

The Results of UGS and RTPS Scenario

Appendix D

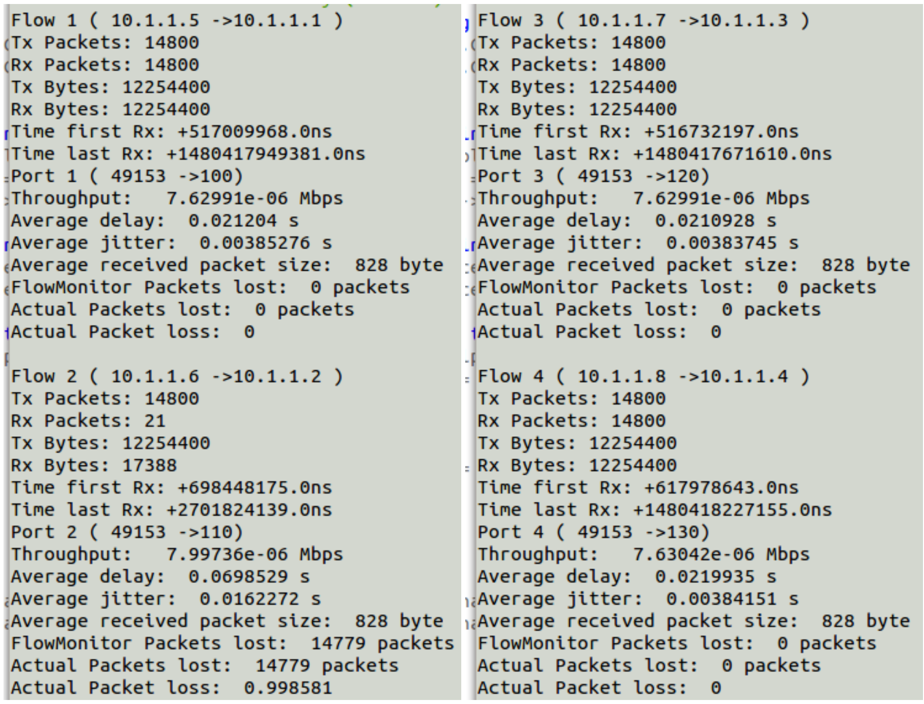


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +527009614.0ns
Time last Rx: +1480417838273.0ns
Port 1 ( 49153 ->100)
Throughput:   7.62996e-06 Mbps
Average delay:  0.0266159 s
Average jitter:  0.00383462 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 20
Tx Bytes: 12254400
Rx Bytes: 16560
Time first Rx: +708558967.0ns
Time last Rx: +2529830251.0ns
Port 2 ( 49153 ->110)
Throughput:   8.3781e-06 Mbps
Average delay:  0.064642 s
Average jitter:  0.0127833 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  14780 packets
Actual Packets lost:  14780 packets
Actual Packet loss:  0.998649
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +526731843.0ns
Time last Rx: +1480417560502.0ns
Port 3 ( 49153 ->120)
Throughput:   7.62996e-06 Mbps
Average delay:  0.0265192 s
Average jitter:  0.00385171 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +556619678.0ns
Time last Rx: +1480418116047.0ns
Port 4 ( 49153 ->130)
Throughput:   7.63011e-06 Mbps
Average delay:  0.0273927 s
Average jitter:  0.00382415 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0
```

The Results of rtPS and MBQOS Scenario

Appendix E

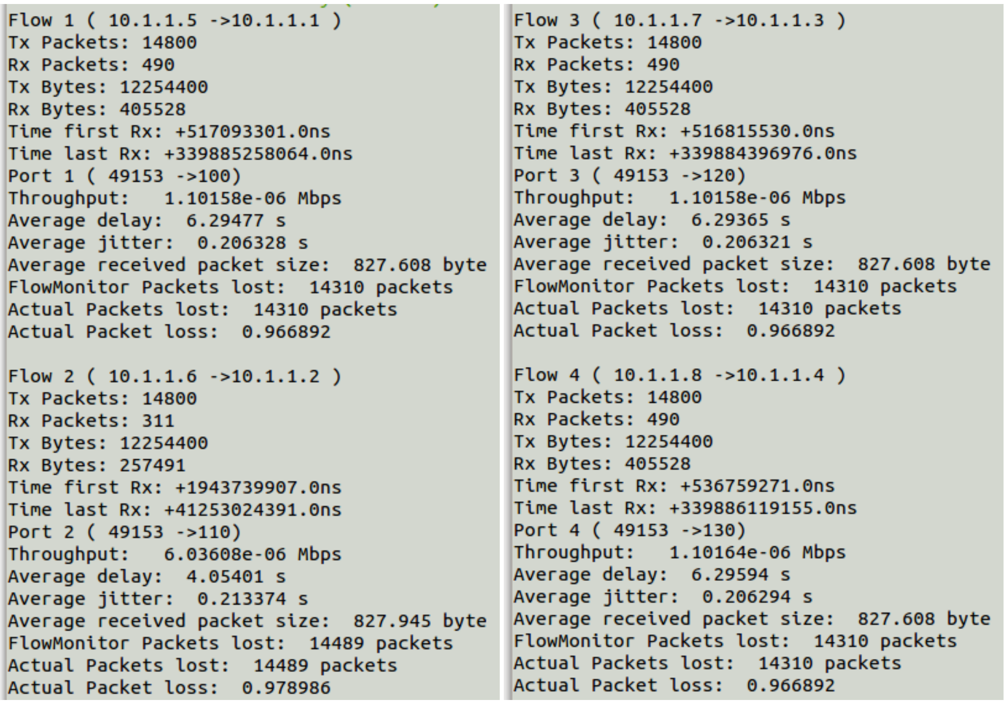


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +517009968.0ns
Time last Rx: +1480417949381.0ns
Port 1 ( 49153 ->100)
Throughput:   7.62991e-06 Mbps
Average delay:  0.021204 s
Average jitter:  0.00385276 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 21
Tx Bytes: 12254400
Rx Bytes: 17388
Time first Rx: +698448175.0ns
Time last Rx: +2701824139.0ns
Port 2 ( 49153 ->110)
Throughput:   7.99736e-06 Mbps
Average delay:  0.0698529 s
Average jitter:  0.0162272 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  14779 packets
Actual Packets lost:  14779 packets
Actual Packet loss:  0.998581
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +516732197.0ns
Time last Rx: +1480417671610.0ns
Port 3 ( 49153 ->120)
Throughput:   7.62991e-06 Mbps
Average delay:  0.0210928 s
Average jitter:  0.00383745 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +617978643.0ns
Time last Rx: +1480418227155.0ns
Port 4 ( 49153 ->130)
Throughput:   7.63042e-06 Mbps
Average delay:  0.0219935 s
Average jitter:  0.00384151 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0
```

