EVALUATION OF THE EFFECTIVENESS AND EFFICIENCY OF COMPRESSION AND DECOMPRESSION OF IP PACKETS IN A WIRELESS LOCAL AREA NETWORK

(WLAN)

Bernard Sentala
(ENG2100073)

Supervisor: Dr Charles S Lubobya

A thesis/dissertation submitted to the University of Zambia in partial fulfilment of the academic requirements of the degree of Master of Engineering in ICT in the School of Engineering.

THE UNIVERSITY OF ZAMBIA, LUSAKA

2022

As the candidate’s supervisor, I have approved this research for submission

Sign .... Date ....March, 2023

Dr Charles S Lubobya

Lecturer (Supervisor)
University of Zambia (School of Engineering)

DECLARATION

I declare that this research is my own work. Where collaboration with other people has taken place or material generated by other researchers are included, the parties and/or materials are explicitly stated with references as appropriate.

This work is being submitted for the degree of Master of Engineering in ICT at the University of Zambia. It has not been submitted to any other university for any other Master's Degree or examination.

Sign .... Date ....March, 2023

Name: Bernard Sentala
(ENG2100073)

CERTIFICATE OF APPROVAL

This dissertation by Bernard Sentala is approved as fulfilling the partial requirements for the award of the degree in Master of Engineering in ICT at the University of Zambia.

---------------------------	----------------------	---------------------
Internal Examiner 1	**Signature**	**Date**
-----------------------------	----------------------	---------------------
Internal Examiner 2	**Signature**	**Date**
-----------------------------	----------------------	---------------------
Internal Examiner 3	**Signature**	**Date**
----------------------------	----------------------	---------------------
Chairperson **Board of Examiner**	**Signature**	**Date**

DEDICATION

I dedicate this work to all the scholars who had earlier worked in the field of IP Packet Compression and Decompression in a Wireless Local Area Network (WLAN). I further dedicate to my family members: My Wife and my children for the period I have been studying the degree of Master of Engineering in ICT.

ABSTRACT

In general, the study focuses on Evaluation of the Effectiveness and Efficiency of Compression and Decompression of IP Packets in a Wireless Local Area Network (WLAN). IP Packet Compression is the process of reducing the packet size before transmitting data in a Wireless Local Area Network (WLAN). WLAN has problems that include network delays, buffer requirements, bandwidth, and network congestion. The goal of the study was to design and develop a model that would help test the effectiveness and efficiency of Compression of IP Packets in a WLAN during transmission. The key specific objectives were; i) to assess the Effectiveness and Efficiency of the Per-Interface Compression during IP Packet transmission in a WLAN, ii) to analyze the Effects of TCP/IP Header Compression during IP Packet transmission in a WLAN, and iii) to evaluate the Effects of Per-Virtual Circuit Compression during IP Packet transmission in a WLAN. The OPNET simulation software tool was used to simulate the four (4) IP Packet Compression schemes namely: None-no compression, Per-Interface compression, TCP/IP Header compression, and Per-Virtual Circuit compression.

Following a series of OPNET simulation scenarios, the study revealed that among all the compression schemes, Per-Interface Compression had the lowest performance indicator for throughput (432,925 bits/sec), Data dropped (717,855,026.7 bits/sec) and Network Load (4,703,848.75 bits/sec). Whereas TCP/IP Header Compression had higher performance indicator for Traffic Sent (121,777,920 bytes/sec) and lowest performance indicator for Delay (0.622,575,955 seconds). This was determined using performance metrics for a particular situation, that include Throughput, Data Dropped, Delay, Network Load, Traffic Sent, and Media Access Delay. Per-Virtual Circuit Compression had performance indicators for throughput (1,873,536 bits/sec), Data dropped (534,815,948.2 bits/sec), Network Load (7102256 bits/sec), Traffic Sent (76,137,600 bytes/sec), Delay (4.171,825,058 seconds), and Media Access Delay (1.716,923,736 seconds). The scheme trailed behind Per-Interface and TCP/IP Header Compression schemes. The None-no compression scheme was merely utilized for comparison reasons. The Per-Interface Compression technique had the best performance and QoS in a WLAN.

Finally, the study's recommendations include: Users that set-up Per-Interface Compression should simulate or test how their network could function under different scenarios using their preferred settings and conditions. When the header size is bigger than the payload, users should use the TCP/IP Header Compression, and when the header size is smaller than the payload, users should use the Per-Virtual Circuit Compression scheme. The researcher believes that the findings will help other researchers and ICT professionals improve network performance and quality of service by varying compression schemes dependent on the kind of network platform.

Keywords: Data Compression, OPNET, WLAN, Effectiveness, Efficiency, performance and Quality of Service (QoS).

ACKNOWLEDGEMENT

I would like to thank the University of Zambia, particularly Dr Charles S. Lubobya, my supervisor, and Dr Ackim Zulu, my co-supervisor, for their assistance, patience, and guidance during the project. I applaud them for their dedication to duty, and I also thank numerous associates for providing me with research materials. My wife and children have been very supportive and understanding throughout the course. Thank you to all of the faculty members in the School of Engineering at the University of Zambia.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS AND ACRONYMS

ASCII	- American Standard Code for Information Interchange
BMP	- Bitmap Image File
Bps	- Bits per second
CD	- Compact Disc
CPU	- Central Processing Unit
DDC	- Delayed Dictionary-based Compression
DS	- Differentiated Services
DSCPs	- Differentiated Services Code Points
ECN	- Explicit Congestion Notification
EF	- Expedited Forwarding
Gbps	- Gigabits per second
GIG	- Global Information Grid
GUI	- Graphical User Interface
Hlen	- Header Length
ICT	- Information and Communication Technology
ICMP	- Internet Control Message Protocol
IGMP	- Internet Group Management Protocol
IP	- Internet Protocol
IPv4	- Internet Protocol Version 4
IPv6	- Internet Protocol Version 6
JIPM	- Joint IP Modem
JPEG	- Joint Photographic Experts Group
LTE	- Long Term Evolution
LZW	- Lempel-Ziv-Welch
MAC	- Media Access Control
MANET	- Mobile ad hoc network
MTU	- Maximum Transmission Unit
Mbps	- Megabits per second
MBMS	- Multimedia Broadcast Multicast Service
MP3	- Moving Picture Experts Group Layer-3 Audio
MPEG	- Moving Picture Experts Group

NS2	- Network Simulator Version 2
OMNeT++	- Objective Modular Network Testbed in C++
OPNET	- Optimized Network Engineering Tool (software)
OSPF	- Open Shortest Path First
PDCP	- Packet Data Convergence Protocol
PHB	- Per-Hop Behaviour
POMDP	- Partially Observable Markov Decision Process
RED	- Random Early Detection
RLE	- Run Length Encoding
RoHC	- Robust Header Compression
RTP	- Real Time Protocol
SATCOM	- Satellite Communication
TCP	- Transmission Control Protocol
TCP/IP	- Transmission Control Protocol/Internet Protocol
TOS	- Type of Service
TTL	- Time to live field
UDP	- User Datagram Protocol
QoS	- Quality of service
WLAN	- Wireless Local Area Network
WLSB	- Window Least Significant Bit

CHAPTER ONE

INTRODUCTION AND BACKGROUND

1.0 Introduction

In general, the study focused on IP Packet Compression, which is defined as the process of reducing the packet size while transmitting data in a Wireless Local Area Network (WLAN). Reducing the number of bits required to represent data is called data compression. Data compression can reduce network storage hardware costs, increase file transfer speed, simplify management, and save storage capacity [1] [2]. In WLAN that has high latency and high bit rates, it is very difficult to attain the much needed bandwidth. Network problems such as queuing, transmission, propagation and processing problems that include buffer requirement and IP Packet network congestion could be addressed when the available resources are used as efficiently and effectively as possible [1] [2]. The research primarily focused on the Evaluation of the Effectiveness and Efficiency of Compression and Decompression of IP Packets in a Wireless Local Area Network (WLAN) as the way to optimize and/or mitigate the network delay problems associated with IP Packet delivery. The study used Optimized Network Engineering Tool (OPNET) software for the simulation, modeling, and analysis of computer network to simulate and come up with better solutions to the key research questions. During simulation, the performance metrics used were throughput, Data dropped, Delay, Network Load, Traffic sent and Media Access Delay that was measured from source to destination end nodes in a WLAN platform [1] [2] [3] [4].

The goal of the study was to Evaluate the Effectiveness and Efficiency of the IP Packet Compression and Decompression in a WLAN with the view of optimizing and or mitigating the delays, buffer requirement and congestion challenges/problems using OPNET network simulator tool. The overall objectives of the study were to design and develop a model that would help test the effectiveness and efficiency of Compression of IP Packets in a WLAN during transmission in a Wireless Local Area Network (WLAN). Specific objectives included the following: i) to assess the Effectiveness and Efficiency of the Per-Interface Compression in a WLAN during IP Packet transmission, ii) to analyze the Effects of TCP/IP Header Compression in a WLAN during IP Packet transmission, iii) to evaluate the Effects of Per-Virtual Circuit Compression in a WLAN during IP Packet transmission.

1.1. **Background**

Compression is the way of making files take up less space. Reducing the size of a frame reduces the time required to transmit the frame across the network. Data compression technique provides a coding scheme at each end of a transmission link that allows characters to be removed from the frames of data at the sending side of the link and then replace them correctly at the receiving side [5] [6]. The IP Packet Compression is the process that reduces the packet size or file size during transmission [2] [7] [5]. The study considered three (3) main kinds of IP Packet Compression techniques namely: stateless, streaming, and offline compressions. In all the three (3) forms of IP Packet Compression technique, if a compressed IP Packet length is greater than or equal to the original one, the original IP Packet is transmitted [8]. The payback of IP Packet Compression can be observed and exploited to optimize, enhance and evaluate the Effectiveness and Efficiency of the network performance and Quality of service in a Wireless Local Area Network (WLAN) [2] [5] [6] [7] [8].

The study used four (4) IP Packet Compression schemes as scenarios supported by Optimized Network Engineering Tool (OPNET) software for simulation, modeling, and analysis of computer network: i) None-no compression, where there is no compression at all, ii) TCP/IP Header Compression, where only TCP/IP header is compressed, iii) Per-Interface Compression where the entire packet is compressed and iv) Per-Virtual Circuit Compression where only the IP Packet's payload (i.e., without the header information) are compressed and that the compression takes place only at the end nodes [9] [10] [11].

Compression uses various mechanism to compress IP Packet before the packet is transmitted [9] [10]. Compression can be classified into two (2) broad classes as lossy or lossless. Lossy compression is when the data is decompressed to reduce a file size by permanently eliminating redundant information, so that the picture and/or image quality for example is reduced to minimal degradation of its quality. Lossless compression is when the data is decompressed and the result is a bit for bit perfect match with the original data or no data is lost at all [8] [9]. Lossless compression scheme, is one in which X is identical to Y, and lossy compression scheme is one in which the scheme generally provide much higher compression than lossless compression but allow Y to be different from X [8] [9] [10].

Compression is the process of encoding information to a representation using fewer bits than the original representation. Data that is compressed can be of characters in a text-file or numbers that represent digital audio, images and video among many others. There are many techniques (algorithm) used for IP Packet compression, for the purpose of this study, stateless, streaming and offline compression techniques is discussed [7] [8] [9]:

Stateless Compression (Packet by Packet Compression): This approach is when IP Packets are compressed and decompressed independently with absolutely no relationship to any other IP Packet and its purpose is to reduce header overhead of a packet [8] [12] [13]. This approach of compression and decompression is beneficial for IP Packets that arrive out of order at the receiving end. Since each IP Packet is independent, it can always be decompressed by the receiver regardless of the IP Packet order of arrival or IP Packet drops [8] [12] [13].

Streaming Compression (Continuous Compression): Using streaming compression, IP Packets are compressed and decompressed with a degree of interdependence. This is attractive when reliable means of communication are available. Each IP Packet is compressed using the history of the previously compressed IP Packets as well as the data within the current IP Packet. When the reliability of a link is poor, streaming compression is unattractive [8] [12] [14].

Offline Compression (Compress and Send): In this techniques, data is compressed as a complete unit, in blocks, rather than in small chunks. The IP Packets are not compressed by the network processor, since it is not worthwhile to compress compressed data. Since compression is performed at the application layer and not inside the network processors, offline compression does not require any buffer for decompression [8] [12].

1.2. **Problem Statement**

The study was initiated during COVID-19 pandemic outbreak, when most countries advocated for a quarantine period ranging from social distancing to self-isolation at home to prevent the virus from spreading. The General Public turned to online activities to gain access to much needed services that would otherwise be inaccessible. Hence, the online activities resulted in increased global internet traffic with the majority of multimedia information such as video streaming and video conferencing being transmitted including electronic learning, meetings and other related electronic means by institutions such as corporate companies, government agencies, universities, and households. The global community quickly adapted by creating online content and the internet bandwidth consumption for both downstream and upstream usage was required to cope with increased demand. The internet traffic caused a number of network challenges, such as network delays (queuing delay, transmission delay, propagation delay and processing problems), network congestion, bandwidth, and buffer requirements, that could be measured by performance metrics such as throughput, data dropped, traffic sent, network load, and media access delay among others.

IP Packet Compression minimizes the amount of network overhead and payload size to increase the transmission speed. Hence, some of the benefits include; increased storage capacity, faster transmission speed, fewer transmission errors, and adding a layer of security to the WLAN. The advantages outlined above may not remain constant due to IP Packet challenges, such as network delays, network congestion, buffer, and bandwidth requirements which may emerge as internet traffic grows. However, despite these challenges, preliminary literature revealed that not much had been done to address the identified gaps in a WLAN.

In order to address these problems, the four compression schemes; None-no compression, Per-Interface compression, TCP/IP Header compression, and Per-Virtual Circuit compression were to be simulated in a WLAN. All the above-mentioned challenges are affected by IP Packet compression, and for that reason, the study was conducted to determine if there was a better way of delivering IP Packets for video conferencing in a WLAN through variation of the compression schemes [1] [2] [3].

It is therefore, against this backdrop that the researcher seeks to use Evaluation of the Effectiveness and Efficiency of Compression and Decompression of IP Packets in a Wireless Local Area Network (WLAN) during IP Packet delivery. The primary focus of the study is IP Packet Compression and Decompression during transmission that was validated through the results of a realistic WLAN using Optimized Network Engineering Tool (OPNET) software through sending a video in a full function network [12]. The research topic that guided the study was Evaluation of the Effectiveness and Efficiency of Compression and Decompression of IP Packets in a Wireless Local Area Network (WLAN).

1.3. **Aim of the Study**

In general, the aim of the study is to evaluate the effectiveness and efficient of compression and decompression of IP Packets in a WLAN by using the four (4) compression schemes considering performance and Quality of Service so as to come up with optimized and/or mitigated IP Packet delivery problems.

1.4. **Objectives of the Study**

The main objective of the study was to design and develop a model that would help test the effectiveness and efficiency of Compression of IP Packets in a WLAN during transmission. The following are the Specific objectives:

i) To assess the Effectiveness and Efficiency of the Per-Interface Compression during IP Packet transmission in a WLAN.

ii) To analyze the Effects of TCP/IP Header Compression during IP Packet transmission in a WLAN.

iii) To evaluate the Effects of Per-Virtual Circuit Compression during IP Packet transmission in a WLAN.

1.5. **Research Question(s)**

The study attempted to answer key evaluation questions mirroring the objectives outlined above:

i) How effective and Efficient is Per-Interface Compression during IP Packet transmission in a WLAN?

ii) How does the TCP/IP Header Compression affects performance and or Quality of service (QoS) during IP Packet transmission in a WLAN?

iii) How does the Per-Virtual Circuit Compression affects performance and or Quality of service (QoS) during IP Packet transmission in a WLAN?

1.6. **Conceptual Framework**

A conceptual framework refers to when a researcher conceptualize the relationship between variables in the study and shows the relationship graphically or diagrammatically. It is a hypothesized model identifying the concepts under study and their relationship [15]. It is a system of concepts, assumptions, expectations, beliefs and theories that support and informs research [16].

The research therefore, was conducted by a conceptual framework representing the potential weaknesses to possible better optimize and/or mitigate (maximize) or design and develop an improved data packet delivery (dependent variable) as being inadequacy in type of algorithm to be used and data format used, data compression scheme used, data type used and data dimension (independent variables). Figure 1.1: Conceptual Framework displays the study's Conceptual Framework for the study.