The Results of rtPS and FCFS Scenario

Appendix F

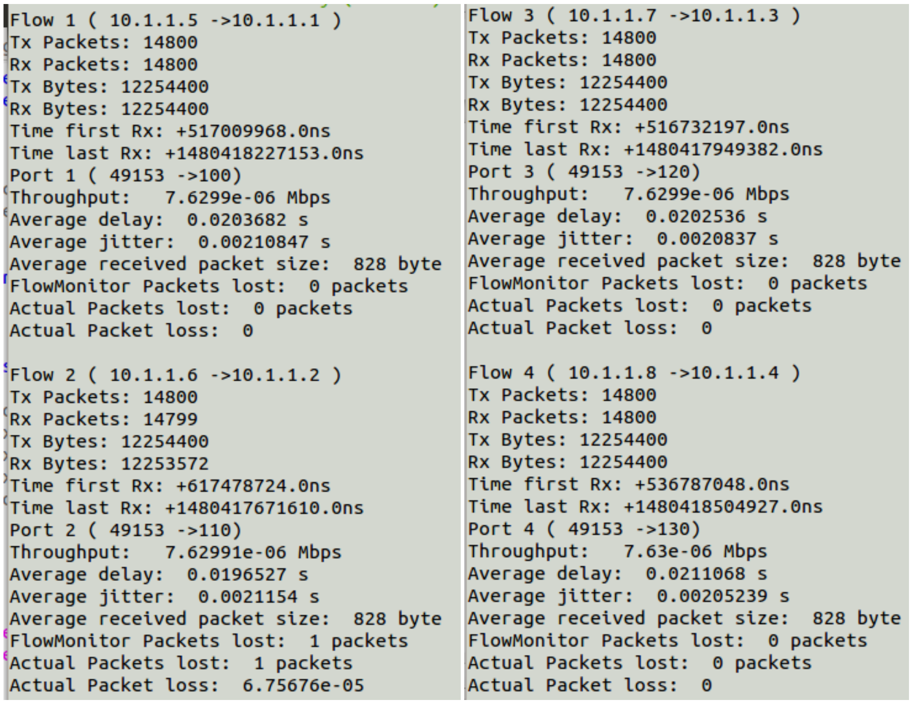


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 490
Tx Bytes: 12254400
Rx Bytes: 405528
Time first Rx: +517093301.0ns
Time last Rx: +339885258064.0ns
Port 1 ( 49153 ->100)
Throughput:   1.10158e-06 Mbps
Average delay:  6.29477 s
Average jitter:  0.206328 s
Average received packet size:  827.608 byte
FlowMonitor Packets lost:  14310 packets
Actual Packets lost:  14310 packets
Actual Packet loss:  0.966892

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 311
Tx Bytes: 12254400
Rx Bytes: 257491
Time first Rx: +1943739907.0ns
Time last Rx: +41253024391.0ns
Port 2 ( 49153 ->110)
Throughput:   6.03608e-06 Mbps
Average delay:  4.05401 s
Average jitter:  0.213374 s
Average received packet size:  827.945 byte
FlowMonitor Packets lost:  14489 packets
Actual Packets lost:  14489 packets
Actual Packet loss:  0.978986
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 490
Tx Bytes: 12254400
Rx Bytes: 405528
Time first Rx: +516815530.0ns
Time last Rx: +339884396976.0ns
Port 3 ( 49153 ->120)
Throughput:   1.10158e-06 Mbps
Average delay:  6.29365 s
Average jitter:  0.206321 s
Average received packet size:  827.608 byte
FlowMonitor Packets lost:  14310 packets
Actual Packets lost:  14310 packets
Actual Packet loss:  0.966892

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 490
Tx Bytes: 12254400
Rx Bytes: 405528
Time first Rx: +536759271.0ns
Time last Rx: +339886119155.0ns
Port 4 ( 49153 ->130)
Throughput:   1.10164e-06 Mbps
Average delay:  6.29594 s
Average jitter:  0.206294 s
Average received packet size:  827.608 byte
FlowMonitor Packets lost:  14310 packets
Actual Packets lost:  14310 packets
Actual Packet loss:  0.966892
```

The Results of rtPS and rtPS Scenario

Appendix G

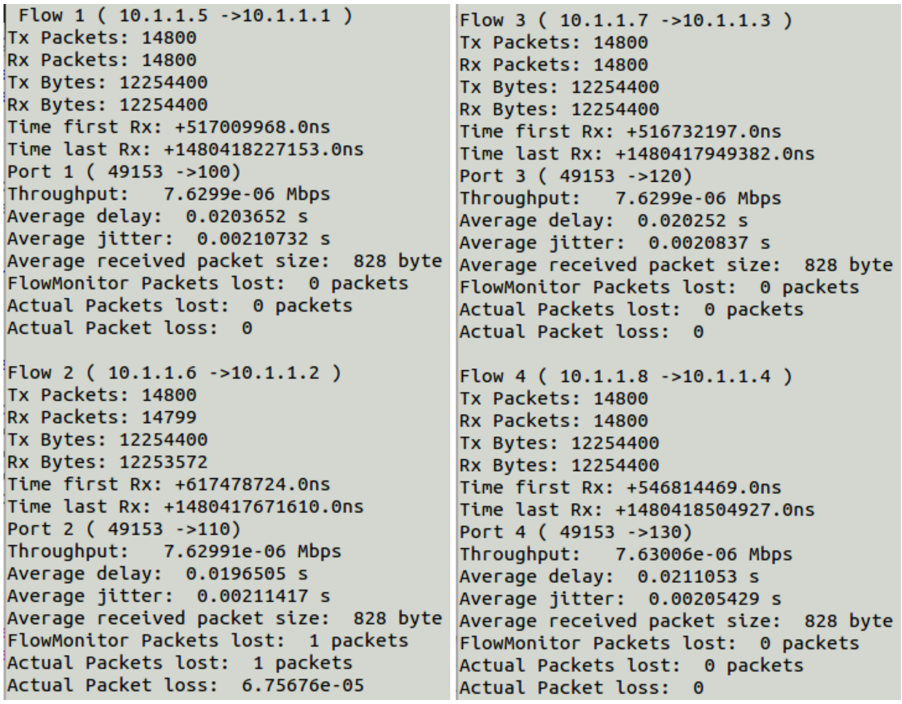


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +517009968.0ns
Time last Rx: +1480418227153.0ns
Port 1 ( 49153 ->100)
Throughput:   7.6299e-06 Mbps
Average delay:  0.0203682 s
Average jitter:  0.00210847 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 14799
Tx Bytes: 12254400
Rx Bytes: 12253572
Time first Rx: +617478724.0ns
Time last Rx: +1480417671610.0ns
Port 2 ( 49153 ->110)
Throughput:   7.62991e-06 Mbps
Average delay:  0.0196527 s
Average jitter:  0.0021154 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  1 packets
Actual Packets lost:  1 packets
Actual Packet loss:  6.75676e-05
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +516732197.0ns
Time last Rx: +1480417949382.0ns
Port 3 ( 49153 ->120)
Throughput:   7.6299e-06 Mbps
Average delay:  0.0202536 s
Average jitter:  0.0020837 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +536787048.0ns
Time last Rx: +1480418504927.0ns
Port 4 ( 49153 ->130)
Throughput:   7.63e-06 Mbps
Average delay:  0.0211068 s
Average jitter:  0.00205239 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0
```

The Results of nrtPS and MBQOS Scenario

Appendix H

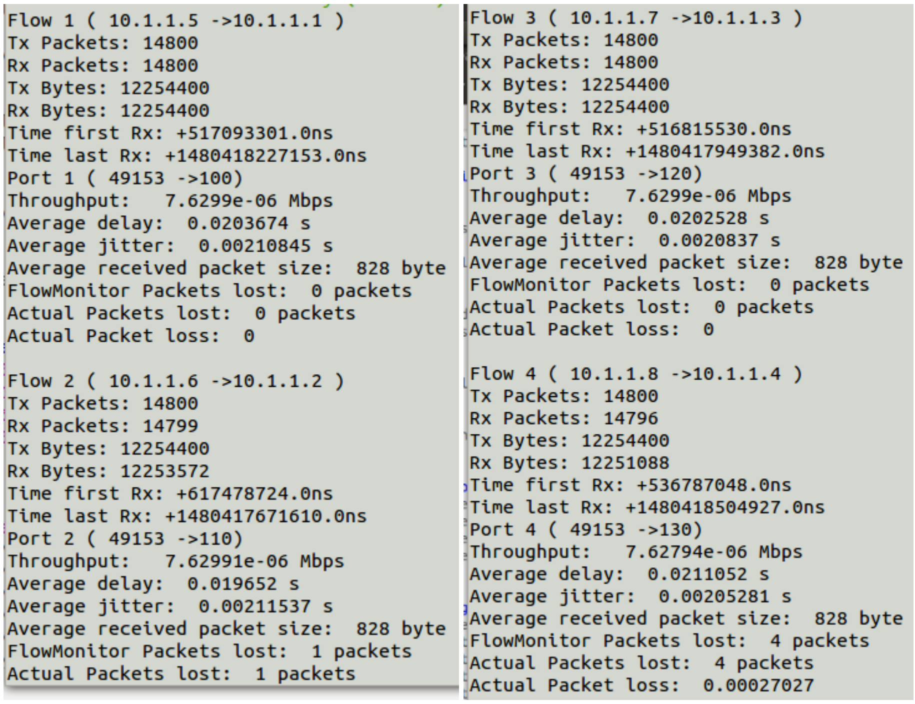


```
 Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +517009968.0ns
Time last Rx: +1480418227153.0ns
Port 1 ( 49153 ->100)
Throughput:   7.6299e-06 Mbps
Average delay:  0.0203652 s
Average jitter:  0.00210732 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 14799
Tx Bytes: 12254400
Rx Bytes: 12253572
Time first Rx: +617478724.0ns
Time last Rx: +1480417671610.0ns
Port 2 ( 49153 ->110)
Throughput:   7.62991e-06 Mbps
Average delay:  0.0196505 s
Average jitter:  0.00211417 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  1 packets
Actual Packets lost:  1 packets
Actual Packet loss:  6.75676e-05
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +516732197.0ns
Time last Rx: +1480417949382.0ns
Port 3 ( 49153 ->120)
Throughput:   7.6299e-06 Mbps
Average delay:  0.020252 s
Average jitter:  0.0020837 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +546814469.0ns
Time last Rx: +1480418504927.0ns
Port 4 ( 49153 ->130)
Throughput:   7.63006e-06 Mbps
Average delay:  0.0211053 s
Average jitter:  0.00205429 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0
```