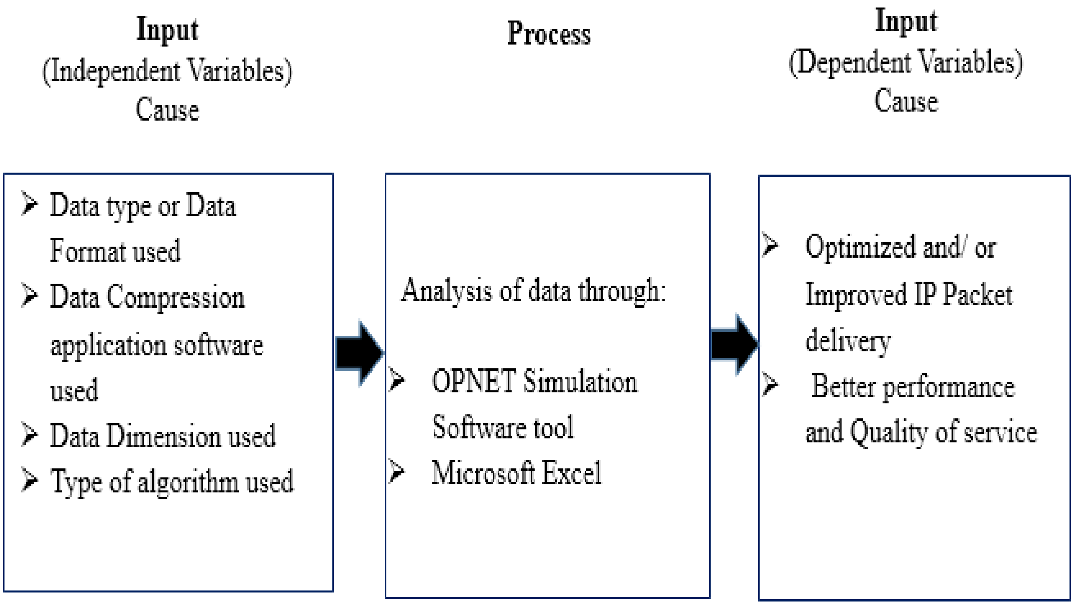

Figure 1. 1: Conceptual Framework
Source: Researchers design March (2023)

From Figure 1.1: Conceptual Framework, the dependent variable is the effect, and its value depends on changes in the independent variable. The independent variable is the cause. Its value is independent of other variables in a study and subject to change as a result of the researcher's actions. Therefore, it is clearly seen that by varying the independent variables, we can achieve dependent variable at optimum level (i.e. maximized or improved level of delivery) in form of IP Packet delivered from source to destination points. In short, the independent variable (cause) is the input or stimulus variable which is manipulated to cause a change in other variables such as dependent variable. Whereas, the dependent variable (effect) is the output or response variable which is observed aspect of behaviour of the entity that has been stimulated [17].

1.7. **Significance of the Study**

IP Packet Compression plays a very important role in Science and Technology because it dramatically decreases the amount of storage a file takes up. The significance of the study to the community/society, researchers, learning institutions, industries, and many other organizations are: i) Storage efficiency, this is the capability to store and manage data in the least amount of space, ii) Efficient usage of transmission bandwidth, this is the transmission of maximum data, iii) Faster transmission time, the uploading and downloading files would take less time, iv) Energy conservation, the electricity and battery-powered devices could last longer, v) Apprises policymakers, in the ICT industry and institutions working with online transactions such as Data science, Cloud computing, Artificial intelligence, Blockchain and Cybersecurity among others [2] [3].

A network simulator fulfills many needs. When compared to the cost and time required to set up an entire WLAN with many network computers, cables and routers, emulators are relatively fast and affordable. They allow researchers to test situations that might be extremely difficult or expensive to simulate using real hardware, such as recreating multi-node scenarios or testing new network protocols [1] [3].

Other reasons for undertaking a simulation are: i) If a mathematical approach cannot be used to solve the problem, simulation is the most suitable for investigating complex and large-scale practical situations, ii) Since the simulation is adaptive, adjustments can be made to the system variables to find the optimal option from a range of options, iii) By understanding the options well in advance and reducing the risk of experimentation in the real system, policy decisions can be made much faster and iv) In simulation, experiments are performed with the model without disturbing the system [1] [2].

The justification for undertaking this study include its findings that are based on both theoretical relevance to other researchers who can use the information as secondary source of data. As a final point, the successful implementation of this research assists the researcher meet the academic obligation as it is conducted in partial fulfilment of the academic requirements for the Master's degree of Engineering in ICT [18].

1.8. **Motivation of the Research**

A study was being carried out to examine if there was a better way of delivering IP Packets for multimedia information such as video on a network such as WLAN. A lot of multimedia information is being transmitted over the internet especially by higher learning institutions such as Universities, Government and Private institutions through virtual transactions, electronic learning, meetings and other related electronic matters. The IP Packet delivery improvement ways that this study came up with enhances the storage space and transmission of IP Packet delivery. The research is expected to contribute immensely to the existing body of knowledge in the technological growth of the IP Packet Compression in a Wireless Local Area Network (WLAN) field.

1.9. **Application of the Research**

The IP Packet is a term first coined by Donald Davies in 1965 [19]. It was used to describe a segment of data sent from one computer or device to the other. The study provides good understanding concepts on IP Packet Compression to make critical applications perform better/faster, it also saves Wide Area Network (WAN) costs and eases buffer requirements, delays and congestion in WLAN links.

There are various application areas of IP Packet and data compression in general that include: i) Audio compression, is one that reduces the transmission bandwidth and storage requirements of audio data, ii) Video compression uses modern coding techniques to reduce redundancy in video data and iii) Emulation, is one that emulate Compact Disc (CD) based consoles such as the play station two (2). Data compression is desirable to reduce huge amounts of disk space used. Compression and Decompression techniques are utilized for a number of applications, such as computer systems for example memory structures, disks and tapes; computer networks especially internet, mobile computing, facsimile system, voice mail, telephone, printer systems, document storage and retrieval systems, video conferencing systems, electronic multimedia messaging systems, imaging and image archival, signal processing, digital and satellite TV [20].

Other applications of IP Packet Compression are that it can reduce the size of images/video so that the data can be stored or be transmitted easily. In the case of a summary, say reducing a book to a paragraph that is possibly designed to help entice people to buy the book without having to read the whole book to know what it is about.

1.10. **Scope of the study**

The study evaluated the performance of the IP Packet Compression and Decompression for transmission in a Wireless Local Area Network (WLAN). The study is limited to validate IP Packet delivery performance and or Quality of service (QoS) through measured performance metrics such as throughput, data dropped, network load, traffic sent, delay and media access delay during transmission using OPNET software simulation tool.

1.11. **Operational Definition**

Bandwidth: This term was used to mean the maximum amount of data to be transmitted over an internet connection in a given amount of time. Bandwidth is the maximum data transmission rate possible on a network. Network bandwidth is a measure of the capacity of the network to transmit video surveillance information in bits per second [21]. Note that bandwidth is not internet speed but the volume of information/data that can be transmitted through a given communications circuit per second or over a connection in a measured amount of time calculated for example in bits per second (Bps) or megabits per second (Mbps). In a wired network we measure it in bits per sec (bps) while in a wireless network we measure it in hertz. [22].

Data Compression: This term was used in this study as the process of encoding, restructuring or otherwise modifying data in order to reduce its size and change attributes of the file that involves re-encoding of information using fewer bits than the original representation for data storage and transmission [23] [24].

Effectiveness: The term has been used as the capability or power to produce/accomplish the desired result or output. It is about how much *time* you take to get to your objectives. Doing the right thing [25].

Efficiency: This is the ability to do something or produce something without wasting materials or resources. Efficiency is about performing a task in a best possible manner. It is about how well you use your resources. Doing the things right [25].

Latency: The term was used to refer to a node or device that experiences delay when there is a delay between when data is requested and when it is received. Regardless of the source, your network performance monitoring can track and log all delays. This delay can occur for many different reasons. You need a monitoring tool to track latencies as they are often invisible to the human eye. Persistent lag or strange spikes in lag duration indicate a serious performance problem.

Lossless: The term describes data compression algorithm which retains all the information in the data, allowing it to be recovered perfectly by decompression. As the name suggests, lossless compression techniques do not lead to any loss of data or information and that the compression is done by representing the file using less number of bits, without any loss of information [8] [24] [26].

Lossy: The term was used to mean the lost data and quality from the original version and that this term is typically associated with files such as Joint Photographic Experts Group (JPEG), Moving Picture Experts Group Layer-3 Audio (MP3) and Moving Picture Experts Group (MPEG) among many others. Lossy compression is a methodology for achieving higher compression ratios at the cost of losing some information about the represented object [8] [24] [27].

Multimedia: This term was used as a form of communication that combines different content such as text, audio, images, animation, or video into a single interface [20].

Offline compression (Compress and Send): The term is used to first compress the data offline, at the application layer, then divide the compressed data into packets, and finally send the packets to the receiver [8] [12].

Performance: This term was used as a Network performance that refers to measures of quality of service of a network as seen by the researcher. There are many different ways to measure the performance of a network, as each network is different in nature and design [22].

Quality of Service: This term was used as the capability of the WLAN to provide the best service to the clients with high efficiency and effectiveness of the traffic flow of voice, video and data over the network system with a good quality according to the priority of what is required to speed up the network and avoid data delay [28].

Stateless Compression: In general, stateless describes any process that does not have a memory of past events, while stateful describes processes that do have such a memory (and use it to make decisions). Stateless means whatever chunk of data it sees, it compresses, without depending on previous inputs [8] [12].

Streaming compression (Continuous Compression): In this approach, the history buffer is not initialized after every packet is encoded. Each packet is encoded by using a history which is derived from all preceding packets and from the current packet. In this method packets are encoded in their consecutive order [8] [12].

Throughput: The term was used in data transmission as network throughput to mean the amount of data moved or transmitted successfully from one end node or place to another in a given time period measured in bits per second (bps), or in megabits per second (Mbps) or gigabits per second (Gbps). Throughput measures your network's actual data transmission rate, which can vary wildly through different areas of your network [21] [29] [30].

Transmission: The term was used to mean data transmission rate as volume of data transmitted over a transmission channel or via a data interface within a specified unit of time (measured in bits/s) [22].

1.12. **Ethical Considerations**

The researcher considered and ensured that the study was undertaken by following the University of Zambia guidelines stipulated in the research handbook for graduate students preparing to submit thesis/dissertation for examination. The study reflected on legal, social, anonymity, plagiarism and fraud, confidentiality and privacy, professional and academic freedom issues before, during and after the study is completed. The researcher also endeavoured to follow code of conduct that conveys moral integrity and consistent values.

1.13. **Plan of Development**

The presentation of the study was structured to give a flow from Introduction, Literature Review, Research Methodology, Results and Discussion, Conclusion and Recommendations. The detailed outline for the report presentation is as follows:

Chapter one: This is the first section of the study. The chapter introduces the general issues around the IP Packet Compression and Decompression. The purpose of this chapter is to provide an overview of the research, its background information, statement of the problem with its objectives, significance of the subject and the conceptual framework underpinning the study. The section was used to direct the path for the remaining chapters.

Chapter two: This is mainly concerned with literature review on global trends in IP Packet Compression and Decompression. It highlights trends in the storage and transmission of multimedia information. It also details the theoretical framework and empirical review (related work).

Chapter three: This section describes the research methodology to be used in the study. All the research procedures for gathering data, variables involved, an explanation of how data was analyzed and any other information that may be required for the study was explained step-by-step process that the researcher selected in this section.

Chapter four: This section forms the core of the research that outlines the presentation of lab Implementation, Data collection/Viewing and analysis of the results from OPNET simulation and Microsoft Excel Spread Sheet analysis. This has been done through the use of tables and graphical representation.

Chapter five: This section gives the research conclusion and recommendations based on the simulations, experimental results and discussions obtained. As a final point, some potential future work was also proposed.

1.14. **Summary**

The main focus of the chapter was IP Packet Compression and Decompression during network transmission. The study has highlighted how this could be one way of optimizing and mitigating the packet network delay and other problems associated with IP Packet transmission such as queuing delay, transmission delay, propagation delay and processing problems including, buffer requirement, IP Packet network congestion and latency.

CHAPTER TWO

LITERATURE REVIEW

2.0 Introduction

The chapter reviewed theories and past related literature (works) on IP Packet Compression and Decompression used globally, and narrowing it down to Wireless Local Area network (WLAN). The review sources included textbooks, conference papers, government documents, reports, journal articles and many other selected electronic sources from the internet. Therefore, the literature review pertaining to the study was categorized into two (2) broad classification. The first section discussed the general aspects of IP Packet Compression capturing the theoretical foundation that explains phenomena. The second part was empirical review (i.e. past related work) on the IP Packet Compression and Decompression. The study focused primarily on the IP Packet Compression and Decompression schemes, performance and/or Quality of service through measured performance metrics such as throughput, data dropped, delay, network load, traffic sent, and media access delay during transmission over the past decade in the context of a Wireless Local Area Network (WLAN).

2.1. Information and Rate-distortion Theories

This sub-section gave a brief explanation of two (2) theories that supports the research topic:

2.1.1 Information theory

Information theory is the mathematical treatment of the concepts, parameters and rules governing the transmission of messages through communication network systems. The theory was founded by Claude Elwood Shannon, who pulled everything together into what is now called information theory [31] [32]. Claude Elwood Shannon was an electrical engineer at Bell Labs toward the middle of the twentieth century, the lab has since then evolved into a vigorous branch of mathematics fostering the development of other scientific fields, such as biology, behavioral science, neuroscience, and statistical mechanics [31] [32]. The concept of transmission which set up many important precedents for digital technology, including the usage of bits as units of measurement of the compression processes [31] [33] [32] [34].

Information theory is a branch of arithmetic that defines efficient and practical methods by which data can be exchanged and interpreted [33][32]. Prior to information theory, electronic communication was conducted mostly through analog transmission, which worked well enough in short distances but became problematic as the distances increased and signals were getting degraded. The concept of information theory was invented to solve communication problems [33] [32] [34].

The field was fundamentally established by the works of Harry Nyquist and Ralph Hartley, in the 1920s, and later Cloud Shannon in the 1940s for Efficient usage of transmission bandwidth. Morse code is an early example of data compression based on using shorter code words [31]. Modern work on data compression began in the late 1940s with the development of information theory [33]. The foundation of information theory was finally laid in 1948 by Shannon titled, “A Mathematical Theory of Communication.” Shannon was interested in how much information a given communication channel could transmit. In the 1970s, software compression programs began to be developed, almost all based on adaptive Huffman coding. Typical compression ratios currently achieved for text are known to be around 3:1, for line diagrams and text images around 3:1, and for photographic images around 2:1 for lossless, and 20:1 for lossy [33] [32] [31] [34].

The number of bytes of data, stored, processed, and transmitted keeps on growing. Data compression is one of the enabling technologies for each of the multimedia revolutions [32]. It would not be practical to put text, images, let alone audio and video, on websites if it were not for data compression algorithms. Listening to music or watch a movie on the internet, you are being entertained through the aid of data compression [33] [32] [34].

Data compression is the art or science of representing information in a compact form. We create compact representations by identifying and using structures that exist in the data. Data can be characters of a text file, numbers that are samples of speech or image waveforms, or sequences of numbers that are generated by other processes. To represent for example two (2) minutes of Compact disc quality music requires more bits [33] [32] [34].

Cloude Elwood Shannon formulated the theory of data compression by establishing that there is a fundamental limit of lossless data compression. This limit he called it the entropy rate and is denoted by H. In short information theory is based on a measure of uncertainty known as entropy (which is designated as (H). For example, the entropy of the stimulus S is written as H(S). The precise/exact value of H depends on the information source, more specifically, the statistical nature of the source. It is possible to compress the source, in a lossless manner, with compression rate close to H. It is mathematically impossible to do better than H [33] [32] [34].

2.1.2 **Rate Distortion theory**

Rate-distortion theory was introduced to the works written in 1948 and 1959 by Cloude Elwood Shannon, the founder of Information Theory. The Rate-distortion theory is the branch of information theory that treats compressed data produced by an information source down to a specified encoding rate that is strictly less than the source's entropy. Rate distortion theory is concerned with the trade-offs between distortions and rate in lossy compression schemes. Rate is defined as the average number of bits used to represent each sample value [32]. Entropy is a numerical measure of the uncertainty of an outcome. This entails that there is some distortion between the original source data and the best approximation that can be produced on the basis of the encoder's output bits. [31] [35].

In lossy data compression, the decompressed data does not have to be exactly the same as the original data. Instead, some amount of distortion, D, is tolerated. Shannon showed that, for a given source (with all its statistical properties known) and a given distortion measure, there is a function, R (D), called the rate-distortion function. The theory states that if D is the tolerable amount of distortion, then R(D) is the best possible compression rate. When the compression is lossless (i.e., no distortion or D = 0), the best possible compression rate is R (0) = H (for a finite alphabet source). In other words, the best possible lossless compression rate is the entropy rate. In this sense, rate-distortion theory is a generalization of lossless data compression theory, where we move from no distortion (D = 0) to some distortion (D > 0) [33] [32] [31].
Lossless data compression theory and rate-distortion theory are known collectively as source coding theory. Source coding theory sets fundamental limits of the performance of all data compression algorithms. The theory, in itself, does not specify

exactly how to design and implement these algorithms. It does, however, provide some hints and guidelines on how to achieve optimal performance.

2.2. Empirical Review

The explosive growth of data in digital world has led to the requirement of efficient techniques to compress data. In general, data compression first appeared in the early 19^{th} Century. Data compression concepts result to effective utilization of available storage area and bandwidth thereby improving the performance and throughput during packet delivery. Compression of data is the process of reducing the size of data into a smaller but yet a compact form. Compression is the procedure of converting data set into a code to save the need for storage and transmission of data [32] [36] [37].

2.2.1 Data compression

The process that involves transforming of information from one representation to another smaller representation from which the original, or a close approximation to it, can be recovered, is termed as data compression. Data compression involves identifying models for the many different types of structures that exist in different kinds of data. These can be in the form of patterns that we can recognize simply by plotting the data, or statistical structures that require a more mathematical approach [32]. Data compression is required because uncompressed data take up a lot of space which reduces the Effectiveness and Efficiency of CPU due to limited availability of hard drive space. Compression has other advantages such as reducing the resource usage and cost. [37] [38].

Compression algorithms are mainly divided into lossless and lossy compression. Lossless data compression represents the data in a compact form without data loss e.g. text data. Lossy data compression allows for the loss of data during the process of compression. When we compress for example audio data some tones are not audible to a human ear because our senses are imperfect. Many compression algorithm techniques can be performed such as the Huffman, Lempel Ziv Welch and Run Length Encoding [32] [37] [39] [38].

Figure 2.1: the data process of data compression.