The Results of nrtPS and FCFS Scenario

Appendix I

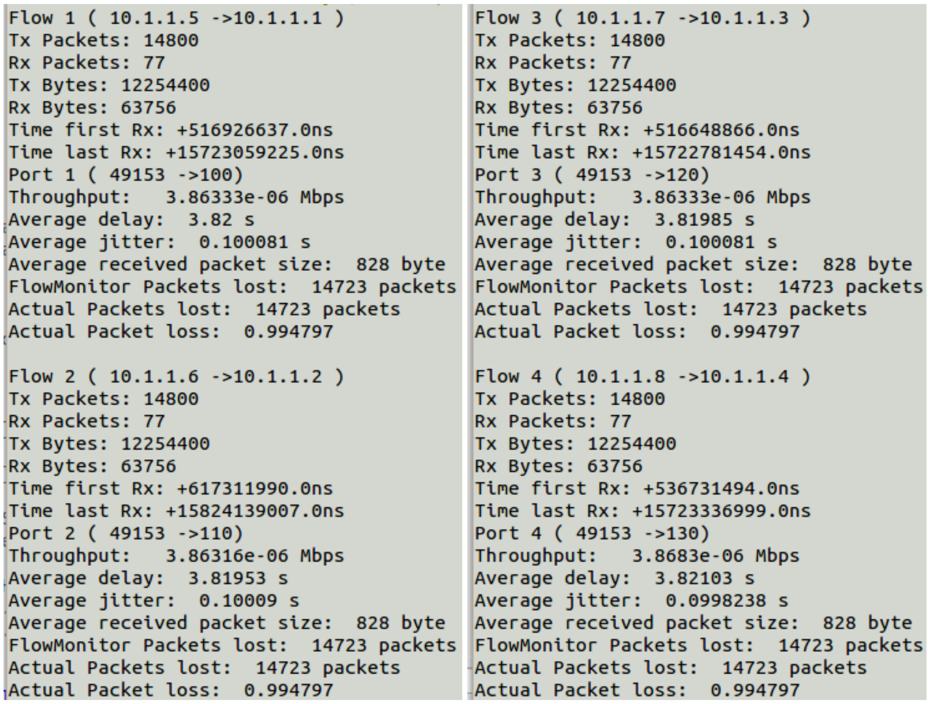


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +517093301.0ns
Time last Rx: +1480418227153.0ns
Port 1 ( 49153 ->100)
Throughput:   7.6299e-06 Mbps
Average delay:  0.0203674 s
Average jitter:  0.00210845 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 14799
Tx Bytes: 12254400
Rx Bytes: 12253572
Time first Rx: +617478724.0ns
Time last Rx: +1480417671610.0ns
Port 2 ( 49153 ->110)
Throughput:   7.62991e-06 Mbps
Average delay:  0.019652 s
Average jitter:  0.00211537 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  1 packets
Actual Packets lost:  1 packets
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +516815530.0ns
Time last Rx: +1480417949382.0ns
Port 3 ( 49153 ->120)
Throughput:   7.6299e-06 Mbps
Average delay:  0.0202528 s
Average jitter:  0.0020837 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14796
Tx Bytes: 12254400
Rx Bytes: 12251088
Time first Rx: +536787048.0ns
Time last Rx: +1480418504927.0ns
Port 4 ( 49153 ->130)
Throughput:   7.62794e-06 Mbps
Average delay:  0.0211052 s
Average jitter:  0.00205281 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  4 packets
Actual Packets lost:  4 packets
Actual Packet loss:  0.00027027
```

The Results of nrtPS and rtPS Scenario

Appendix J

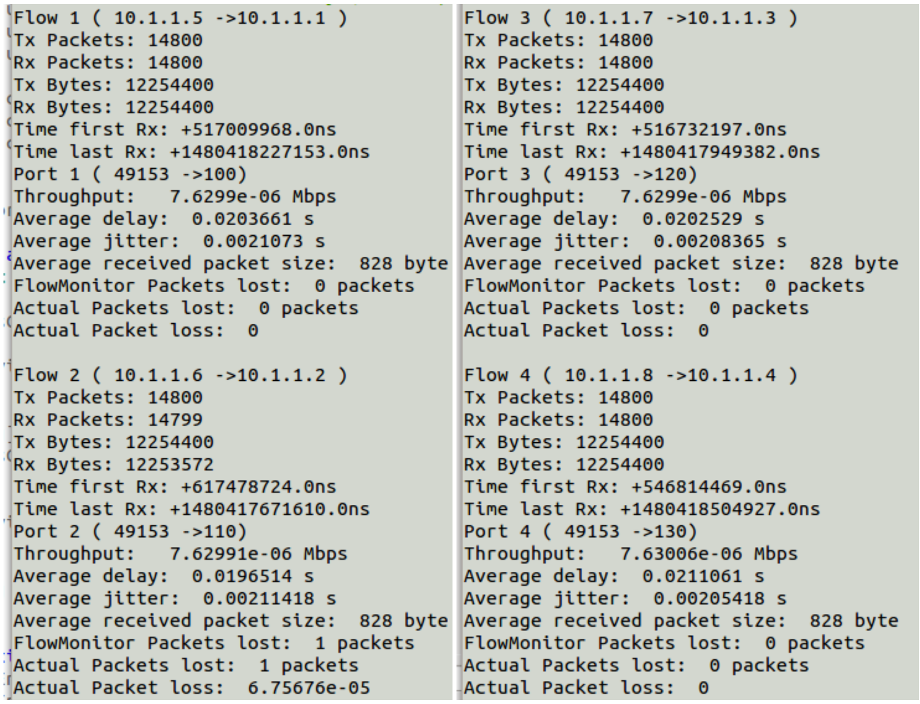


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 77
Tx Bytes: 12254400
Rx Bytes: 63756
Time first Rx: +516926637.0ns
Time last Rx: +15723059225.0ns
Port 1 ( 49153 ->100)
Throughput:   3.86333e-06 Mbps
Average delay:  3.82 s
Average jitter:  0.100081 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  14723 packets
Actual Packets lost:  14723 packets
Actual Packet loss:  0.994797

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 77
Tx Bytes: 12254400
Rx Bytes: 63756
Time first Rx: +617311990.0ns
Time last Rx: +15824139007.0ns
Port 2 ( 49153 ->110)
Throughput:   3.86316e-06 Mbps
Average delay:  3.81953 s
Average jitter:  0.10009 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  14723 packets
Actual Packets lost:  14723 packets
Actual Packet loss:  0.994797
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 77
Tx Bytes: 12254400
Rx Bytes: 63756
Time first Rx: +516648866.0ns
Time last Rx: +15722781454.0ns
Port 3 ( 49153 ->120)
Throughput:   3.86333e-06 Mbps
Average delay:  3.81985 s
Average jitter:  0.100081 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  14723 packets
Actual Packets lost:  14723 packets
Actual Packet loss:  0.994797

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 77
Tx Bytes: 12254400
Rx Bytes: 63756
Time first Rx: +536731494.0ns
Time last Rx: +15723336999.0ns
Port 4 ( 49153 ->130)
Throughput:   3.8683e-06 Mbps
Average delay:  3.82103 s
Average jitter:  0.0998238 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  14723 packets
Actual Packets lost:  14723 packets
Actual Packet loss:  0.994797
```

The Results of BE and MBQOS Scenario

Appendix K

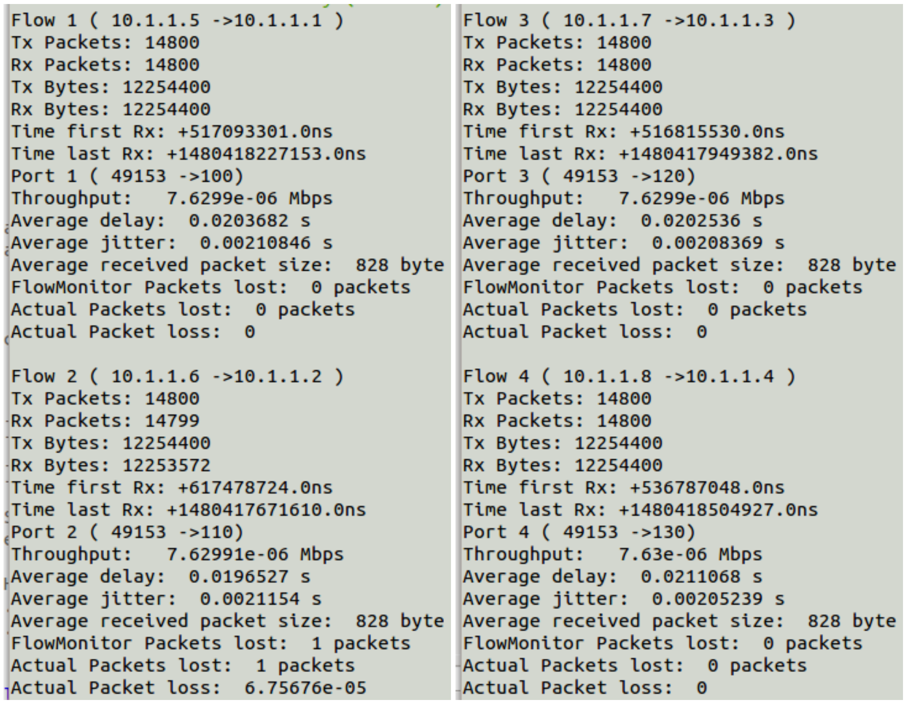


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +517009968.0ns
Time last Rx: +1480418227153.0ns
Port 1 ( 49153 ->100)
Throughput:   7.6299e-06 Mbps
Average delay:  0.0203661 s
Average jitter:  0.0021073 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 14799
Tx Bytes: 12254400
Rx Bytes: 12253572
Time first Rx: +617478724.0ns
Time last Rx: +1480417671610.0ns
Port 2 ( 49153 ->110)
Throughput:   7.62991e-06 Mbps
Average delay:  0.0196514 s
Average jitter:  0.00211418 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  1 packets
Actual Packets lost:  1 packets
Actual Packet loss:  6.75676e-05
```

```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +516732197.0ns
Time last Rx: +1480417949382.0ns
Port 3 ( 49153 ->120)
Throughput:   7.6299e-06 Mbps
Average delay:  0.0202529 s
Average jitter:  0.00208365 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +546814469.0ns
Time last Rx: +1480418504927.0ns
Port 4 ( 49153 ->130)
Throughput:   7.63006e-06 Mbps
Average delay:  0.0211061 s
Average jitter:  0.00205418 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0
```

The Results of BE and FCFS Scenario

Appendix L

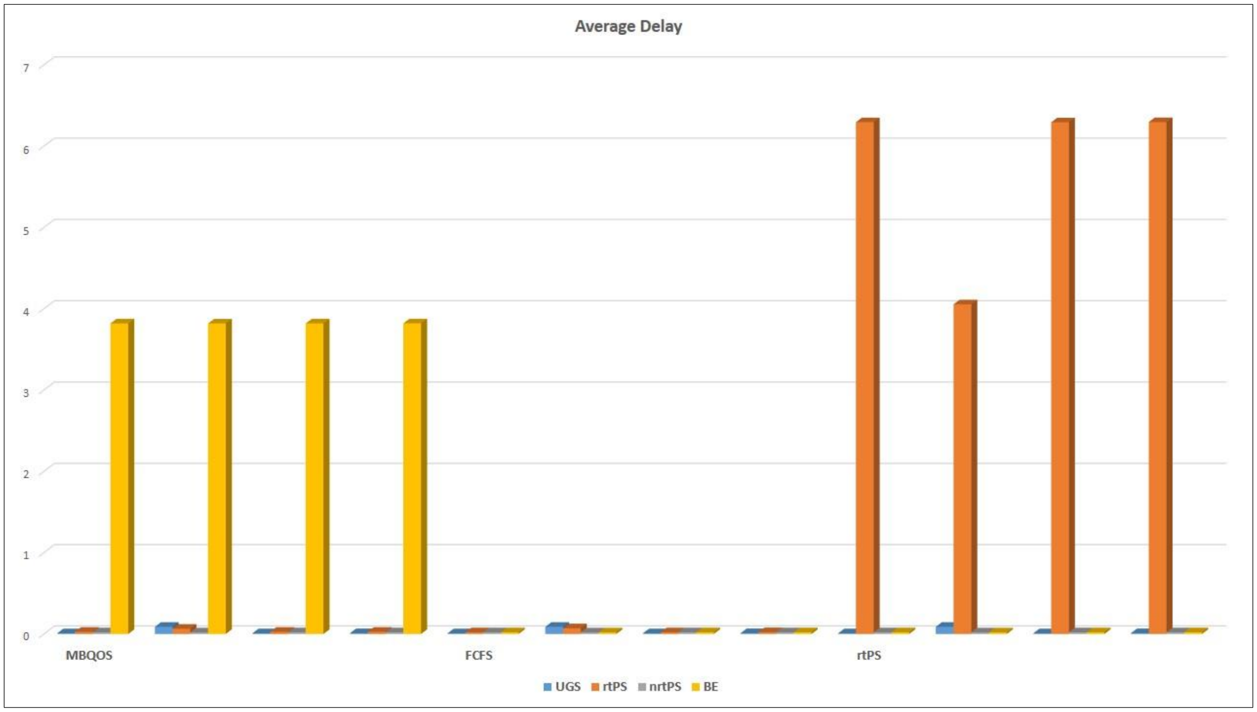


```
Flow 1 ( 10.1.1.5 ->10.1.1.1 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +517093301.0ns
Time last Rx: +1480418227153.0ns
Port 1 ( 49153 ->100)
Throughput:   7.6299e-06 Mbps
Average delay:  0.0203682 s
Average jitter:  0.00210846 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 2 ( 10.1.1.6 ->10.1.1.2 )
Tx Packets: 14800
Rx Packets: 14799
Tx Bytes: 12254400
Rx Bytes: 12253572
Time first Rx: +617478724.0ns
Time last Rx: +1480417671610.0ns
Port 2 ( 49153 ->110)
Throughput:   7.62991e-06 Mbps
Average delay:  0.0196527 s
Average jitter:  0.0021154 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  1 packets
Actual Packets lost:  1 packets
Actual Packet loss:  6.75676e-05
```

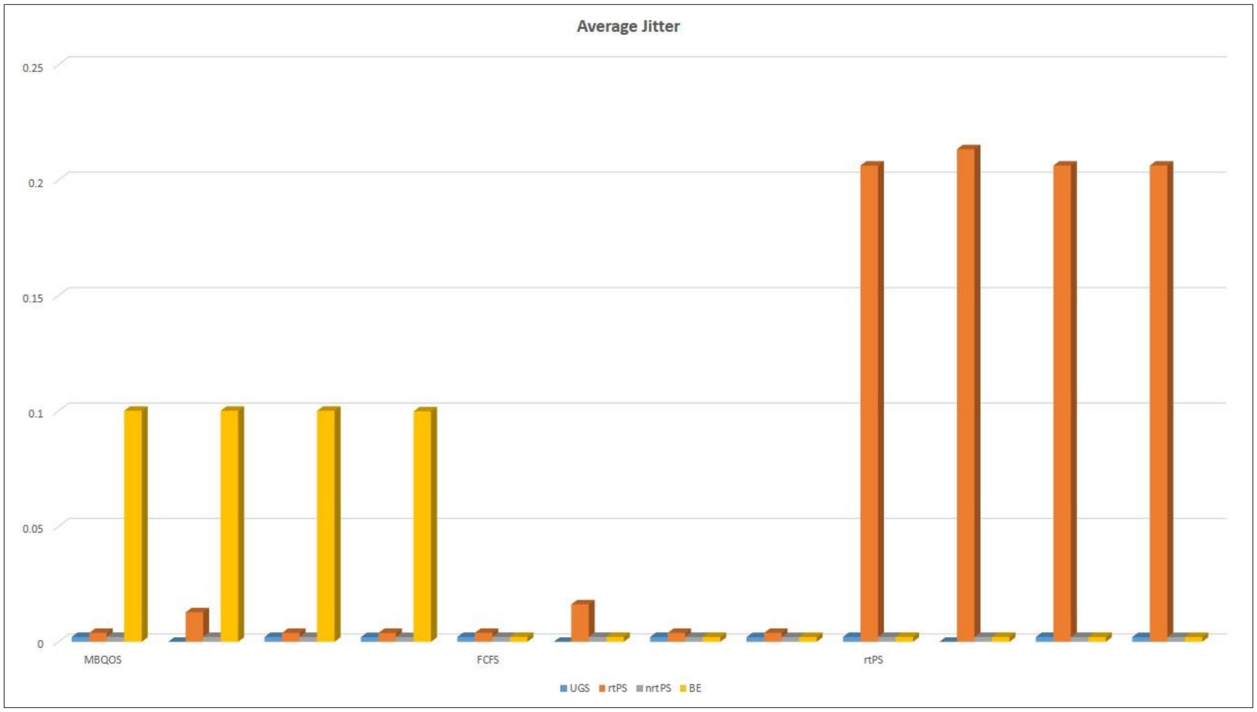
```
Flow 3 ( 10.1.1.7 ->10.1.1.3 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +516815530.0ns
Time last Rx: +1480417949382.0ns
Port 3 ( 49153 ->120)
Throughput:   7.6299e-06 Mbps
Average delay:  0.0202536 s
Average jitter:  0.00208369 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0