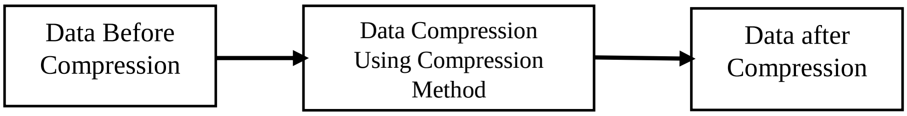

Figure 2. 1: The Data Process of data compression
Source: [38]

2.2.2 **Packet compression Techniques**

The recent emergence and popularity of wireless internet have triggered a demand for improved transmission efficiency. Packet compression at the link layer may speed up the delivery process and provide more efficient use of available capacity. The packet compression can be targeted to cover only the header or payload part, or both parts of the packet. There are several techniques for compressing IP Packets that are indicated in Figure 2.2: Compression techniques [40].

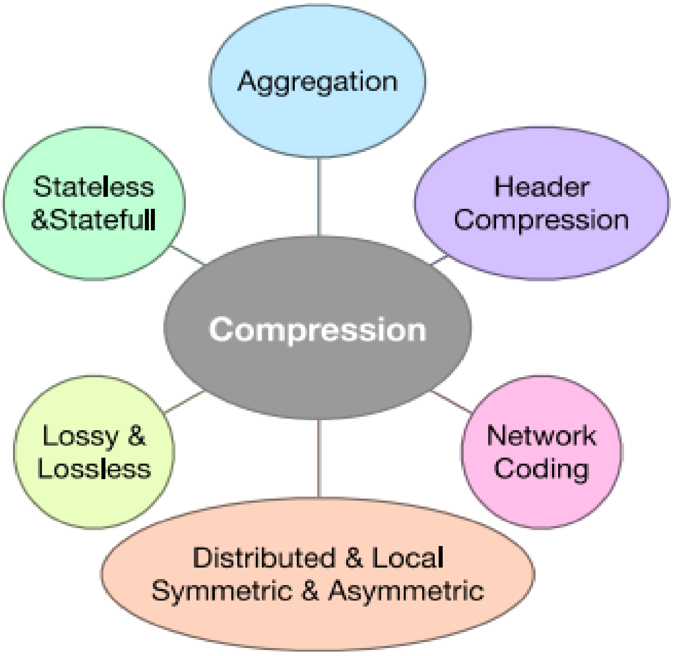

Figure 2. 2: Compression techniques
Source: [40].

The study however mainly focused on three (3) compression techniques namely; Stateless and stateful, Lossless and Lossy, and Header compression in relation to V.44 tabulated in International Telecommunication Union (ITU-T) the International Standards. V.44 is an ITU standard for analogue modem data compression that was published in 2000. V.44, which is optimized for web traffic, has a compression ratio of up to 6:1 versus 4:1 in the V.42bis standard [4] [7].

V.44 is an ITU-T standard for compression of data sent by modem data compression standard that provides much higher data rates than V.42bis, which was previously the most widely used modem compression protocol. More data can be transmitted simultaneously with a higher compression ratio due to the fact that HTML files are highly compressible, V.44 offers particularly dramatic improvements in Web browsing. 20% to 60% increase in data throughput compared to V.42bis for most users [4] [7]. Hughes Network System, the world's leading provider of broadband satellite services, developed the compression algorithms used in V.44 for these services. Ideal for packet-switched networks like the internet, it was introduced in late 1999 as a replacement for V.42bis [4] [7].

2.2.3 **Stateless and Stateful**

Compression methods used with Transport Layer Security (TLS) can either be stateless (process that does not have a memory of past events and is used for unreliable link) or stateful (process that does have a memory and uses it to maintains its state over all compressed records and uses reliable link) compression techniques. There are several compression techniques that preserve historical data when compressing and decompressing the data portion of a packet. Compression history allows better compression ratios for streams than packet compression, but preserving history across packets means that a packet can contain the data needed to fully decompress data stored in another packet. Therefore, maintaining history requires both a reliable network and ordered packet delivery. Compression history information can be preserved and used if the compression technology supports it, as TLS and lower layers ensure reliable and ordered packet delivery [4] [7] [40].

The stateless compression approach is when packets are compressed and decompressed independently, with absolutely no relationship to any other packet. The approach of compression and decompression is beneficial when packets arrive out of order at the receiving end. Stateless means whatever chunks of data it sees, it compresses, without depending on previous inputs [4] [7] [40].

The term **stateful** describes the processes that do not yet have access to memory, but do use it to make decisions. In the case of stateful compression (or shared context), there would always be negotiation between the sender and the receiver. During the negotiation, both parties agree on the semantics how the compression would be performed [4] [7] [40].

The stateful approach would allow much higher compression rate than the stateless, because there is no need to assume anything before the negotiation so the packet formats can change freely. Because of the nature of wireless communication, there would always be some packet loss and bit errors, but if the losses and errors are very small, then stateless approach would outperform stateful approach [4] [7] [40].

2.2.4 **Lossy and Lossless**

Compressing data helps in enhancing storing, managing, and transferring data becomes essential in data communication and other data-driven solutions. This is because no matter the degree of advancement in computer hardware (RAM, ROM, CPPU) and forms of communication (internet), these resources are scarce. To utilize these resources efficiently, the data is often required to be compressed, i.e., reduced to a smaller size without losing any or losing minimal information. Data Compression methods can be divided into lossy and lossless techniques. Lossless compression saves bits by detecting and eliminating statistical redundancies. With lossless compression, no information is lost. Lossy compression reduces bits by removing unnecessary or less important information.

Lossless techniques: Lossless technique aims to return exactly the original data. This technique can be applied on data such as text or programs though even some kinds of image and video information could also be compressed. There are different methods or technique for lossless compression which include:

Run Length Encoding (RLE): Is a simple data compression algorithm supported by bitmap file formats such as bitmap image file (BMP). RLE basically compresses the data by minimizing the physical size of a repeating string of characters. It can also be used in combination with other compression techniques [9] [10].

Lempel-Ziv-Welch (LZW): The LZW is a general compression algorithm capable of working on almost any type of data. It replaces the actual data with references onto the table of strings commonly occurring in the data being compressed. The table is formed during compression at the same time as the data is encoded and decoded [9] [10].

Huffman Coding: The Huffman coding deals with data compression of American Standard Code for Information Interchange (ASCII) characters. It is used in compression of many types of data such as text, audio, video and image. This method is based on building a full binary tree for the different symbols that are in the original file [9] [4].

Lossy Techniques: Lossy techniques differ from lossless compression, as its name implies, some amount of data may be lost in the process. Lossy techniques are generally used for the compression of data that originate as analog signals, such as speech, video, pictures and music files that can be trimmed at the edges. Unlike text files and processing files, pictures do not require reconstruction to be identical to the original, especially if the data dropped is insignificant or undetectable. In video, image and audio/voice applications, lossy compression could be applied. The Common lossy compression-based image formats are Joint Photographic Experts Group (JPEG), Moving Picture Experts Group Layer-3 Audio (MP3) and Moving Picture Experts Group (MPEG) and the various techniques include; Chroma subsampling, Colour reduction, Fractal compression, Transform coding, Vector quantization and many others [40] [41] [42].

2.2.5 Header Compression

The network packet is a packet that contains the data and header information coming from different protocols. The header part is interesting because there is redundant information about different protocol headers and, especially, between consecutive packets belonging to the same flow. This kind of redundant information can be elided, merged or deleted. Over a single link, not all that information is needed and part of it can be removed temporarily [43]. The diagrammatic IPv4 packet format which is used in the IP layer of the TCP/IP suit is described in the following paragraphs.

2.2.6 **IPv4 Packet Format**

IPv4 is Internet Protocol Version 4, which is used in the IP layer of the TCP/IP suite to identify each device connected to the internet. IP datagrams are a key part of the IP Packet service model. The IP datagram, like most packets, consist of a header followed by a number of bytes of data. This section discussed the IPv4 packet formats under which OPNET simulation tool operated on as shown in Figure 2.3: IPv4 Packet Format [44] [45] [46]

The Internet Protocol version 4 (IPv4), is used in the IP layer of TCP/IP protocol suite to identify each device connected to the internet. IPV4 address is unique and universal, 32 bit long having address space of 2^{32} or 4,294,967,296 unique addresses [47]. An IPv4 packet header consists of 14 fields in which 13 fields are required; among these fields, only one is optional, which is aptly known as the options component [44] [31] [46]. Figure 2.3: is IPv4 Packet format.

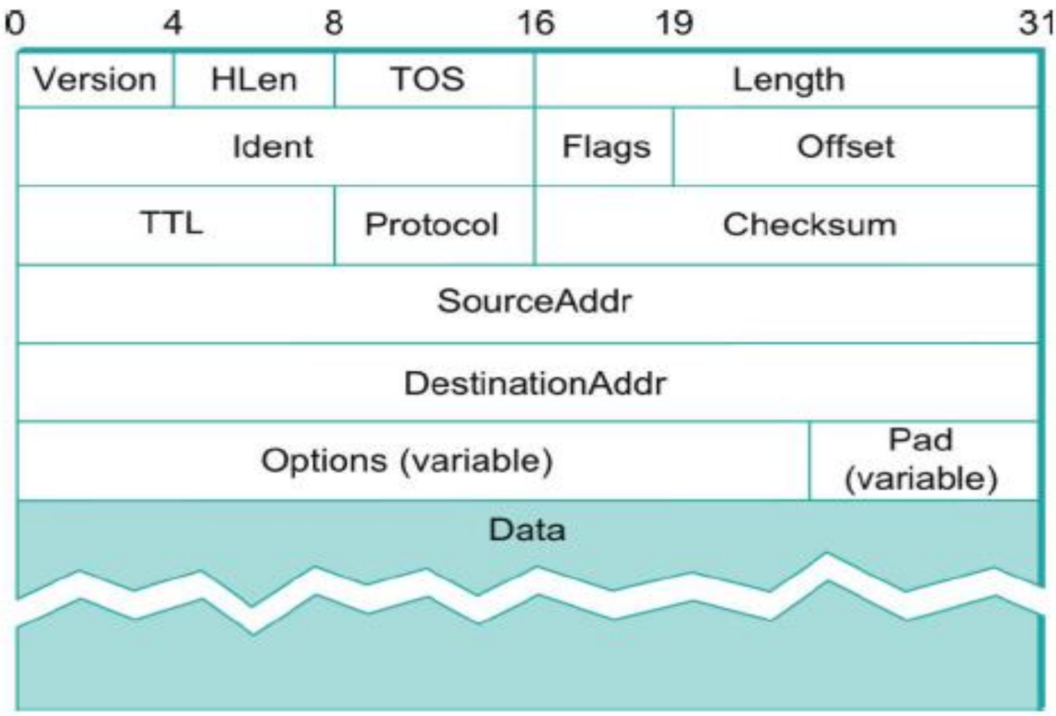

Figure 2. 3: IPv4 packet format
Source: [44] [45] [46]

Considering each field in the IPv4 packet header, we can identify the following components/features, their sizes and what they can do during packet transmission or delivery:

Version (Ver): The 4-bits version field that specifies the version of IP Packet header, for example IPv4 or IPv6 [44] [45] [46].

Header Length (Hlen): The 4-bits Hlen, specifies the length of the header in 32-bit words. When there are no options, which is most of the time the case, the header is 5 words (20 bytes) long [44] [45] [46].

Type of Service (TOS) or Differentiated Services: The 8-bits Type of Service (TOS) field is used to allow packets to be treated differently based on application needs. For example, the TOS might determine whether or not a packet should be placed in a special queue that receives low delay, reliability or high throughput. The first 6-bits of this byte have been allocated for Differentiated Services Code Points (DSCPs), where each DSCP is a 6-bit value that identifies a particular per-hop behaviour (PHB) to be applied to a packet. The remaining 2 bits are used by Explicit Congestion Notification (ECN). The per-hop behaviour (PHBs), a term that indicates that they define the behaviour of individual routers rather than end-to-end services. Because there is more than one new behaviour, there is also a need for more than 1 bit in the packet header to tell the routers which behaviour to apply [44] [31] [46].

The simplest explanation for PHBs is known as expedited forwarding (EF). Packets marked for EF treatment should be forwarded by the router with minimal delay and loss. The only way that a router can guarantee this to all EF packets is if the arrival rate of EF packets at the router is strictly limited to be less than the rate at which the router can forward EF packets. For example, a router with a 100-Mbps interface needs to be sure that the arrival rate of EF packets destined for that interface never exceeds 100 Mbps. It might also want to be sure that the rate would be somewhat below 100 Mbps so that it occasionally has time to send other packets such as routing updates [44] [31] [46].

Explicit Congestion Notification (ECN): This a technique by which routers inform end hosts about congestion by setting a flag in packets they are forwarding. Used in conjunction with active queue management algorithms like Random Early Detection (RED) [44] [31] [46].

Length: This is the 16 bits of the header that contain the Length of the datagram, including the header. Unlike the HLen field, the Length field counts bytes rather than words. Thus, the maximum size of an IP datagram is 65,535 bytes (2^{16} - 1). The physical network over which IP is running, however, may not support such long packets. For this reason, IP supports a “fragmentation and reassembly” process [44] [31] [46].

The central idea here is that every network type has a Maximum Transmission Unit (MTU), which is the largest IP datagram that it can carry in a frame. Note that this value is smaller than the largest packet size on that network because the IP datagram needs to fit in the payload of the link-layer frame. The second word of the header contains information about fragmentation (Identification, Flags and Offset) [44] [31] [46]:

Identification: This is the 16-bits field in the IPv4 header which indicates an identifying value assigned by the sender to aid in assembling the fragments of an IPv4 datagram and that all fragments carry same identification [44] [31] [46].

Flags: Flag in an IPv4 header is a three-bit field that is used to control and identify fragments. The following can be their possible configuration: Bit 0: this is reserved and has to be set to zero, in other words it is not used, ii) Bit 1: Don’t Fragment (DF) or do not fragment (If DF is set to 0, it can be fragmented and if set to 1, it should not be fragmented), iii) Bit 2: More fragments (MF). If MF is set to 0 it indicates that it is not the last fragment, and if set to 1 it means it is the last or the only segment [44] [31] [46].

Fragment Offset: This field is 13 bits long in length, and it is measured by blocks that units of 8-byte blocks. These are used to specify the offset of a fragment relative to the start of the IP datagram, which when it was not fragmented. As you can expect, the first offset of a fragment is always set to zero. The maximum possible offset is (2^{13}- 1) * 8 = 65528, but it is more than the maximum possible IP Packet length, which is 65,535 bytes long with the length of a header added in [44] [31] [46].

TTL (time to live) field: Time to live (TTL) is an 8-bit field that indicate the maximum time the datagram shall live in the internet system. The time here is measured in seconds, and in case the value of TTL is zero, the datagram is erased. Every time a datagram is processed, it's Time to live decreases by one second. It is usually a measure of the number of hops (routers) an IP datagram can visit before it is discarded. The intent of the field is to catch packets that have been going around in routing loops and discard them. The current default value is 64 seconds. These are used so that datagrams that are not delivered are discarded automatically. TTL can be between 0 – 255 [44] [31] [46].

Protocol: The Protocol field is of 16-bit length, the Protocol field is simply a demultiplexing key that identifies the higher-level protocol to which this IP Packet should be passed. There are values defined for the Transmission Control Protocol (TCP is 6), User Datagram Protocol (UDP is 17), for ICMP it is 1, IGMP it is 2, OSPF it is 89 and many other protocols that may sit above IP in the protocol graph [44] [31] [46].

Checksum: The checksum field is of 16-bit length and it is used to check the header for any errors or it is used to maintain the integrity of the IPv4 header. The header is compared to the value of its checksum at each hop, and in case the header checksum is not matching, the packet is discarded. Since a corrupted header may contain an error in the destination address and as a result, may have been misdelivered, it makes sense to discard any packet that fails the checksum [44] [31] [46].

Source IP Address: It is a 32-bit address of the source of the IPv4 packet. The source address is required to allow recipients to decide if they want to accept the packet and to enable them to reply [44] [31] [46].

Destination IP Address: The destination address is also 32 bit in size, and it contains the receiver's address and that it is the key to datagram delivery. For every packet contains a full address for its intended destination so that forwarding decisions can be made at each router [44] [31] [46].

Options (Variable): This is an optional field of the IPv4 header. It is used only when the value of Internet Header length (IHL) is set to more than 5. Note that the minimum valid Internet Header length (IHL) is 5, when there are no options. These options contain values and settings for things related to Security, Record route, time stamp and padding etc. You find that the list of options component ends with an End of Options or end-of-list (EOL) in many cases. The presence or absence of options may be determined by examining the header length (HLen) field [44] [31] [46].

Pad (Variable): Padding is basically used to make sure that the IP Packet header has a length that is a multiple of 32 bits. It is needed because of the varying length of the options field in the IP header. If the IPv4 option does not use all 32 bits, padding options should be added so that the IPv4 header is an integral number of 4 byte blocks that can be indicated by the internet header length (IHL) [44] [31] [46].

Data (Payload): The payload of an IP Packet is typically a datagram or segment of the higher-level transport layer protocol, but may be data for an internet layer (for example, ICMP or ICMPv6) or link layer (e.g., OSPF) instead [44] [31] [46].

2.2.7 **Related Work**

Compression and decompression of data files for storage is essentially the same task as sending and receiving compressed data over a communication media such as Wireless Local Area Network (WLAN). There has been a substantial body of work on IP Packet Compression in Wireless Local Area Networks (WLANs) and other networks such as Mobile Ad Hoc Networks (MANETs) and Satellite networks. Much of the existing work on IP Packet Compression tackles the subject from the standpoint of data transmission, and this is an active field of research. This section gave a review of related literature review (works) on IP Packet Compression and Decompression:

Mate Tomoskozi et al. [48] states that in modern cellular networks utilizing the Long Term Evolution (LTE) set of standards face an ever increasing demand for mobile data onto connected devices. IP Packet Header compression could be employed to minimize the overhead for IP based cellular network traffic. The study evaluated three header compression; Real Time Protocol (RTP), User datagram protocol (UDP) and Transmission Control Protocol (TCP) profile implementations used by such networks with respect to their potential throughput increase and complexity for different mobile service scenarios. The results indicate that all implementations have great potential for saving bandwidth in IP based wireless networks, even under varying channel conditions [48]. The research does not suggest that the approach used would achieve the best saving of bandwidth in IP based wireless network [48].

Jianan Sun et al. [49], carried out a study on improving bandwidth utilization by compressing small payload traffic for vehicular networks. They argue that the low bandwidth utilization is a serious problem with mobile services over vehicular networks mainly resulted from the high-rate transmission of packets carrying small payloads. Some solutions were proposed to forward packets by context locator/identifier, instead of IP address, based on header compression and software-defined networking technology. The researchers proposed to utilize a forwarding locator/identifier to indicate the compressor's location, separating the header compression process of the packet forwarding process. Extensive simulations were conducted and the results demonstrated that scalable end-to-end header compression experiences outstanding performances in bandwidth utilization and delay, showing its greater suitability for vehicular network transmission optimization. They proposed a scalable end-to-end header compression scheme, which takes advantage of the locator/identifier separation concept [49].

Yukon Niu et al. [50], asserts that header compression can effectively improve the bandwidth utilization, especially in satellite links which have limited bandwidth. The existing header compression schemes compress packet headers from network layer. Media Access Control (MAC) header can play a role in routing since forwarding rules can be dynamically deployed based on input port, Internet Protocol/Media Access Control (IP/MAC) addresses etc. In this case, we can also compress MAC header before packets are transmitted through satellite links. The study proposed backfill scheme to compress packets of MAC header in Open Flow network of satellite links and get Context Identifier (CID) back to the existing frame format after generating it. Open Flow network makes it possible to route header-compressed packets. Three kinds of traffic patterns were considered (MAC/IP/TCP, MAC/IP/UDP and MAC/IP/UDP/RTP). Both IPv4 and IPv6 were tested, and higher header compression ratios (more than 86% for IPv4 and 89% for IPv6) were obtained for the considered kinds of traffic. The results shows that backfill has higher header compression ratios than other header compression schemes [50]. Though, the research indicate that backfill had higher header compression ratios for the above traffic pattern in satellite links, it may not yield the same results when tested in WLAN platform with different kind of payload and traffic patterns.

Mate Tomoskozi et al. [51], posits that 5th Generation (5G) mesh networks could suffer from a protocol encapsulation overhead that can outweigh the amount of data one is supposed to send. The study reviews that a solution would be to employ header compression algorithms in order to reduce the size of the individual protocol headers. Robust Header Compression (RoHC) is an already established standard that tackles IP-based compression well. The current scheme at the time of undertaking the study did not allow deployment of wireless mesh networks as the compression works on a peer-to-peer, single-hop basis. The study shows that one can cut the payload delivery overhead in half for RTP transmissions even in cases when standard such as RoHCv2 would struggle and potentially fail, and that we can even improve on RoHCv2 under the assumption of erasure channels. The ongoing research focused on decreasing the coding overhead even further by applying progressive shortening to header compressed packets and evaluating it in a real mesh environment, as well as, generalizing the header compression schemes for the application to protocol headers [51]. The study was tested only for RTP transmission, the results may not be similar in TCP and UDP transmission scenarios.

Carlos Feres [52], states that packet-switched wireless networks such as 4G and 5G cellular systems apply Robust Header Compression (RoHC) to reduce Packet Data Convergence Protocol (PDCP) header length and improve payload efficiency. Recent works have demonstrated the benefit of applying a trans-layer approach that exploits lower layer information on RoHC control based on a Partially Observable Markov Decision Process (POMDP) formulation. The benefit of the POMDP solution comes at significant computation complexity of existing RoHC. The study focused on simplicity by designing RoHC compressors with lower layer awareness including channel adaptive transport block size as a result of link adaptation present in many wireless networks. The new models directly address practical implementation and can deliver transmission efficiency close to an optimized POMDP compression [52].

Tomoskozi, Mate; Lucani, Daniel; Fitzek, Frank H. P and Ekler, Peter; [53], argue that compression approaches reduce the protocol encapsulation between two endpoints of a link. The reduction in protocol overheads relies on redundancies found in the different headers of individual packets and between consecutive packets belonging to the same IP-flow. A common design theme all over the various compression standards are that the compressor chooses the smallest packet types from a given compressed packet pool which it deems sufficient for the successful transmission of the header fields. The study was carried out and it proposed a solution that circumvents some concerns of traditional header compression context initialization by employment of network coding, which is called the reliable base technique. The technique provides a finely tunable method of balancing reliability and delay of decompression with bandwidth gain. The results shows that both compression gain and reliability can be increased over the previous standards [53].

Cha, Hyejin et al. [54]; asserts that with multimedia streaming network, voice and video packets is transmitted through IP, UDP and RTP. Redundant header data onto contiguously transmitted packets that can result in the inefficiency of data transmission. To prevent this, RoHC (Robust Header Compression) mechanism is applied. It uses a Window Least Significant Bit (WLSB) which is an encoding method using the value of sliding the window. The study presented an adaptive method which adjusts the value of sliding the window in RTP packet transmission. The study states that the value of sliding the window is adjusted according to the number of packet loss. The experimental results from the proposed method indicate that header compression ratios are increased and the average header size is decreased as the value of sliding the window is adjusted.

Hung, Brian; Defrancesco, Dana; Cheng, Bow-Nan; Sukumar and Prasanna, [55], argues that with wireless communications, improving bandwidth efficiency is a necessity in the presence of limited spectrum resources and increasing demand. This is especially true of WLAN and satellite communications where resources are shared among many of the users. The Joint IP Modem (JIPM) is a satellite modem system that enables extension of Global Information Grid (GIG) services to remote warfighters. One method to improve Satellite Communication (SATCOM) bandwidth efficiency is header compression, which reduces the size of the packet header and the overall packet size. The study examined by applying Robust Header Compression (RoHC) to the JIPM system and evaluated its performance over various scenarios [55]. The increased demands on SATCOM networks, decreasing network payload enables additional users to leverage JIPM capabilities. The researcher examined leveraging a commercial header compression scheme for usage in the Joint IP Modem system.

From the results and detailed analysis, it showed that IPHC can increase bandwidth efficiency by reducing the traffic volume and improve the VoIP application performance [55]. However, the research indicate that IPHC can increase bandwidth efficiency by reducing the traffic volume and improve the VoIP application performance in SATCOM network of which the results may be different if used in a WLAN platform and using Video as the payload.

Chauhan, Dipti; Jain, Jay Kumar and Sharma Sanjay, [56], asserts that with the exponential growth of internet, it's impossible to sustain with IPv4 protocol due to its limited space capability and the only option is to move towards new next generation internet protocol IPv6. Different transition techniques have been proposed to enable the smooth interpretation of the two protocols: Dual Stack, Tunneling, and Header Translation. Tunneling is generally used solution to carry an IPv6 packet of the IPv4 network. Tunneling comes with several imperfections like inefficient routing, header overhead due to multiple headers present, Quality of service and high bandwidth usage. The study proposed a new approach to compress the header of the packet, which would improve the efficiency of the packet tunneling mechanism. The study addressed the bandwidth issue for multi-hop IPv6 tunnels over wired and wireless networks by simulating the algorithm over small scale networks with limited nodes, even better results could be obtained when tested on large scale networks [56].

Matias and Refua [12], argues that data compression in IP Packet networks, is transmitted by partitioning it into IP Packets. IP Packet Compression allows better bandwidth utilization of a communication line resulting in much smaller amounts of the packet drops, more simultaneous sessions, and smooth behaviour of applications. IP Packet Compression can be obtained by a combination of header compression and payload compression, which are complementary methods [12]. However, the study does not suggest or recommend the use of (Delayed Dictionary-based Compression) DDC as the best approach for optimisation.

Measel R [57], state that IPv6 offers many advantages over its predecessor in terms of flexibility, security and routing in addition to a vast larger addressing space. A great amount of overhead accumulates for IPv6 transmissions with small packet sizes, such as network management and real-time traffic. The study stated that payload compression should only be employed for streams or packets that are less than

768kbps due to the processing delay time that could increase as the size of the packet increases [57], and when the processing time exceeds the transmission time, the compression algorithm will actually begin to decrease bandwidth. Several header and payload compression techniques were proposed in the study to mitigate the overhead. The study reviewed that RObust Header Compression (RoHC) was found to perform the best under network constraints with the exception of packet reordering. Payload compression techniques were less useful as they require overhead in the form of either a per-packet dictionary or a synchronization mechanism and data payload cannot be compressed or encoded.

Lubobya et al. [58], investigated the computation time of Single Input Multiple Data (SIMD) when compressing static video packets (or datagrams) on a stand-alone personal computer. In their study, they achieved SIMD computation time increases of 14,000ms and 17,000ms for pairs of Integer DCT and Hadamard transforms, which were limited to static video that was compressed without sending the video over a Wireless Local Area Network (WLAN).

2.3. **Comparison to this Study**

In view of the reviewed related literature (works) in the preceding sections, the literature reveals that not much has been done to address the identified gaps such as network delays, network congestion, buffer, and bandwidth requirements in a WLAN using OPNET tool to optimize the IP Packet network problems associated with IP Packet transmission. The benefits of performing network simulations are many, some of which are listed as follows: i) With Information and Communication Technology (ICT), new ideas can be tested, developed faster and at a lower cost using Computer Aided Engineering (CAE); ii) The network simulation approach helps in locating any unusual behaviour or phenomena that may be costly or difficult to locate in the actual network; iii) Saving money and time is a network simulation's main benefit; iv) Additionally, if a mathematical approach cannot be used to solve the problem, simulation is the most suitable for investigating complex and large-scale practical situations, it is adaptive, adjustments can be made to the system variables to find the optimal option from a range of options and experiments are performed with the model without disturbing the system.

In this study, an effort was made to focus on simulating the IP Packet Compression and Decompression in WLAN by critically considering the packet overhead and payload that was validated through the results in realistic WLAN using OPNET tool in a full function network platform. For the purpose of the simulation, video conferencing transmission was configured for testing to achieve the research objectives and be able to answer key research questions.

2.4. **Summary**

The study reviewed several related work concerning the IP Packet Compression and decompression. The theoretical framework in the study is guided by two theories namely; i) Information Theory and Rate Distortion Theory. Information theory is the mathematical treatment of the concepts, parameters and rules governing the transmission of messages through communication networks. Rate-Distortion Theory is the branch of information theory that treats compressed data produced by an information source down to a specified encoding rate that is strictly less than the source's entropy.

CHAPTER THREE

RESEARCH DESIGN AND METHODOLOGY

3.0 Introduction

This section dealt with research procedures for gathering data, variables involved, an explanation of how data was going to be analyzed and any other information that may be required to efficaciously achieve the aim and objectives of the study. Methodology is the general approach that the principle investigator would critically consider in carrying out the study; to some extent, this approach dictate particular tools, techniques and procedure or a step-by-step process that the researcher would select [59]. In other words, it is a way to systematically solve the research problem or finding answers to the key evaluation research questions. The research methodology deals with general laws and principles of organizing the research activity by choosing an efficient (adequate or rational) research technique [60]. The path to finding answers to key evaluation research questions constitutes what is called research methodology [59] [60] [61]. Therefore, this section was organized through the following sub-sections: research approach, research design, research tools review, selection of simulator software, IP Packet Compression model development process, Network Performance Metrics, Network Topology System Design, Simulation Setup, Implementation of OPNET Simulations, Data collection and analysis, reliability and validity procedures by using OPNET Simulator tool [62].

3.1. Research Approach

Research approaches is used to answer key questions/test hypotheses and that the results are based on actual evidence, as opposed to theory or assumptions. Research approaches are plans and procedures for research that span the steps from broad assumptions to detailed methods of data collection, analysis and interpretation based on the nature of the research problem/questions being addressed and the reasoning behind the choice. The approaches are essentially of three (3) types namely; quantitative, qualitative and mixed research approaches [59] . Theoretical research leads to a better understanding of science with the results of the experiment. With computers, new ideas can be tested, developed faster and at low cost. In this study we took a similar line of thinking by using a combination of theoretical research with empirical review and undertook an experiment (in form of simulation) to answer the key research questions and that no questionnaire approach was administered in this study. The mixed approach integrated the two forms with the view of providing a more comprehensive understanding and that it would have more impact on the results and enhance the validity and reliability of research findings [59] [60] [61].

3.2. **Research Design**

The researcher designed the Wireless Local Area Network that would have all the components for source of data to be compressed and the destination of the compressed data. The type of multimedia information (video conferencing) was selected for the study. Each type of the compression schemes of multimedia was highlighted in each scenario that was created and it included; None-no compression where there was no compression, TCP/IP Header Compression where there was only TCP/IP Header compression, Per-Interface Compression where there was the entire packet being compressed, and Per-Virtual Circuit Compression where there was only packet's payload (i.e., not the header information) compressed. The researcher further considered all the procedures required in simulating the study through the use of OPNET network simulator software to simulate the behaviour and performance of the network. Under this section all the procedures for setting-up OPNET network simulator software was designed to ensure that the result that would be obtained depicts what could be the real outcome of the actual situation.

3.3. **Research Tools Review**

A network simulator is a software program that imitates the working of computer network. In simulators, the computer network is typically modelled with devices, traffic etc., and the behaviour and performance are analyzed [63]. Simulation software is an important platform of finding results that cannot be obtained from a practical hardware setup which is costly and tedious to modify frequently [64]. Network simulators are widely used by researchers to evaluate new theories and hypotheses [65].

Under this section, a survey and review for existing network simulators were considered. There are a lot of Network Simulators in the communication world. Some of them are dedicated wireless network, some of them are dedicated to wired or both types of networks. Considering the wide variations in operating systems, hardware requirements, software requirements, output features and scalability, it is challenging to select an appropriate simulator for a specific study. The appropriate network simulators that satisfy the physical physiognomies of the study include; Commercial software - OPNET modeler and Open Source - Objective Modular Network Testbed in C++ (OMNeT++) and Network Simulator Version 2/3 (NS2/3) among many others [66] [67] [68] [69].

The study has made the analysis of the three simulator tools as follows:

3.3.1 **OPNET**

OPNET was created by two Massachusetts Institute of Technology researchers in 1986. It is the leading network technology development environment in the industry at present. An object-oriented modeling method and graphical editor are used to effectively reflect the actual network structures and network components. An actual system can be visually mapped onto the model. Nowadays, OPNET has entered the fields of military, science, education, banking and network operation [66] [67] [68] [69].

Advantages of OPNET

i) Customizable wireless modeling
ii) Fast discrete event simulation engine
iii) Object-oriented modeling
iv) Grid computing support
v) Set of element library with source code
vi) Hierarchical modeling environment
vii) Scalable wireless simulations support
viii) Discrete Event, Hybrid, and Analytical simulation

Disadvantages of OPNET

i) Exactness of results is limited by the sample resolution
ii) It does not permit a set of nodes within a single connected device
iii) If there is prolonged inactivity in the simulation, it is ineffective.
iv) Complicated GUI operation

3.3.2 **OMNET++**

OMNET is a Modular Discrete Event Simulator implemented in C++. It provides a powerful library for animation and tracing and debugging support. Its major drawback is the lack of available protocols in its library, compared to other simulators [66] [67] [68] [69].

Advantages of OMNET++

i) Source code is openly available
ii) Not limited to network protocol simulation
iii) Simulation model for internet, IPV6, mobility is also accessible
iv) Extremely modular

Disadvantages of OMNeT++

i) Poor analysis and management of typical performance
ii) It does not offer a great variety of protocols
iii) The mobility extension is comparatively incomplete
iv) Users with significant background work

3.3.3 **NS3**

NS-3 is a C++ library which provides a set of network simulation models implemented as C++ objects and wrapped through python. It is targeted mainly for research and learning use and is free and open source software. Users interact with this library by writing a C ++ or python application to set up the simulation scenario of importance [66] [67] [68] [69].

Advantages of NS3

i) To allow for modular libraries
ii) The system has been modularized
iii) To allow the node to use external routing
iv) Individual modules contain with directory structure

Disadvantages of NS3

i) NS3 suffers from lack of credibility

ii) Modules, component based on NS2

iii) NS3 needs lot of maintainers

iv) Active maintainers are required

Table 3.1: Analysis and comparison of the three (3) network simulators.

Table 3. 1: Analysis and comparison of Simulators
Source [63] [64]

Features	**SIMULATOR NAME**		
	OPNET	**OMNeT ++**	**NS3**
License Type	Commercial	Open Source	Open Source
GUI Support	Excellent	Good	Poor
Supported Operating System or Platform	Windows XP, Vista, 7 and Windows NT	Windows XP, Linux, Mac OS X, and other Unix like system	SD, Mac OS X, Windows XP, Windows Vista and 7
Ease of Use	Easy	Easy	Hard
Document available	Good	Good	Excellent
Availability of analysis tool	√	√	√
Network Visualization tool	√	√	√
Per-Interface Compression	√	√	√
TCP/IP Header Compression	√	-	-
Per-Virtual Circuit Compression	√	-	-
Possibility to design and modify scenarios	√	√	√
Fast simulation capabilities	√	-	-
Communication with other modules	√	-	-
Interactions with real systems	√	√	√

For the purpose of this research, OPNET simulator tool was selected and used as it supports IP Packet Compression through the IP Compression Information Attribute in the IP Attribute Config object. By default, the IP Compression Information attribute in OPNET contains a set of pre-configured compression schemes that include; None - no compression, TCP/IP header compression, per-interface compression, per virtual circuit compression, image compression, and Telnet application compression [3] [66] [67] [68] [69].