Flow 4 ( 10.1.1.8 ->10.1.1.4 )
Tx Packets: 14800
Rx Packets: 14800
Tx Bytes: 12254400
Rx Bytes: 12254400
Time first Rx: +536787048.0ns
Time last Rx: +1480418504927.0ns
Port 4 ( 49153 ->130)
Throughput:   7.63e-06 Mbps
Average delay:  0.0211068 s
Average jitter:  0.00205239 s
Average received packet size:  828 byte
FlowMonitor Packets lost:  0 packets
Actual Packets lost:  0 packets
Actual Packet loss:  0
```

The Results of BE and rtPS Scenario

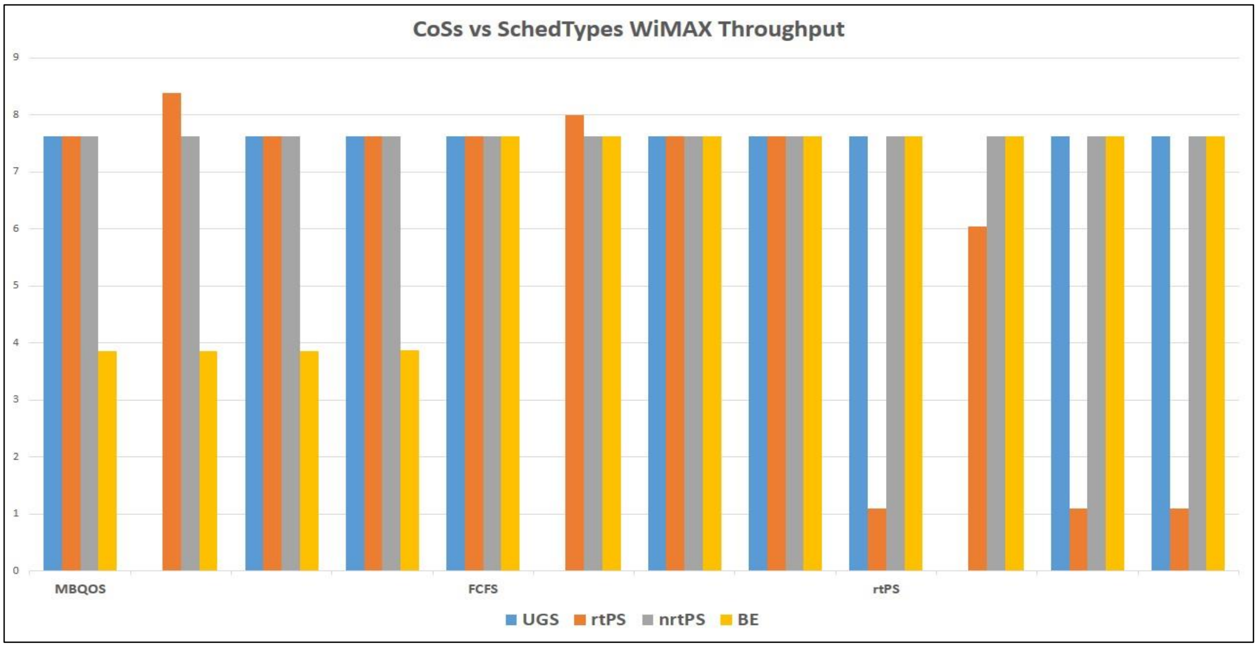
Appendix M



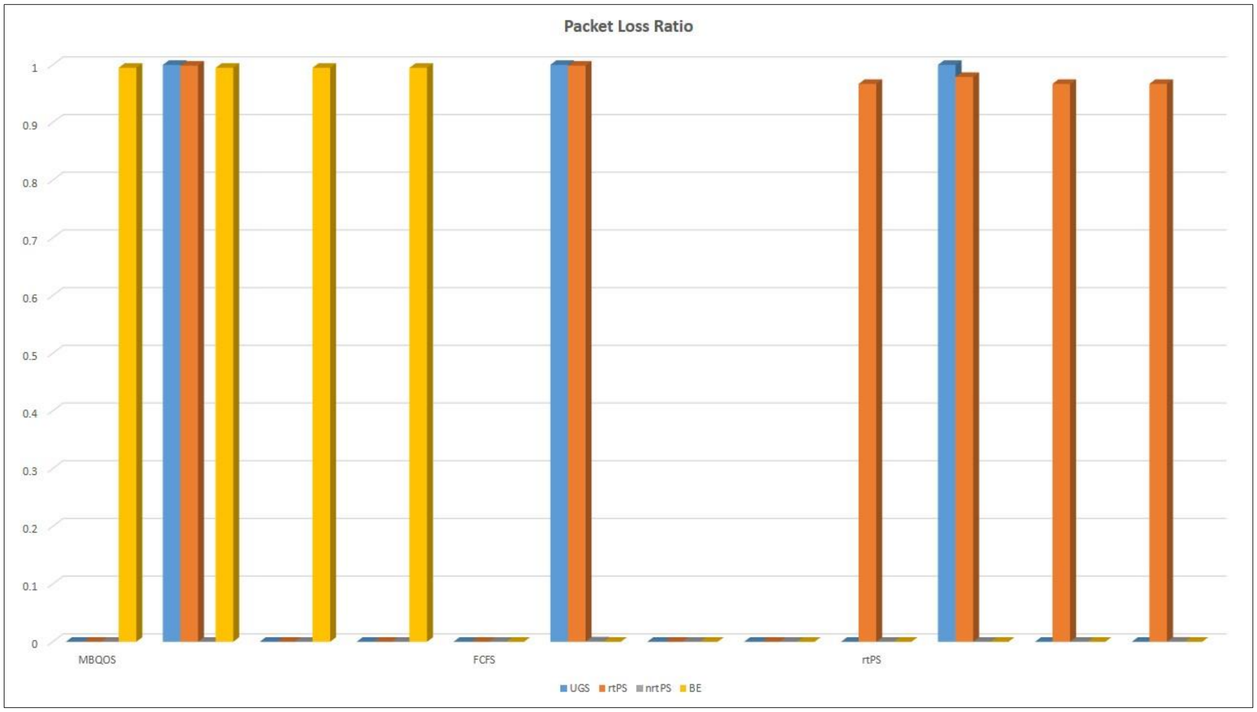
Appendix N



Appendix O



Appendix P


Packet Loss Ratio
1
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
MBQOS
FCFS
rtPS
UGS
rtPS
nrtPS
BE

References

2016 (1995) *Ng_netflow*. Available at: http://www.freebsd.org/cgi/man.cgi?query=ng_netflow&

Afzali, M., Khatibi, V. and Harouni, M. (2010).“Connection availability analysis in the wimax mesh networks.” In *Computer and Automation Engineering (ICCAE), 2010 The 2nd International Conference on*(Vol. 5, pp. 699-703). IEEE.

Ahmad, I. and Habibi, D.(2010).“Call admission control scheme for improved quality of service in wimax communication at vehicular speeds.”In *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*(pp. 1-6).IEEE.

Ahmadi, S. (2010). “Mobile WiMAX: A systems approach to understanding IEEE 802.16 m radio access technology.”Academic Press.

Ali, S. andAli, A. (2010) “Performance Analysis of AODV, DSR and OLSR in MANET” *Department of Electrical Engineering with emphasis on Telecommunication Blekinge Institute of Technology*, Sweden 2009

Anouari, T. and Haqiq, A. (2012)."Performance Analysis of VoIP Traffic in WiMAX using Various Service Classes",*International Journal of Computer Applications* (0975 -8887), Volume 52, No. 20, August.

Awal, M.A. and Boukhatem, L. (n.d). “Wimax and end-to-end qos support. *Radio Resources Management in WiMAX: From theoretical capacity to system simulations*,” pp.105-138.

Bakharev, A. (2010). “Simulator-Controlled Real-World Experiments InMultihop Wireless Mesh Networks.” *Lulea University of Technology Dept. of Computer Science and Electrical Engineering Communication Networks Research Group*

Belghith, A. and Nuaymi, L.(2008).“Design and implementation of a QoS-included WiMAX module for NS-2 simulator".In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops* (p. 28). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering

Carvalho, T., Júnior, J. J., Warley Valente, C. N., Francês, R., and Dias, K. L. (2013).“A Mobile WiMAX Mesh Network with Routing Techniques and Quality of Service Mechanisms.”

Chaudhari, M. K., andKarule, M. P. (2014). “WiMAX network model performance testing using NS3 Simulation software.” *International Journal of Application or Innovation in Engineering & Management (IJAIEM), Volume 03, Issue 09, September 2014, pp.223-230. ISSN 2319 - 4847*

Chauhan, M., Choubey, R. and Soni, R. (2013).“An Advanced Scheduler for HybridMbqos in WiMAX Network.” *International Journal of Computer Technology and Electronics Communication,* Will published volume-2, issue-3, June-2013

Cicconetti, C., Lenzini, L. and Mingozzi, E.(2010).“A bandwidth request reiteration mechanism for IEEE 802.16 wireless networks.” *Wireless Networks*, *16*(3), pp.731-742.

Cicconetti, C., Lenzini, L., Mingozzi, E., and Eklund, C. (2006).“Quality of service support inIEEE 802.16 networks,” Network IEEE, vol. 20, Issue 2, pp.50-55, Mar.-Apr. 2006.

Eklund, C., Marks, R. B., Stanwood, K. L. and Wang, S. (2002). “IEEE Standard 802.16: a technical overview of the WirelessMAN air interface for broadband wireless access,” *IEEE Communications Magazine*, pp. 98-107,June 2002.

Farooq, J. and Turletti, T. (2009).“An IEEE 802.16 wimax module for the ns-3 simulator.”In *Proceedings of the 2nd International Conference on Simulation Tools and Techniques* (p. 8). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Fernando, W.A.C., Rajatheva, R.M.A.P. and Ahmed, K.M. (1998).“Performance of coded OFDM with higher modulation schemes.”In*Communication Technology Proceedings, 1998.ICCT'98.1998 International Conference on* (Vol. 2, pp. 5-pp).IEEE.

Fraenkel, J.R., Wallen, N.E. and Hyun, H.H.(1993). “How to design and evaluate research in education.” (Vol. 7). New York: McGraw-Hill.

Google code archive - long-term storage for Google code project hosting (no date) Available at: https://code.google.com/p/softflowd (Accessed: 19 July 2016).