3.4. Selected Simulator

OPNET was selected because it contains a vast amount of models of commercially available network elements and has various real-life network configuration capabilities. This has made the simulation of real-life network environment close to reality. Other features of OPNET include Graphical User Interface (GUI) including drag and drop approach for easy deployment of simulation networks devices, the core source code is not provided, and hence the inbuilt models cannot be modified as compared to OMNET++ and NS3 which provides the source code for all the modules. OPNET has a comprehensive library of network protocols and models, graphical results and statistics among others. Most importantly, OPNET has gained considerable popularity and the most trusted among all the network simulators in academia as it is being offered free of charge to academic institutions and can be used to evaluate almost any of todays' network paradigms. Furthermore, OPNET is considered by researchers as a reference simulator that has much larger scientific acceptance. OPNET has got a large online research and developer's community readily available through free mailing list. As a result, OPNET has been given an edge over other simulators on the market place and academia platform. The selected OPNET simulator tool was therefore used to simulate by designing a desired wireless local area network and component based configuration for this particular study [66] [67] [68] [70].

OPNET modeler compression method specifies the type of method of its IP Attribute Config object that accepts only one at a time of the following four values: i) None, no compression, ii) TCP/IP Header Compression, here only TCP/IP header was compressed, iii) Per-Interface Compression, the entire packet was compressed, and iv) Per-Virtual Circuit Compression, only the packet's payload (i.e., not the header information) was compressed.

3.5. **Model Development Process**

In this sub-section, the strategy was to describe a general plan about how the researcher intended to answer the key questions relating to the study. The study undertook four (4) scenarios as indicated in Figure 3.1, to determine the behaviour and network performance in terms of Effectiveness and Efficiency. The results of the research were then analyzed/interpreted, validated, and documented.

3.6. **Network Performance Metrics**

The network performance is the evaluation and analysis of all network statistics for all the four compression schemes to determine the level of service provided by the underlying computer network, usually from the perspective of the end nodes. The following were the network performance metrics considered in the study:

Throughput is the amount of information successfully delivered or received in a given amount of time (throughput is measured in bits per second or data per second) [21]. It is the actual amount of data that moves from one node to another across the medium in a given amount of time. Throughput uses data units at a time, such as bits per second (bps). The real or actual amount of data delivered from one node to another, taking into account extra considerations, is known as throughput [29] [30]. Since wireless bandwidth and the wireless medium are limited resources in networks, effective and efficient use of the network resources is vital. The throughput of the network indicates how well these resources are being used [71]. One of the most important measures used to gauge how well a network performance is operating is throughput, which is a fundamental idea for effective network performance [72] [73].

Throughput is considered high/good if most messages are delivered successfully which means that there is good network performance. When a Packet is compressed, the size of the packet decreases thereby increasing the throughput capacity as well as the transmission time of each packet. Compression reduces the amount of bandwidth consumed during IP Packet transmission in a WLAN. Conversely, low/poor delivery success rates lead to lower throughput. As throughput decreases, network performance decreases. Higher throughput is desirable because it means the system is able to process more data at once, which improves system network performance. The network performance and efficiency values are indicators of the network's overall Quality of Service (QoS) [21] [29] [30].

Data dropped (Packet Loss) is the total number of bits supplied by a wireless node but never received by another node for a variety of reasons, including network delays, congestion, queuing, and the likelihood of packet loss increases with packet distance. Packets lost during network transmission are known as packet loss or data dropped. [71] [74] [75] [76]. The number of packets dropped during data transfer on the network is measured by packet loss. Data requests will be fulfilled more slowly when more packets are lost or dropped. Packet loss may also be due to lack of media availability, which has a big impact on WLAN dependability [72] [71].

Packet loss occurs when packets are lost during data transmission due to network congestion, hardware problems, software glitches, and various other reasons. The impact of packet loss varies by application and protocol (TCP/UDP). TCP is designed to resend dropped or unacknowledged packets. However, due to lack of retransmission capability of UDP, the ability to control packet loss is poor. When more packets are transmitted from one end node to the other, it means there is good network performance [72] [71].

Network load is the volume of data that flows over a computer network at any given time (also known as data traffic or network traffic) [70] [73]. Network Traffic is the amount of data moving over a computer network (Wireless Local Area Network) [77]. Data traffic, often referred to as network traffic, is divided into packets and sent over the network before being reassembled by the computer or other device that receives it. Traffic has an impact on network performance since exceptionally high traffic might make downloads to take longer time [70] [73] [77].

A network may experience performance issues if a large number of WLAN devices attempt to connect to the network at once. You will need to invest in equipment that can handle high throughput as bandwidth requirements on the network rise in order to maintain high levels of performance and network dependability. For example, video streaming or video conferencing are bandwidth-intensive application.

Delay is the amount of time it takes for a packet to be processed in a computer network. This comprises processing delays, queuing delays, transmission delays, and propagation delays. Delays are an essential statistic for evaluating network performance. Propagation delay, transmission delay, queuing delay, and processing delay are added together to determine the packet delay [69] [70]. A network is considered to have good performance when there is less network delay during packet processing in a network.

Media Access Delay is the time it takes for data to reach the Medium Access Control (MAC) layer and be successfully transported out on the network's interface is referred to as Media Access Delay [69] [70]. Media Access Delay is the total amount of time queue and contention delays of data packets it takes for the head of the signal to access a medium [75] [73]. A network is considered to have good performance when there is less delay to reach the Medium Access Control layer.

The model was developed using the following process shown in Figure 3.1: Research Methodology Process.

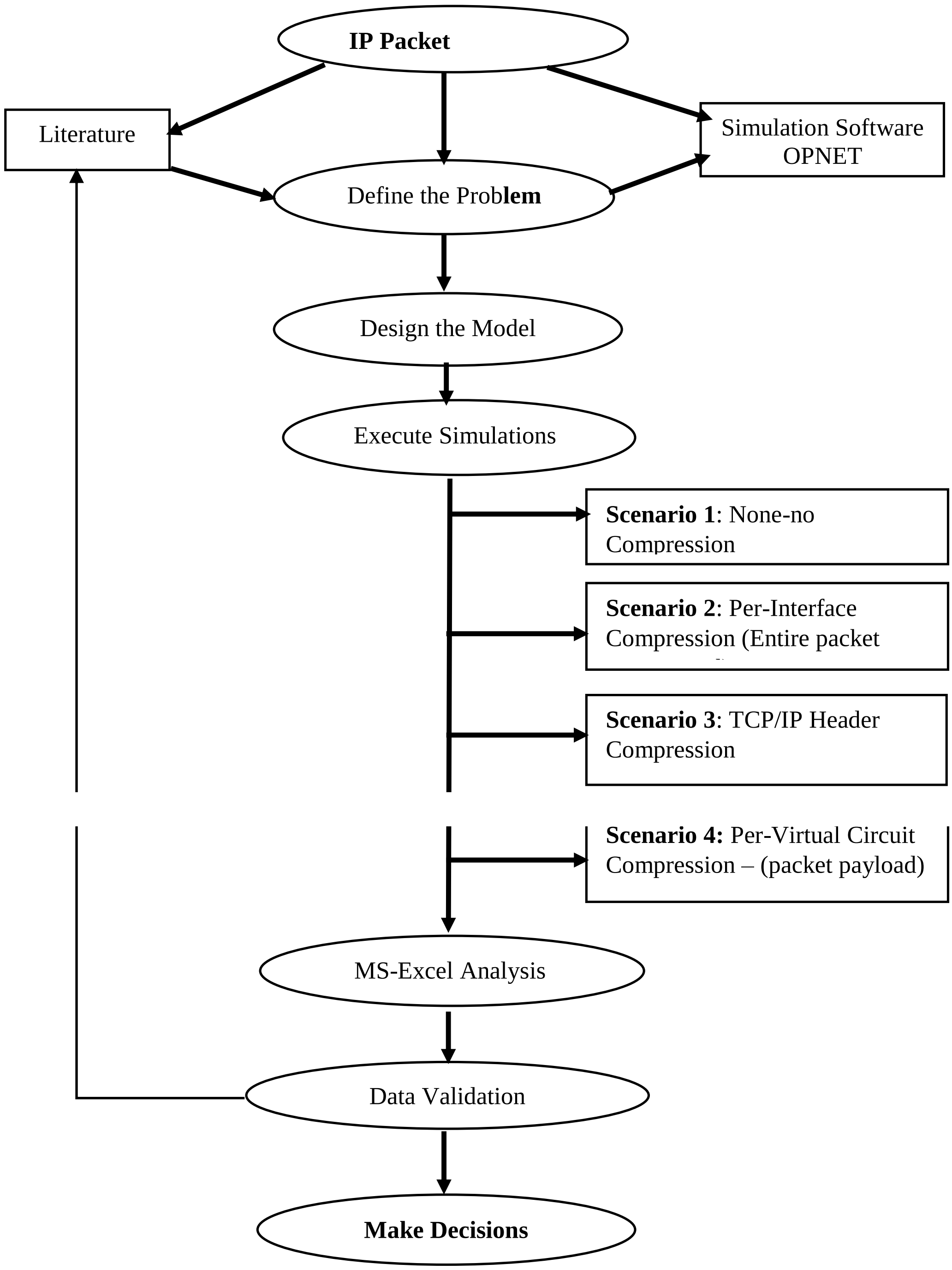

Figure 3. 1: Research Methodology Process

Source: [79]

3.7. **Network System Design**

The Network Topology implementation of the Wireless Local Area Networks (WLAN) was done by designing the network that included the Server, Access Point (AP), IP-Cloud, Gateway Router (Ethernet4_slip_gtwy), Seven (7) Wireless Workstations, Applications, Profiles, Task and IP Attribute Config object as shown in Figure 3. 2: WLAN Topology. The 100base-T cable was used to connect the devices from Access Point (AP) to the Gateway router and from Gateway router to the IP cloud Point-to-Point Protocol (PPP) Digital Signal 3 (DS3) normally called PPP_DS3 was used. Similarly, from IP Cloud to PPP Server, PPP_DS3 cable was used. It is important to note that the 100Base-T link represents an Ethernet connection operating at 100Mbps (i.e. 10 times faster than standard Ethernet). The Basic Service Set Identifier (BSSID) was set to 1 for all the nodes and Access Point (AP). The BSSID is the Media Access Control (MAC) physical address of the Access Point or wireless router that is used to connect to the Wi-Fi and that the term is used in wireless network. The Basic Service Set (BSS) is the cornerstone topology of any 802.11g network. Figure 3.2: WLAN Network Topology.

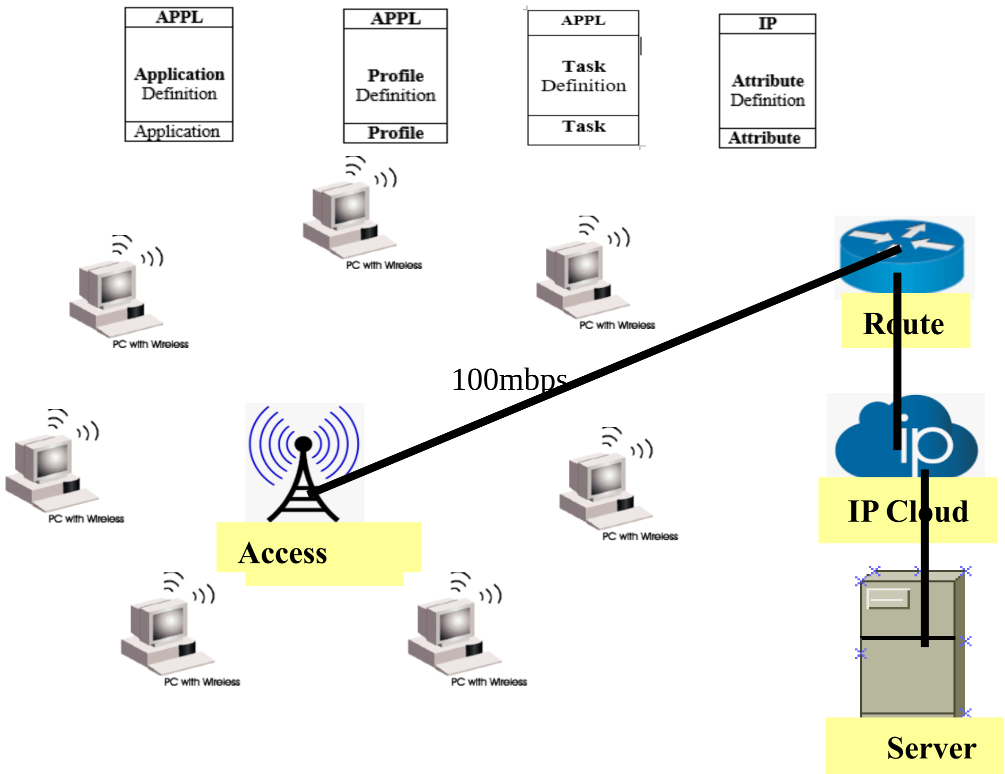

Figure 3. 2: WLAN Network Topology
Source: Researchers design March (2023)

3.8. OPNET Simulation Setup

In this study, the Access Point and mobile node parameters were set as shown in table 3.2: Access Point and Mobile node parameters:

Table 3. 2: Access Point and Mobile node parameters

Wi-Fi 802.11g		
	AP (Access Point)	**Mobile Node**
Tx Power	0.1W	0.1W
Data Rate	11Mbps	11Mbps
Received Power Threshold	-95dBm	-95dBm
Buffer Size	1024000 bits	256000 bits
Short retry Limit	7	7
Long Retry Limit	4	9
Traffic Sent	Video	Video
Simulation Time	1200 Seconds	1200 Seconds
Large Packet Processing	Fragment	Fragment
Access Point Functionality	Enabled	Disabled

All other dataset (settings) of the study were configured as per research standard requirements of the OPNET Simulation tool.

3.9. Simulation process

To simulate a network, statistics have to be chosen about the network, global statistics which will tell the results of the whole network rather than concentrating on single node or single aspect of any node within a network. In this report, global statistics for WLAN have were selected in which Throughput, Data dropped, Delay, Network Load, Traffic sent and Media Access of the WLAN network were selected.

3.10. Implementation of OPNET Simulations

The modelling and simulation of a network is performed in OPNET (Optimized Network Engineering Tool), which is a commercial tool. It covers a wide range of protocols for both wired and wireless networks and is applicable to all network types. By compressing the datagram, the packet size is reduced. In OPNET, this functionality is supported through IP Attribute Config object that allows one to specify preferred compression algorithms, and you may deploy them on the node interfaces of your choice. A number of present compression algorithms are included by default in the IP Compression Information attribute that include None-no compression, TCP/IP Header Compression, Per-Interface Compression, and Per-Virtual Circuit Compression. This attribute accepts only one of the aforementioned four compression methods at a time, according to the OPNET Compression Technique, which defines the kind of compression method [3].

3.11. **Data Collection**

Mugenda [15], reviews that depending on the kind of research, the primary data gathering process may differ from one study to the other. The research instrument tool of data collection that was used during this study is OPNET network simulator tool that was configured accordingly to address the key questions within the study phenomenal [15].

3.12. **Simulation and Analysis**

The approach used OPNET Modeler 14.5 Release simulation package. The researcher determined the best method of analyzing and presenting data obtained from OPNET network simulator tool so that it is well sorted and arranged. The following metrics were investigated for the IP Packet Compression simulation: throughput, data loss, delay, network load, traffic sent, and media access. We then measured what OPNET tool supports such as network performance and quality of service through performance metrics that include Throughput, Data dropped, Delay, Network Load, Traffic sent and Media Access.

The OPNET Modeler simulator was selected because it supports IP Packet Compression through its IP Attribute Config object that accept only one of the following four (4) values: i) None-no compression, ii) TCP/IP Header Compression, here only TCP/IP header was compressed, iii) Per-Interface Compression, the entire packet was compressed, and iv) Per-Virtual Circuit Compression, only the packet's payload (i.e., not the header information) was compressed. We simulated all the four

(4) scenarios that the study undertook. Therefore, in this research the data analysis application namely Microsoft Excel application for qualitative and quantitative analysis was used.

3.13. Study Work

A Wireless Local Area Network (WLAN) was created for the purpose of simulating the IP Packet Compression and Decompression for delivery. The steps in the research procedures for creating the scenarios included the following:

i) Literature study
ii) Define the Problem
iii) Create and Design Simulation Model
iv) Create Interior and Exterior Routing Configurations
v) Select Input/output Parameters
vi) Choose the statistics
vii) Run the Simulation (Scenario)
viii) View the simulation results
ix) Duplicate the scenarios
x) Make Changes
xi) Re-run the simulation
xii) Compare and Analyse the obtained results to get answers to key research questions

3.14. Reliability and Validity

Validity in any study is concerned with the degree to which the data gathering instrument measures precisely what it is expected to measure whereas reliability is the level to which an instrument can consistently obtain the same results in another setting that is denoted as internal validity. Internal validity was obtained in this study by conducting testing with the OPNET simulator program.

3.15. Limitations of the Study

During conducting of this study, the following are the limitations:
i) OPNET is proprietary software which is limited in terms of customizability and this can have a significant effect on the simulation results.

3.16. **Summary**

The chapter, discussed the methodology that was used to conduct the research. The study evaluated the Efficiency and Effectiveness of IP Packet Compression and Decompression for transmission in a Wireless Local Area Network (WLAN) with the view of optimizing and/or mitigating these delays, buffer requirement and congestion challenges/problems. The study used the OPNET Modeler simulator, chosen because it supports IP Packet Compression through the IP Attribute Config object that accept only one of the four compression schemes namely: i) None-no compression, ii) TCP/IP Header Compression, iii) Per-Interface Compression, and iv) Per-Virtual Circuit Compression. The four (4) IP compression attributes was the scenarios that the study carried out during simulation.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.0 Introduction

This Chapter forms the core section of the study that outlines the data analysis and discussion of the results obtained from OPNET simulation software tool. The results of the study are presented and discussed with reference to the aim of the study, which was to evaluate the effectiveness and efficient of compression and decompression of IP Packets in a WLAN using four (4) compression schemes considering performance and Quality of Service, in coming up with optimized and/or mitigated IP Packet delivery challenges. The OPNET simulations which were undertaken was aligned with data and information to be captured that answers the three (3) key evaluation study questions in form of research scenarios. Each of the research scenario created is followed-up with data presentation and discussion of the results. The principal focus was performance and Quality of Service and the measure of performance metrics were; Throughput, Data dropped, Delay, Network Load, Traffic sent and Media Access Delay. The goal in this section was to model a WLAN that has sections consisting of the introduction to the chapter, simulation results, discussion of the results and summary of the chapter.

4.1. Results and Discussion

In this subsection we present and discuss results of OPNET simulations obtained using the method that was described in Chapter three (3). The study undertook OPNET simulations through modelled WLAN using four (4) compression schemes namely: i) None-no compression, ii) Per-Interface Compression, the entire packet is compressed, iii) TCP/IP Header Compression, only TCP/IP header is compressed, and iv) Per-Virtual Circuit Compression, only the packet's payload is compressed. The compression schemes were simulated as separate scenarios and each scenario had Seven (7) client/users, the duration for each scenario was 1200 seconds (i.e. 20 minutes) and the performance metrics that were measured are; Throughput, Data dropped, Network Load, Delay, Traffic sent, Delay and Media Access Delay. These performance metrics are determining factors in term of overall behaviour, network performance and quality of service of the WLAN. As the simulation of the designed scenarios are completed, the data collected were exported to Microsoft Excel for further analysis of the results and comparison discussion in form of graphs.

4.1.1 **Throughput**

Throughput is the measure of how much data is successfully sent or received each second (throughput is measured in bits per second or data per second) [21]. Throughput refers to the amount of data that can be transmitted between two nodes [29] [30]. Effective use of the network resources is essential because wireless bandwidth and the wireless medium are limited resources in any network. The network's throughput reveals how well these resources are being utilized. Throughput was measured between each node connected to the access point [71] [72] [73]. The graphs for each of the four (4) compression algorithms were then compared. The outcome of IP Packet Compression schemes from the OPNET simulation software are as shown in Figure 4.1. The labels on the x-axis (horizontal) represent the time taken in seconds, whereas the labels on the y-bar (vertical) shows throughput in bits.

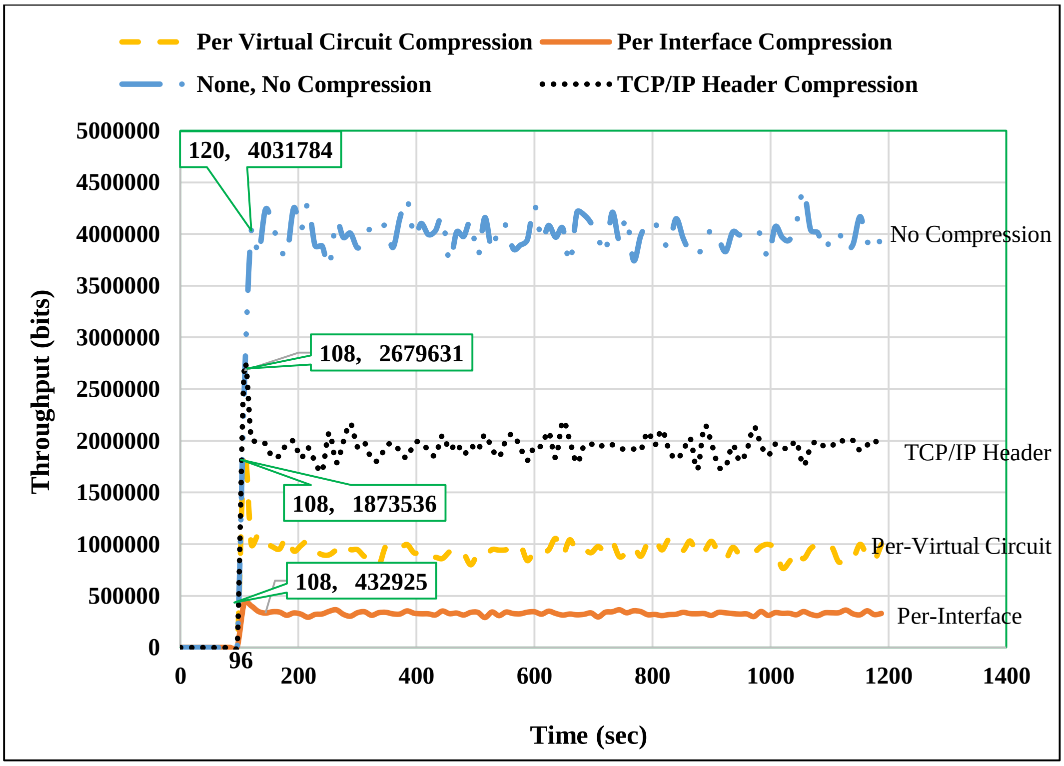

Figure 4. 1: Throughput (bits/sec)

Figure 4.1 shows that all the graphs for compression schemes started to rise when the WLAN connection was established, which took time from 0 to 96 seconds. According to the data, the Per-Interface compression's throughput rose from 0 to 432,925 bits while its processing time increased from 96 to 108 seconds. Per-Virtual Circuit compression’s throughput rose from 0 to 1,873,536 bits, and its processing time increased from 96 to 108 seconds. TCP/IP Header Compression's throughput rose from 0 to 2,679,631 bits and the time increased from 96 to 108 seconds. The throughput for None-no compression rose from 0 to 4031784 bits and the time increased from 96 to 120 seconds.

In explaining the results in Figure 4.1, we note that throughput is one of the parameter metrics used to measure performance and Quality of Service of the network. The performance of Throughput is determined by the effective and efficient use of the limited resources in a network such as the available medium and the kind of compression scheme used:

i) When the graphs in figure 4.1 trend upward over time and as the number of clients/users increases, it means that the server is successfully handling all the requests and that the network bandwidth is sufficient.

ii) When the graphs become comparatively flat towards the right side as time goes on and the number of clients/users rises, the transmission channel has low bandwidth, the network is congested, or the internet connection is poor. Throughput is the actual amount of data being transmitted from clients to the server, whereas bandwidth is the internet transmission medium's maximum capacity. Such flat graph can be seen which indicate that the transmission has reached a steady state given the nature of video conferencing data.

iii) We can therefore, state that Per-Interface compression had the lowest performance indicator for throughput (432,925 bits/sec) because the IP Packet had been compressed thereby creating more room for other packets to be transmitted so as to achieve the best performance resulting to good Quality of Service for the WLAN. When a Packet is compressed, the size of the packet decreases thereby increasing the throughput capacity as well as the transmission time of each packet. The No compression scheme had the highest throughput (4,031,784 bits/sec) as the scheme could not use the available resources effectively and efficiently and probably had poor network connectivity which could not achieve the desired performance and Quality of Service for the WLAN.

iv) When using compression, we gain time because the frames are smaller and the overall transmission delay is reduced. However, we also lose time because the compression and decompression processes consume a certain amount of time. The above two factors must be balanced to determine the effectiveness and efficiency of the compression scheme.

Although the throughput between the clients and the access point using compression is less than the throughput between the clients and the access point that is not using the None-no compression, the response time for the client using compression is higher. In other words, the client does not benefit from the transmission of the smaller frames and instead loses time during the compression and decompression operations.

4.1.2 **Data Dropped**

The number of bits that a wireless node sends but the receiving node never receives is referred to as data dropped. This can happen for a number of reasons, that include network delays, network congestion, access point or router overload, queuing, and the probability of data loss increases with packet distance [71] [74] [75] [76]. Additionally, it could be as a result of propagation delay within the network [72] [71]. The parameter metric Data Dropped was used to compare the four (4) compression techniques under consideration using graphs. Figure 4.2 shows the results of the OPNET simulation's IP Packet Compression algorithms. The labels on the x-axis (horizontal) shows the time taken in seconds while the labels on the y-bar (vertical) reflect Data Dropped in bits per second.

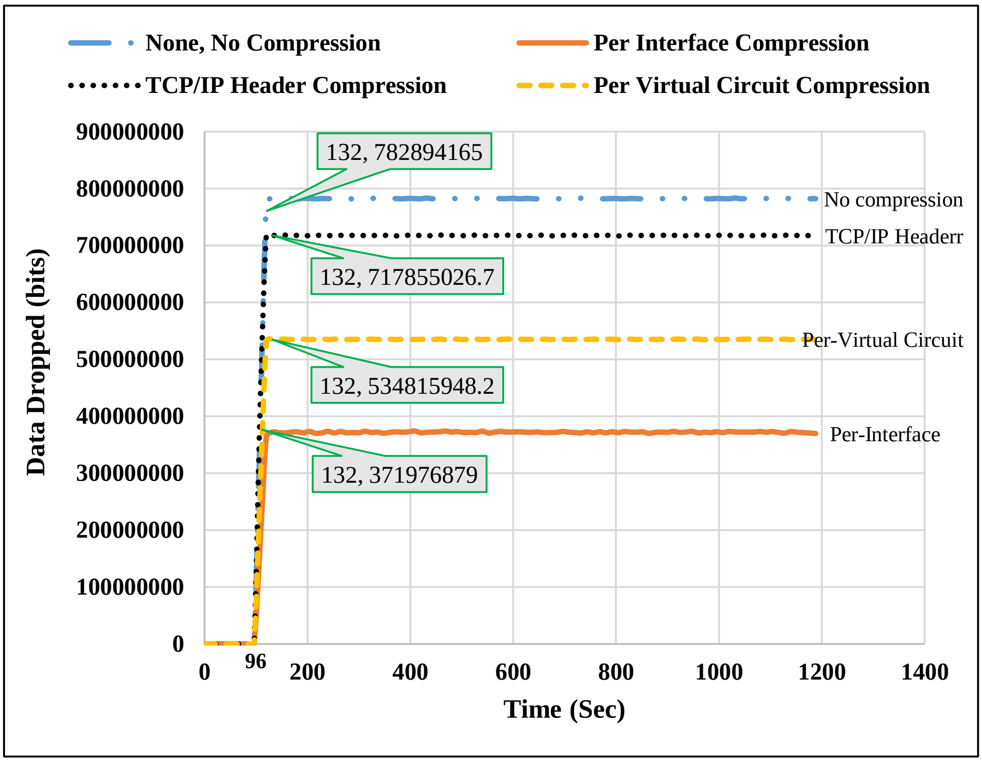

Figure 4. 2: Data Dropped (bits/sec)

Figure 4.2 shows that all the graphs for compression schemes started to rise when the WLAN connection was established, which took time from 0 to 96 seconds. According to the presented data, the Per-Interface compression's Data dropped rose from 0 to 371,976,979 bits. Per-Virtual Circuit Compression’s Data dropped rose from 0 to 534,815,948.2 bits and the time increased from 96 to 132 seconds. TCP/IP Header Compression’s Data dropped rose from 0 to 717,855,026.7 bits and the time increased from 96 to 132 seconds. The Data dropped for None-no compression rose from 0 to 782,894,165 bits and the time increased from 96 to 120 seconds.

In explaining the results in Figure 4.2, we note that data dropped is one of the parameter metrics used to measure performance and Quality of Service of the network:

i) The Operating System had a lower capacity than it was intended for at the start-up of the network. This may have made the network less robust, making it difficult to handle the requests or packets, resulting to lost packets.

ii) A WLAN may potentially lose packets as a result of weak signals, distance, and actual obstructions like walls, trees, among others.

iii) Once the graph becomes relatively flat as the time progresses from 96 to 132 seconds, then there is sufficient bandwidth or all requests are being processed by the server without any queue errors. If the issue initially of Data Dropping was with bandwidth as clients/users were trying to access the sever at the same time, then we can see a flat graph like the one in figure 4.2.

If the graph shows some drop during steady state, it means that the network bandwidth is more than sufficient and the server can handle all the requests without any queue errors:

i) Per-Interface compression scheme had the lowest Data dropped (371,976,979 bits/sec) and better network connection to be able to achieve high performance, resulting in good quality of service in the WLAN. This is attributed to the compression that had taken place for both the header and payload as the frames become smaller and the overall transmission delay reduces despite the scheme losing time due to compression and decompression processes that consume a certain amount of time, though response time may be faster.

ii) The compression scheme with the highest data loss (782,894,165 bits/sec) was None-no Compression which had poor network connectivity and packets were not compressed which could not achieve good results in a WLAN compared to other compression schemes.

The Per Interface compression had the lowest Data dropped (432,925 bits/sec) in this case which had good network connectivity that achieved good results in a WLAN. In video conferencing transmission, a single packet drop can result in the video impairment and could degrade the video quality. When there are many packet drops in a short period of time, that may lead to poor video quality that may become unacceptable.

4.1.3 **Network Load**

The volume of data that passes through a network at any particular time is known as network traffic also called Data traffic [70] [73]. In the study, the network load of the four (4) investigated compression algorithms were compared. The results of the compression schemes from OPNET simulations software are displayed in Figure 4.3.

The labels in the y-bar (vertical) reflect Network Load in bits/second, while the labels on the x-axis (horizontal) shows time taken in seconds.

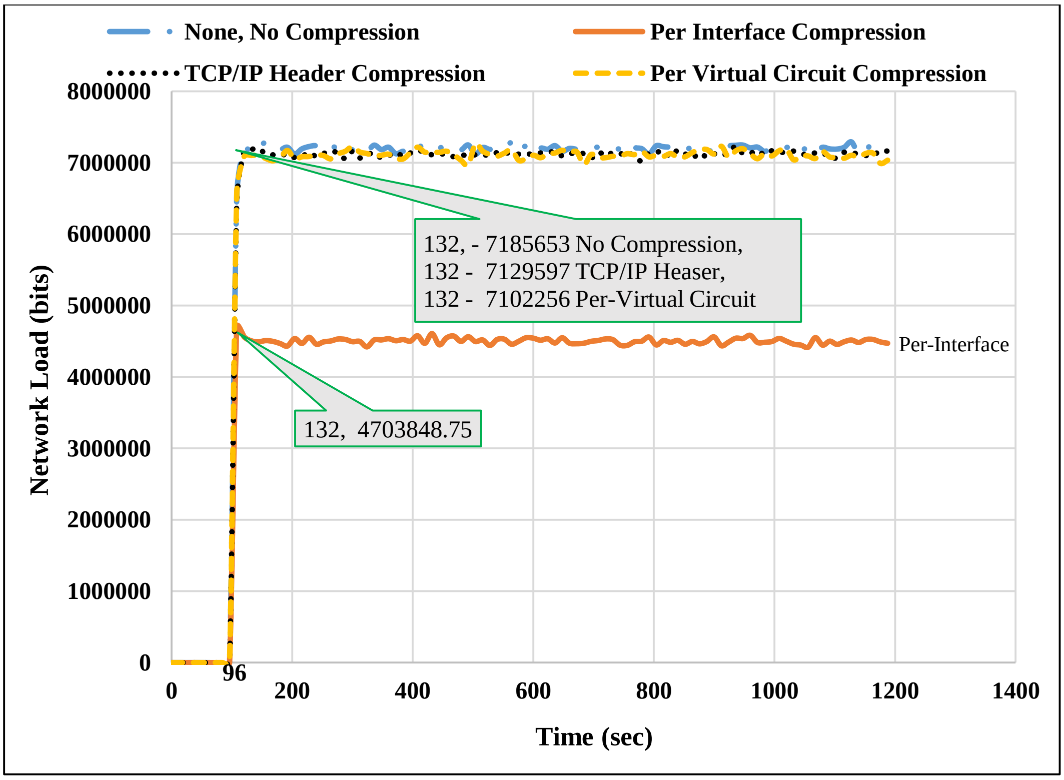

Figure 4. 3: Network Load (bits/sec)

Figure 4.3 shows that all the graphs for compression schemes started to rise when the WLAN connection was established, which took time from 0 to 96 seconds. According to the data, the Per-Interface compression's Network Load rose from 0 to 4,703,848.75 bits while its processing time increased from 96 to 132 seconds. Per-Virtual Circuit Compression's Network Load rose from 0 to 7,102,256 bits, and its processing time increased from 96 to 132 seconds. TCP/IP Header Compression's Network Load rose from 0 to 7,129,597 bits and the time increased from 96 to 132 seconds. The Network Load for None-no compression rose from 0 to 7,185,653 bits and the time increased from 96 to 132 seconds.

In explaining the results in Figure 4.3, we note that Network Load is one of the parameter metrics used to measure performance and Quality of Service of the network:

i) When the graphs heads upwards as the time progresses from 96 seconds to 132 seconds and as more clients/users attempt to access the network simultaneously, the network may experience Network Traffic. The network stress experienced as the graph ascends may be brought about by the lengthy path a packet must take, obstructions like walls, the sheer volume of connected devices and clients, wireless network range, and signal penetration.