Gray, D. (2007). “A comparative analysis of mobile WiMAX deployment alternatives in the access network.”In *WiMAX Forum*.

Graziani, R. and Johnson, A. (2007). “Routing Protocols and Concepts, CCNA exploration companion guide.”*Cisco Press*.

Gupta, V. K., MuntaserA.F.A, and Mohamed S.A.H. (2012)."Grade of Service in End-To-End Service Quality of Service Broadband Networks." *International Journal of Advanced Research in Computer Science and Software Engineering,*Volume 2,pp. 148-151, Issue 12, December 2012 ISSN: 2277 128X

Haider, A. and Harris, R.J. (2009).“Scheduling in WiMAX.” *WiMAX Network Planning and Optimization*, p.19.

Inayatullah, M., Ullah, F. K., & Khan, A. N. (2006).“An Automated Grade Of Service Measuring System.”In *Emerging Technologies, 2006.ICET'06. International Conference on* (pp. 230-237). IEEE.

Islam, S., Rashid, M., &Tarique, M. (2011).Performance analysis of WiMax/WiFi system under different codecs. *literatures*, *15*, 16.

Ismail, M. A., Piro, G., Grieco, L. A., and Turletti, T. (2010). An improved IEEE 802.16 WiMAX module for the ns-3 simulator. In *Proceedings of the 3rd International ICST Conference on Simulation Tools and Techniques* (p. 63). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

Kaarthick, B., Yeshwenth, V. J., andNagarajan, N. (2009).“Performnance analysis of Video Conferencing and Multimedia application Services over WiMAX.”In *Advance Computing Conference, 2009.IACC 2009. IEEE International* (pp. 1109-1113). IEEE.

Kafhali, S. E., Bouchti, A. E., Hanini, M., &Haqiq, A. (2012).“Performance Analysis for Bandwidth Allocation in IEEE 802.16 Broadband Wireless Networks using BMAP Queueing.” *arXiv preprint arXiv:1203.1757*.

Kannisto, I., Hämäläinen, T., Joutsensalo, J., &Wallenius, E. (2007, April).“Adaptive Scheduling Method for WiMAXBasestations.” In *The 13th European Wireless Conference, EW2007, Paris*.

Khatkar, A., Singh, Y. and Nandal, V. (2013).“A QoS Oriented Analysis of ertPS, rtPS and nrtPS flows in WiMAX.” *International Journal of Advanced Research in Computer Science and Software Engineering*, *3*(7), pp.85-90.

Khoei, A. H., Mirjalily, G., &Sarram, M. A. (2014).“Improving The Quality Of Services By Scheduling Algorithms In Wimax Networks.” *Indian J. Sci. Res*, *7*(1), 1174-1184.

Khosroshahy, M., Vivien, N., and Philippe, G. (2006)."A study of WiMaxQoS mechanisms." *RMOB Project, Telecom Paris*.

Kunjuzwa, D. T., andThinyane, M. (2011). “Exploring user-driven telephony services in an information and communication technology for development context *(Doctoral dissertation, University of Fort Hare*).

Lansbergen, D. and Koolstra, K. (2002)."Analysis of Quality of Service Mechanisms in IP and Road Networks."

Leclerc, A., and Crosby, M. (2010). “Test and Evaluation of WiMAX Performance Using Open-Source Modeling and Simulation Software Tools.” SPACE AND NAVAL WARFARE SYSTEMS CENTER NORTH CHARLESTON SC.

Liu, C.H., Gkelias, A. and Leung, K.K. (2008).“Connection admission control and grade of service for QoS routing in mesh networks.” In*2008 IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications* (pp. 1-5). IEEE.M. Kas, B. Yargicoglu, I.

Ma, M. and Denko, M.K. eds.(2008). “Wireless quality of service: techniques, standards, and applications.” CRC Press..

Makkar, N. and Indu, I. (2014).“Review of Ad-Hoc on Demand Distance Vector and Optimized Link State Routing Protocol for Mobile Ad-Hoc Network.”International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 10, October 2014 ISSN: 2277 128X

Mehta, V. and Gupta, D.N.(2012).“Performance analysis of QoS parameters for WiMAX networks.” *International Journal of Engineering and Innovative Technology (IJEIT) Volume*, *1*, pp.105-110.

Mewara, H. S., andPorwal, S. (2013). “Throughput Performance Comparison of Reactive and Proactive Routing Protocols in Mobile Ad Hoc Networks using OPNET v14. 5.” *IJCA Proceedings on Recent Trends in Information Technology and Computer Science*, 9-14.

Mohamed, M. A., Zaki, F. W., &Mosbeh, R. H. (2010).“Simulation of WiMAX physical layer: IEEE 802.16 e.” *IJCSNS*, *10*(11), 49.

Mohan, S., Prabhakaran, A., and Suresh, M. (2013).“Evaluating The Performance of Wimax Using Proactive, Reactive and Hybrid Routing Protocols.” Transactions on Engineering and Sciences ISSN: 2347-1964 Online 2347-1875 Print Vol 1, Issue 4, November 2013

Ndlovu, N., Thinyane, M. and Terzoli, A.(2009).“Deployment and Extension of a Converged WiMAX/WiFi Network for Dwesa Community Area South Africa.”In *Southern African Telecommunication Networks and Applications Conference*.

NS-3: WiMAX models (no date) Available at: http://www.nsnam.org/docs/release/3.8/doxygen/group___wi_m_a_x.html (Accessed: 19 July 2016).

Paolini, M. and Consulting, S.F.(2007). “Building end-to-end WiMAX networks.” *SenzaFili Consulting*.

Pathak, S. (2013).“Performance Analysis of WiMAX PHY Layer under different channel conditions.” *arXiv preprint arXiv:1306.2425*.

Policy, I.-S.C. (2005) *IEEE SA - 802.16-2004 - standard for local and metropolitan area networks - part 16: Air interface for fixed Broadband wireless access systems*. Available at: https://standards.ieee.org/findstds/standard/802.16-2004.html (Accessed: 19 July 2016)

Policy, I.-S.C. (2007) *IEEE SA - 802.11-2007 - IEEE standard for information technology - telecommunications and information exchange between systems - local and metropolitan area networks - specific requirements - part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications*. Available at: https://standards.ieee.org/findstds/standard/802.11-2007.htm (Accessed: 7 September 2016)

Prasad, R., and Velez, F. J. (2010). "*WiMAX networks: techno-economic vision and challenges*."Springer Science & Business Media.

Ranga, M.M. and Terzoli, A.(2010)."Improving the robustness and effectiveness of the wireless telecommunication infrastructure in Dwesa South Africa."SATNAC.

RuhaniAb R., MurizahK., Cik Ku H., Che Ku Y., Mariamah, I. (2011). "Performance Analysis of Routing Protocol in WiMAX Network," IEEE International Conference on System Engineering and Technology (ICSET), 27-28 June 2011, pp. 153-157.

Saravanaselvi, P and Latha, P. (2012)"Weighted Fair Queue Scheduling Algorithm for IEEE802.16 Networks."*International Journal of Computer Applications* (IJCA) , International Conferenceon Recent Trends in Computational Methods, Communication and Controls (ICON3C 2012) Pp-ICON3C(8):5-8 , ISSN 0975-8887, 2012

Sharma, G., and Chawla, R. (2014)."Comparative Analysis of Quality of Service for Various Service Classes in WiMAX Network using NS-3." *International Journal of Computer Applications Technology and Research,3*(7), 427-433.

Shuaib, K.A. (2009)."A performance evaluation study of WIMAX using Qualnet."*Proceedings of the World Congress on Engineering*.Vol. 1. 2009.

Siebörger, I. and Terzoli,A. (2010). "WiMAX for rural SA: The experience of the Siyakhula Living Lab." *Southern African Telecommunication Networks & Applications Conference: A society enabled by Innovation and Applications, Stellenbosch, South Africa*. 2010.

Sin-seok, S., Joon-Myung, K., Nazim, A., John S. and James.W.H. (2011). "FAST: A Fuzzy- based Adaptive Scheduling Technique for IEEE 802.16 Networks." IntegratedNetwork Management (IM), 2011 IFIP/IEEE International Symposium at Department of Computer Science and Engineering, Pohang University of Scienceand Technology. (POSTECH), Pohang, South Korea,E-ISBN:978-1-4244-9220-6,Print ISBN:978-1-4244-9219-0,Pp-201 - 208 , 23-27 May 2011

Smirnoff, G., Motin, A., and Chernikov, A. (2 November 2012). FreeBSD Kernel Interfaces Manual.FreeBSD Man Pages.

So-In, C., Jain, R. and Al-Tamimi, A.-K. (2010). "A Scheduler for unsolicited Grant Service (UGS) in IEEE 802.16e mobile WiMAX networks", *IEEE Systems Journal*, 4(4), pp. 487–494. doi: 10.1109/jsyst.2010.2083171.

Tarwireyi, P., Terzoli, A. and Thinyane, M.(2007)."Implementation of an internet access cost management system for disadvantaged communities." *SATNAC 2007*.

Thomas, C. (2011). "NS-3 Simulation of WIMAX Networks" Department of Computer Science & Engineering - Washington University in St. Louis Campus Box 1045 - St. Louis, MO - 63130 - ph: (314) 935-6160

Thontadharya, H.J., Shwetha, D., Subramanyabhat, M. and Devaraju, J.T.(2011).“Performance Study of Bandwidth Request Mechanisms in IEEE 802.16 e Networks.” *International Journal of computer Science and Communication Networks*, *1*(2), pp.136-141.Toy, M. (2015). *Cable Networks, Services, and Management* (Vol. 13).John Wiley & Sons.

Wang, C., Yan, W.J. and Lo, H.K. (2012).“Dynamic admission control and bandwidth reservation for IEEE 802.16 e mobile WiMAX networks.” *EURASIP Journal on Wireless Communications and Networking*, *2012*(1), pp.1-20.

Xiaojing, M.(2007).“An efficient scheduling for diverse QoS requirements in WiMAX.”Waterloo, Ontario, Canada, 2007