Bottlenecks could form when the capacity to manage the current volume of traffic is insufficient. An increase in clients or users is another element that might create a bottleneck in a WLAN. The transmission of data between different components can potentially cause latency (or delay) and increase network traffic. Other issues include slow connections that cannot manage an increase in clients/users, or too many activity, which can cause bandwidth problems.

ii) When the graph becomes generally flat as the time progresses from 96 seconds to 132 seconds, the server is processing all requests without any queue problems, and that there is no latency (or delay), no bottleneck and bandwidth is sufficient.

iii) If there is a little drop in the graph during a steady state, it means that the network capacity is sufficient and the server is processing requests without experiencing any queue errors.

We can therefore, state that the Per-Interface compression scheme had the lowest Network Load of (4,703,848.75 bits/sec which had better network connectivity to achieve high performance resulting to good Quality of Service in WLAN. The Per-Interface compression had the lowest Network Load with 4,703,848.75 bits/sec, while No compression had the highest Network Load with 7,185,653 bits/sec. This is because both the header and the payload had been compressed, resulting in smaller frames and a reduction in total transmission latency, despite the scheme losing time due to compression and decompression processes that consumed a certain amount of time by introducing additional delays by compressing and decompressing at each hop on the route, though response time was faster.

4.1.4 **Traffic Sent**

Traffic Sent (bytes/sec) is the statistic which is defined as the average number of packets per second forwarded to file transport protocol (FTP) applications by the transport layers in the network [77]. The traffic sent for the four (4) compression algorithms under consideration was compared in the study. The results of IP Packet compression schemes derived from OPNET simulation are shown in Figure 4.4. The x-axis (horizontal) shows time in seconds, while the labels on the y-bar (vertical) shows Traffic Sent in bytes per second.

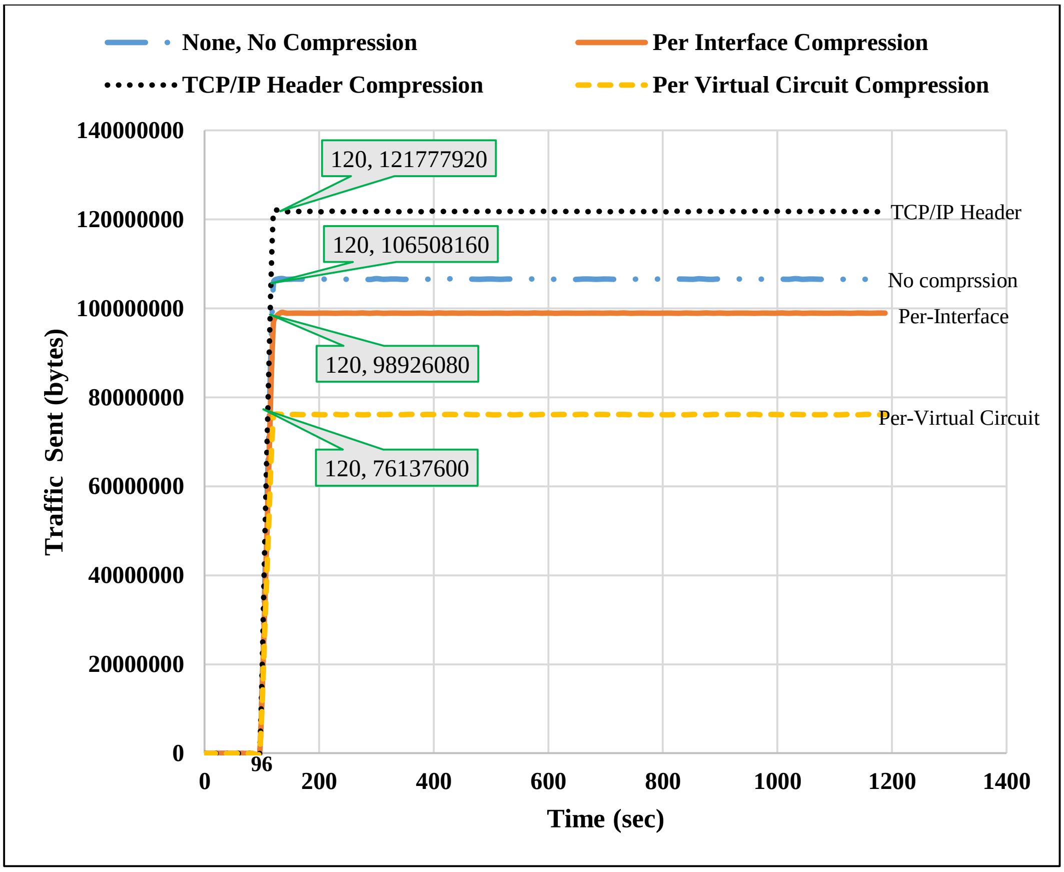

Figure 4. 4: Traffic Sent (bytes/sec)

Figure 4.4 shows that all the graphs for compression schemes started to rise when the WLAN connection was established, which took time from 0 to 96 seconds. According to the data, the Per-Virtual Circuit Compression's Traffic Sent rose from 0 to 76,137,600 bytes while its processing time increased from 96 to 120 seconds. Per-Interface compression's Traffic Sent rose from 0 to 98,926,080 bytes, and its processing time increased from 96 to 120 seconds. None-no Compression's Traffic Sent rose from 0 to 106,508,160 bytes and the time increased from 96 to 120 seconds. The Traffic Sent for TCP/IP Header Compression's rose from 0 to 121,777,920 bytes and the time increased from 96 to 120 seconds.

In explaining the results in Figure 4.4, we note that Traffic Sent is one of the parameter metrics used to measure performance and Quality of Service of the network:

i) When the graphs in figure 4.4 progresses from 96 seconds to 120 seconds and the number of clients/users attempting to access the network at the same time increases, so does the Traffic Sent. When the graph is rising upwards over time and the number of clients/users increases, it indicates that the server is handling requests without any problem and that the network capacity is enough.

ii) If the graph becomes relatively flat over time as the number of clients/users rises, this indicates that bandwidth is getting low or that Traffic Sent has piled up on the network server.

iii) If the graph displays a drop during steady state, it means that the network bandwidth is insufficient or that there is a problem with the network or the problem may be with the server and other devices on the network.

We can therefore, state that TCP/IP header compression had the highest transmitted traffic sent with 121,777,920 bytes/sec and better network connection to achieve high performance, resulting to improved network quality of service. The Per-Virtual circuit compression had the lowest Traffic Sent with 76,137,600 bytes/sec. Furthermore, we observed that compression and decompression occur at each individual node in the TCP/IP header scheme. TCP/IP header compression lowers packet transmission overhead and is advantageous when the application payload is smaller than the header. The scheme also minimises serialisation delay and uses less data, resulting in increased throughput and more available bandwidth. The fixed time required to clock a video/speech or any data frame onto the network interface is what is referred to as serialization delay. The Per-Virtual Circuit Compression technique had the least amount of transmitted traffic.

4.1.5 **Delay (Latency)**

The delay is the amount of time it takes for a packet to be processed in a computer network. This comprises processing, queuing, transmission, and propagation delays [69] [70]. The study compared the delay for the four (4) compression techniques under consideration. Figure 4.5 depicts the results of IP Packet Compression strategies generated from OPNET simulation. The x-axis (horizontal) displays the time in seconds, while the labels on the y-bar (vertical) represent the delay per second.

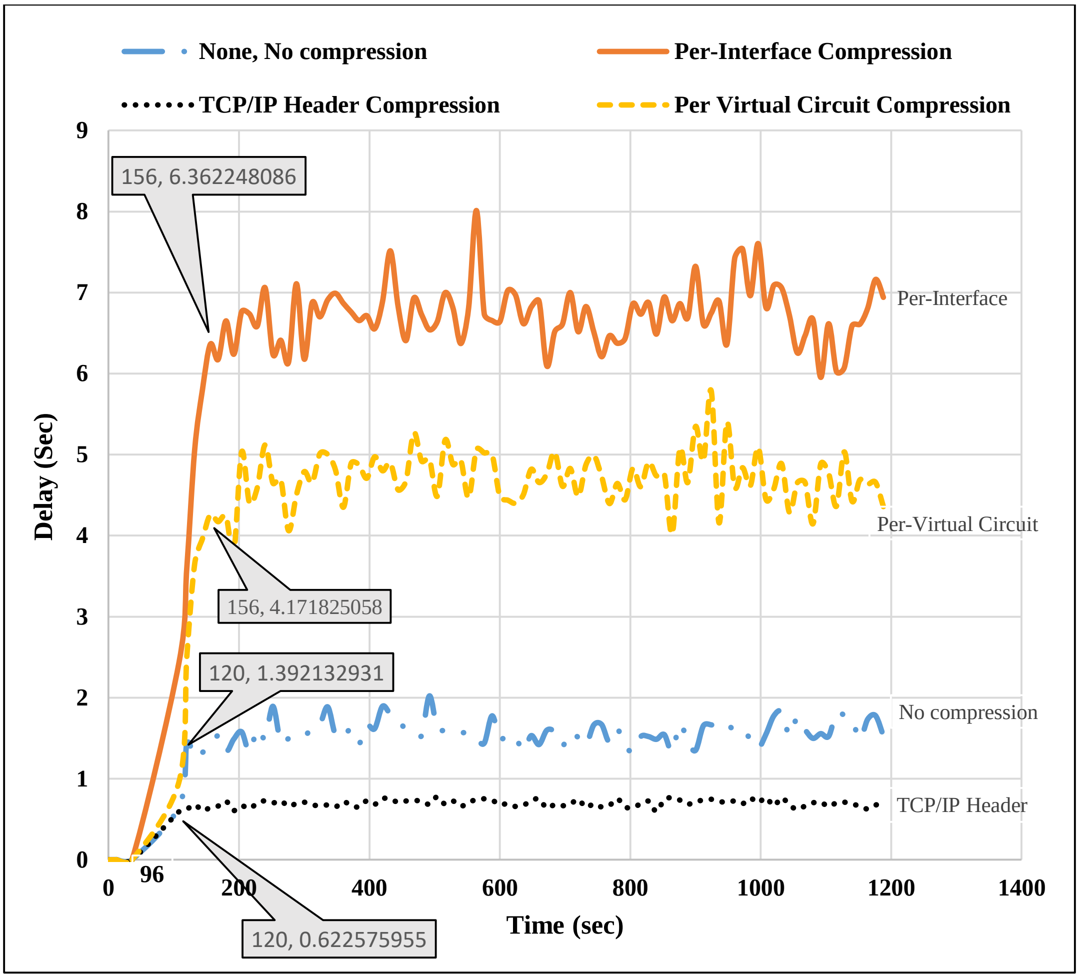

Figure 4. 5: Delay (Sec)

Figure 4.4 shows that all the graphs for compression schemes started to rise when the WLAN connection was established, which took tome from 0 to 96 seconds. According to the data, the TCP/IP Header Compression's Delay rose from 0 to 0.622575955 seconds while its processing time increased from 96 to 120 seconds. None-no Compression's Delay rose from 0 to 1.392132931 seconds, and its processing time increased from 96 to 120 seconds. Per-Virtual Circuit Compression's Delay rose from 0 to 4.171825058 seconds and the time increased from 96 to 120 seconds. The Delay for Per-Interface compression rose from 0 to 6.362248086 second and the time increased from 96 to 120 seconds.

In explaining the results in Figure 4.5, we note that Delay is one of the parameter metrics used to measure performance and Quality of Service of the network:

i) When the graphs in figure 4.5 progresses from 96 seconds to 120 seconds and the number of clients/users attempting to access the network at the same time increases, so does the Delay. When the graph is rising upwards over time and the number of clients/users increases, it indicates that the server is overburdened with requests, or the wireless access point may be having some problems.

Otherwise, the four types of delay (queuing, processing, transmission, and propagation) could also have an impact on the network, including network bandwidth problems caused by slow connections, particularly in WLANs that are not designed to handle an increase in the number of clients/users or the amount of activities.

ii) When the graphs become relatively flat as the time progresses from 96 seconds to 120 seconds, there is adequate bandwidth or the server is processing all requests without any queue errors. If the original cause of Delay was lack of bandwidth due to several clients/users attempting to connect to the server at the same time, now there is sufficient bandwidth.

iii) If a specific drop appears in the graphs during the steady state, it means that network bandwidth is adequate and the server can process all requests without any delays.

We can therefore, state that TCP/IP Header Compression had the lowest Delay with 0.622575955 seconds and improved network connectivity to produce high performance, which leads to good Quality of Service. The Per-Virtual circuit compression had the highest Delay with 6.362248086 second. We also wish to state that under the TCP/IP Header Compression technique, compression and decompression occur at every node. TCP/IP Header Compression decreases packet transmission overhead, lowers packet transmission costs and is effective in circumstances when the application payload size is smaller than the header size. It also minimizes serialization time and uses less bandwidth, resulting in higher throughput and more available bandwidth. The Per-Interface Compression technique had the longest delay and weak network connectivity, making it unable to achieve the much needed WLAN performance and Quality of Service.

4.1.6 **Media Access Delay**

The time it takes for data to reach the Medium Access Control (MAC) layer and be successfully transported out on the network's interface is referred to as Media Access Delay [69] [70]. Media Access Delay is the total amount of time queue and contention delays of data packets it takes for the head of the signal to access a medium [75] [73]. The study compared the Media Access Delay of four (4) compression schemes under investigation. Figure 4.6, shows the results of IP Packet Compression schemes obtained from OPNET simulation. On the x-axis (horizontal) we have time in seconds taken and the labels in the y-bar (vertical) represent Media Access Delay in bits per second.

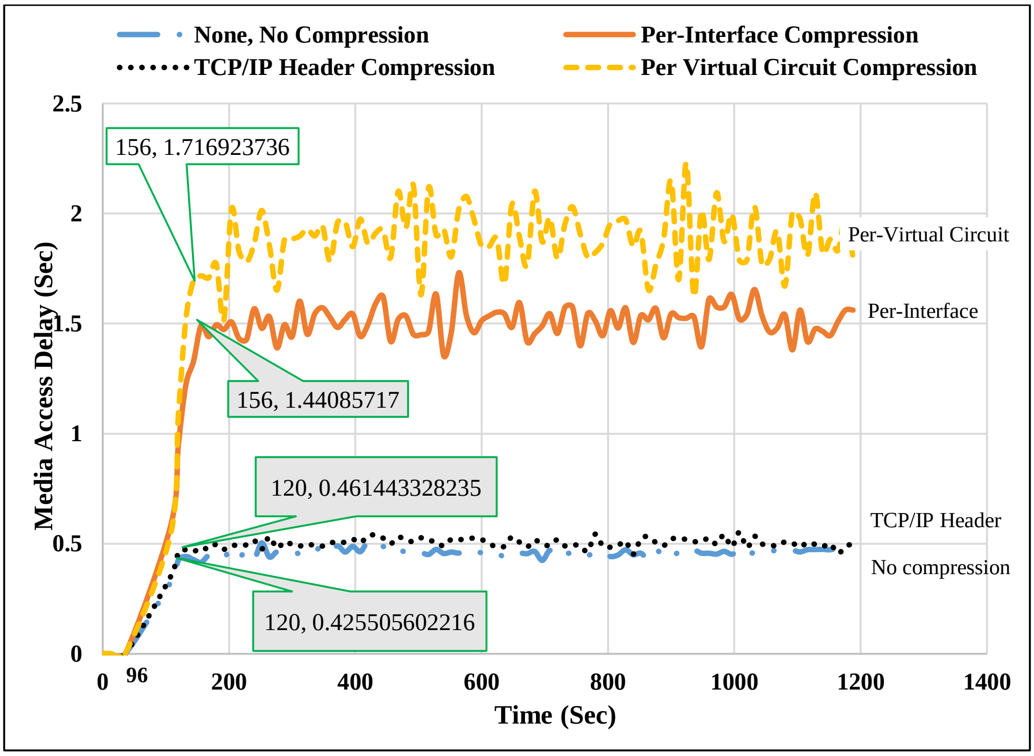

Figure 4. 6: Media Access Delay (Sec)

Figure 4.6 shows that all the graphs for compression schemes started to rise when the WLAN connection was established, which took time from 0 to 96 seconds. According to the data, the None-no compression's Media Access Delay rose from 0 to 0.425505602216 seconds while its processing time increased from 96 to 120 seconds. TCP/IP header compression's Media Access Delay rose from 0 to 0.461443328235 seconds, and its processing time increased from 96 to 120 seconds. Per-Interface compression's Media Access Delay rose from 0 to 1.44085717 seconds and the time increased from 96 to 120 seconds. The Media Access Delay for Per-virtual circuit compression's rose from 0 to 1.716923736 seconds and the time increased from 96 to 156 seconds

In explaining the results in Figure 4.5, we note that Media Access Delay is one of the parameter metrics used to measure performance and Quality of Service of the network:

i) When the graphs in figure 4.5 progresses from 96 seconds to 120/156 seconds and the number of clients/users attempting to access the network at the same time increases, so does the Media Access Delay. When the Media Access Delay graph rises with time and the number of clients/users increases, it indicates that the clients/users are competing for network media access. Other factors influencing the network include network congestion and capacity concerns caused by poor connections, particularly in WLANs that were not built to handle an increase in client/user numbers or activities.

ii) When the graph becomes generally flat over time, from 96 seconds to 120/156 seconds, the media is sufficiently accessible. If the Media Access Delay was initially caused by a bandwidth problem while several clients/users attempted to access the media at the same time, now there is sufficient bandwidth, we would get a media access delay graph flat similar to Figure 4.6.

iii) If there is a drop in the graph during a steady state, it indicates that the Media Access Delay is adequate and the packets may receive the necessary access without any delays.

The None-no Compression algorithm had the shortest Media Access Delay with 0.425505602216 seconds and the best medium access to the network, resulting in a best performance and a high Quality of Service. The Per Virtual Circuit compression scheme had the longest Media Access Delay with 1.716923736 seconds and inadequate medium access to the network, making it difficult to achieve good network performance and Quality of Service.

4.2. **Summary**

In this chapter, the OPNET simulation programme was used to model several compression algorithms. The data from the OPNET simulation programme was exported and analyzed in Microsoft Excel. The major emphasis was on the performance and quality of service (QoS) of IP Packet Compression and Decompression in a WLAN. The study results were documented and compared, among the parameters measured. The results indicate that Per-Interface Compression and TCP/IP Header Compression performed better than other compression algorithms. Per-Virtual Circuit Compression, on the other hand, trailed behind the two techniques, Per-Interface and TCP/IP Header Compression.

CHAPTER FIVE

CONCLUSION, RECOMMENDATIONS AND FUTURE WORK

5.0 Introduction

This chapter presents the conclusion and recommendations that arose from the findings in relation to the study aim and addresses the core research questions to meet the research objectives, contributions, and proposes opportunities for future research. The study evaluated a wide range of secondary literature and conducted compression algorithm in form of scenarios using OPNET simulations tool with well-defined performance metrics as the basis for the study's conclusion and recommendations. The data collected from OPNET simulations is then exported to Microsoft Excel for analysis and creation of graphs. As a result, this chapter was organized as follows: introduction, conclusion, recommendations, future work, and chapter summary.

5.1. Summary of results and findings

The obtained results and findings reveals the following:

i) **Per-Interface Compression** had throughput of 432,925 bits/sec, Data dropped of 371,976,879 bits/sec, Network Load of 4,703,848.75 bits/sec, Traffic Sent of 98,926,080 bytes/sec, Delay of 6.362,248,086 seconds, and Media Access Delay of 1.44085717 seconds.

ii) **TCP/IP Header Compression** had throughput of 2,679,631 bits/sec, a Data dropped of 717,855,026.7 bits/sec, Network Load of 7,129,597 bits/sec, Traffic Sent of 121,777,920 bytes/sec, Delay of 0.622,575,955 seconds, and Media Access Delay of 0.461443328235 seconds.

iii) **Per-Virtual Circuit** had Compression had throughput of 1,873,536 bits/sec, Data dropped of 534,815,948.2 bits/sec, Network Load of 7102256 bits/sec, Traffic Sent of 76,137,600 bytes/sec, Delay of 4.171,825,058 seconds, and Media Access Delay of 1.716,923,736 seconds.

iv) **No Compression** had throughput of 4,031,784 bits/sec, Data dropped of 782,894,165 bits/sec, Network Load of 7,185,653 bits/sec, Traffic Sent of 106508160 bytes/sec, Delay of 1.392,132,931 seconds, and Media Access Delay of 0.425,505,602,216.

5.2. Conclusion

The obtained results and findings revealed that among all the compression schemes, Per-Interface Compression had the lowest performance indicator for throughput (432,925 bits/sec), Data dropped (717,855,026.7 bits/sec) and Network Load (4,703,848.75 bits/sec). Whereas TCP/IP Header Compression had higher performance indicator for Traffic Sent (121,777,920 bytes/sec) and lowest performance indicator for Delay (0.622,575,955 seconds). This was determined using performance metrics for a particular situation, that include Throughput, Data Dropped, Delay, Network Load, Traffic Sent, and Media Access Delay. Per-Virtual Circuit Compression had performance indicators for throughput (1,873,536 bits/sec), Data dropped (534,815,948.2 bits/sec), Network Load (7102256 bits/sec), Traffic Sent (76,137,600 bytes/sec), Delay (4.171,825,058 seconds), and Media Access Delay (1.716,923,736 seconds). The scheme trailed behind Per-Interface and TCP/IP Header Compression schemes. The None-no compression scheme was merely utilized for comparison reasons. The Per-Interface Compression technique had the best performance and QoS in a WLAN.

5.2.1. Per-Interface Compression

In most cases, Per-Interface Compression algorithm had the lowest performance indicator for throughput (432,925 bits/sec), Data dropped (717,855,026.7 bits/sec) and Network Load (4,703,848.75 bits/sec). The related work which was revealed in the research indicate that little was done about the performance of Per-Interface algorithm, but the results of this study shows that Per-Interface compression algorithm resulted in smaller packets created, despite the fact that the scheme introduces additional delays by compressing and decompressing at each hop on the route. Per-interface compression is therefore, a better solution for WLAN with fewer hops between end nodes.

5.2.2. **TCP/IP Header Compression**

The study's results and findings shows that TCP/IP Header Compression had higher performance indicator for Traffic Sent (121,777,920 bytes/sec) and lowest performance indicator for Delay (0.622,575,955 seconds). TCP/IP Header compression decreases packet transmission overhead, lowers packet transmission costs and is effective in circumstances when the application payload size is smaller than the header size. The technique also minimizes serialization time and bandwidth requirements, resulting in increased throughput. According to the relevant studies evaluated in the study [57] [48] [51] [53], header compression decreases overhead to increase throughput, which is effective for limited bandwidth networks where throughput is a top consideration. This study confirms similar results that header compression reduces overhead to improve throughput which is most useful on low bandwidth links.

5.2.3. **Per-Virtual Circuit Compression**

Based on the indicated results and findings in terms of performance and quality of service, Per-Virtual Circuit Compression had performance indicators for throughput (1,873,536 bits/sec), Data dropped (534,815,948.2 bits/sec), Network Load (7102256 bits/sec), Traffic Sent (76,137,600 bytes/sec), Delay (4.171,825,058 seconds), and Media Access Delay (1.716,923,736 seconds). The scheme trailed behind Per-Interface and TCP/IP Header Compression in the simulations that were undertaken. Compression and decompression for this technique happen only at the end nodes. The scheme is considered inappropriate for use in a WLAN platform as evidenced by the findings. According to related work [57] [48] [51] [53], per-virtual circuit compression should be used for packets that have header size smaller than payload owing to the processing delay time, which can rise as the size of the packet increases [57]. This research verifies previous findings that when processing time exceeds transmission time, the compression technique begins to reduce bandwidth.

5.2.4. **Study Contribution to the Body of Knowledge**

Generally, the study achieved its objectives and has contributed to the body of knowledge regarding IP Packet Compression and decompression from a WLAN standpoint. First, using OPNET simulations, this work investigated the four (4) compression algorithms in a WLAN. Second, this study extends the conclusion that Per-Interface Compression performed better than other compression algorithms investigated in this study. Thirdly, network designers should use TCP/IP Header Compression when the header size is bigger than the payload because the improvement is dramatic, and network designers should use Per-Virtual Circuit Compression when the header size is smaller than the payload because the improvement is significant. The researcher believes that this study will assist other academicians and ICT professionals improve network performance and quality of service in WLAN by varying compression schemes on the network platform to ensure that they implement the best compression scheme.

5.3. **Recommendations**

The purpose of this study was to evaluate the effectiveness and efficient of compression and decompression of IP Packets in a WLAN. None-no compression, per-interface compression, TCP/IP header compression, and per-virtual circuit compression received favourable attention in this study. The study recommends that compression algorithm variation should be considered when designing and developing a network, as it may possibly produce superior results without incurring substantial expenses when comparing one compression strategy to another. The following recommendations are made to improve on the identified problems in the study:

5.3.1. **Per-Interface Compression**

Based on our study, we advise that network designers and developers who implement the Per-Interface Compression model should consider testing the scheme to determine how well their network performs under various conditions and situations. It should be noted that vast networks, even if they are distinct or different in most circumstances, can be used as a baseline.

5.3.2. **TCT/IP Header Compression**

According to the study's findings and conclusion, network designers and developers should avoid using the TCP/IP Header Compression technique when the Header size is smaller than the payload since the benefits are negligible. Due to the huge gain, the TCP/IP Header Compression technique is better utilized when the Header size is bigger than the payload size. We highly advise simulating or testing TCP/IP Header Compression and Per-Interface compression to determine which one is optimal for the network architecture being deployed.

5.3.3. **Per-Virtual Circuit Compression**

Based on the study findings and conclusions, network designers and developers should use the Per-Virtual Circuit Compression mechanism when the header size is smaller than the payload size since the benefits are enormous. When the Header size is bigger than the payload size, it is worthwhile to consider employing the TCP/IP Header Compression approach. To identify which strategy is appropriate for the existing network architecture, we recommend simulating or testing both Per-Virtual Circuit Compression and TCP/IP Header Compression to determine which one works better.

5.4. **Future Research/Work**

The work given in this study lays the groundwork, yet there is still potential for improvement. The report addressed specific concerns and made recommendations. What remains is for these recommendations to be implemented. As a result, it is evident that similar or more study should be conducted to explore more simulation situations comparable to a larger network real-life scenario to determine if the same findings could be established.

5.5. **Summary**

We considered the conclusion based on the research results and findings on the Evaluation of the Effectiveness and Efficiency of Compression and Decompression of IP Packets in a Wireless Local Area Network (WLAN) that were provided and major key research questions have been answered. The benefits of undertaking the study have been presented based on the findings from the simulations undertaken and the analysis. The chapter also included ideas and future work that might help to enhance the compression schemes, which did not perform well in this study.

References

[1] I. M. Omar and S. Prasad, "A review of data compression techniques," *International Journal of Advance Research in Science and Engineering,* vol. Volume Number 07, no. Special issue n. 07, pp. 1-8, 2018.

[2] G. Goetz and L. D. Shapiro, "Data Compression and Database Performance," *ACM/IEEE-CS Symp. on Applied Computing,* p. 10, 1990.

[3] A. S. Sethi and V. Y. Hnatyshin, "The Practical OPNET® User Guide for Computer Network Simulation," CRC Press Taylor & Francis Group, London New York, 2013.

[4] K. Anubhuti, M. Saxena and B. Patil, "Voice Data Compression and Decompression," in *Cost Electronics & Communication Conference Proceedings,* 2011.

[5] K. Sayood, Introduction to Data Compression, San Francisco, San Francisco: Elsevier Inc., 2006, pp. 1-5.

[6] I. Pu, "Data compression," *Goldsmith: University of London - International Programmes,* 2004.

[7] B. Patil, "Voice Data Compression and Decompression," 2016.

[8] F. Shamieh, "Advanced Compression and Latency Reduction Techniques Over Data Networks," The University of Western Ontario, Western Ontario, 2015.

[9] P.Kavitha, "A Survey on Lossless and Lossy Data," *International Journal of Computer Science & Engineering Technology (IJCSET),* p. 5, 2016.

[10] G. Tano, "Transparent Data Compression and Decompression for Embedded Systems," *Department of Information Technology,* p. 42, April 2008.

[11] W. Lawson, Building Cisco Remote Access Network, Rockland,: Syngress Publishing, Inc., 2000.

[12] M. Yossi and R. Refua, "Delayed-Dictionary Compression for Packet Networks," *Tel-Aviv University, Israel,* pp. 3-5, 2013.

[13] G. Cenk, P. Kietzmann and T. C. Schmidt, "ICNLoWPAN – Named-Data Networking for Low Power IoT Networks," *IFIP Networking 2019 1570512314,* p. 3, 2019.

[14] G Liwei; D Jan and A Anne, "Compression Performance Comparison of x264, x265, libvpx and aomenc for On-Demand Adaptive Streaming Applications," *IEEE,* pp. 1-5, 2018.

[15] O. Mugenda and A. Mugenda, Research Methods (Quantitative and Qualitative Approach), Nairobi: Africa Centre for Technology Studies (ACTS) Nairobi Kenya, 2003.

[16] R. Kumar, Research Methods: Step by step guide for beginners, Washington: SAG E Publication Inc, Asia Pacific Pte Ltd, 2022.

[17] S. M. Tichapondwa, Preparing your Dissertation at a Distance: A Research Guide, Virtual University for Small States, 2013, pp. 38-39.

[18] B. R. Stojkoska and Z. Nikolovski, "Data Compression for Energy Efficient IoT Solutions," *25th Telecommunications forum (TELFOR),* pp. 1-4, 2017.

[19] D. W. Davies, "A history Study of the beggings of Packet Switching," *The Computer Journal,* Vols. Volume 44, Issue 3, no. Issue 3, pp. 1-2, 2001.

[20] I. M. Pu, Fundamental Data Compression, London: British Library Cataloguing in Publication Data, 2006, pp. 24-25.

[21] C. S. Lubobya, M. E. Dlodlo and G. d. Jager, “Performance Comparisons of Wireless Mesh IP Video Surveillance Models,” *IEEE European Modelling Symposium,* p. 3, 2015.

[22] D. Mahanta, M. Ahmed, and U. J. Bora, 2013. A study of bandwidth management in computer networks. International Journal of Innovative Technology and Exploring Engineering, 2(2), pp.69-73.

[23] S. David and G. Motta, Handbook of Data Compression, London: British Library Cataloguing in Publication Data, 2010, pp. 22-23.

[24] V. Britanak, K. R. Rao and P. C. Yip, The Transform and Data Compression Handbook, Boca Raton London New York Washington, D.C.: CRC Press, 2001 and 2017, p. 182.

[25] M. Wilson, K. Wnuk, J. Silvander and a. T. Gorschek, “A Literature Review on the Effectiveness and Efficiency of Business Modeling,” *Wilson, Magnus, Krzysztof Wnuk, Johan Silvander, and Tony Gorschek. "A literature revieE-Informatica Software Engineering Journal,* vol. Volume 12, no. Issue 1, p. 38, 2018.

[26] G. Apoorv, A. Bansal and V. Khanduja, “Modern Lossless Compression Techniques: Review, Comparison and Analysis,” *Department of Computer Engineering, Netaji Subhas Institute of Technology, Delhi, India,* pp. 1-8, 2017.

[27] R. Ori, “Lossy Compression of Packet Classifiers,” *IEEE,* pp. 1-12, 2015.

[28] S. Canbaz, and G. Erdemir, 2022. Performance analysis of real-time and general-purpose operating systems for path planning of the multi-robot systems. International Journal of Electrical and Computer Engineering, 12(1), p.285.

[29] D. M. Havey, “Throughput and Delay on the Packet Switched Internet,” University of California, Santa Barbara, 2015.

[30] S. A. Jyothi, A. Singla, P. G. Brighten and A. Kolla, “Measuring and Understanding Throughput of Network Topologies,” *IEEE,* pp. 1-12, November 2016.

[31] K. Sayood, “Data Compression in Remote Sensing Applications,” *Department of Electrical Engineering and Centre for Communication and Information Science,* p. 2, 1993.

[32] K. Sayood, Introduction to Data Compression, 50 Hampshire Street, 5th Floor, Cambridge, MA 02139: Morgan Kaufmann is an imprint of Elsevier, 2017, p. 770.

[33] S. Walfram, A new Kind of Science, Trade Center Drive, Champaign, IL 61820, USA: Wolfram Media, Inc., 2002, p. 1280.

[34] D. Salomon, Data compression (4th Edn), London, Great Britain: British Library Cataloguing in Publication Data, 2007, pp. 48-50.

[35] T. Berger, “Rate-Distortion Theory,” *John Wiley & Son,* pp. 1-3, 2003.

[36] W. Wang, “A Distributed Covert Channel of the Packet Ordering Enhancement Model Based on Data Compression,” *Researcgate: Computers, Materials and Continua,* Vols. CMC, vol.64, no.3,, pp. 4-5, June, 2020.

[37] L. A. Fitriya, “A Review of Data Compression Techniques,” *International Journal of Applied Engineering Research ISSN 0973-4562,* Vols. Volume 12, Number 19, p. 8, 2017.

[38] R. K. Kurmi and S. Gupta, “A Review of Lossless and Lossy Based Image Compression Techniques,” *International Journal of Engineering and Advanced Technology,* Vols. Volume-6 Issue-6, August 2017, no. ISSN: 2249 – 895817, p. 5, August 2017.

[39] N. A. Azeez and A. A. Lasisi, “Empirical and Statistical Evaluation of the Effectiveness of Four Lossless Data Compression Algorithms,” *Nigerian Journal of Technoloical Development,* Vols. VOL. 13, NO. 2,, pp. 1-2, 2 December 2016.

[40] P. Koskela, M. Majanen and M. Valta, “Packet Header Compression for the Internet of Things,” *IFSA Publishing,* Vols. Vol. 196,, no. Issue 1,, pp. 2-3, 2016.

[41] Z. Mehdi and J. S. Thompson., “Impact of Compression and Aggregation in Wireless Networks on Smart Meter Data,” *Digital Communications Dept., The University of Edinburgh,,* p. 2, 2016.

[42] K. Sayood, Introduction to Data Compression, 50 Hampshire Street, 5th Floor, Cambridge, MA 02139, United States: Morgan Kaufmann is an imprint of Elsevier, 2017, pp. 17 - 18.

[43] S. Sharma, “Analysis of Header Compression Techniques for Networks: A Review,” *Researchgate.net,* pp. 1-9, 2013.

[44] L. L. Peterson and B. S. Davie, Computer Networks a systems approach, 50 Hampshire Street, 5th Floor, Cambridge, MA 02139, United States: Morgan Kaufmann is an imprint of Elsevier, 2016, pp. 200 - 320.

[45] J. F. Kurose and K. Ross W, Computer Network: A top down approach 8th Edition, Pearson education, Inc, 2017.

[46] A. Shiranzaei and Z. B. Hamid, “A Comparative Study between IPv4 and IPv6,” *International Conference on Engineering, Technology & Vocational Education (ICETVE),* pp. 1-8, 07 November 2020.

[47] S. Isaac, Comparative Analysis of IPV4 and IPV6, vol. Vol. 7 (2), International Journal of Computer Science and Information Technologies,, 2016.

[48] T. Máté, P. Seeling, P. Ekler and F. H. Fitzek, “Performance Evaluation of Network Header Compression Schemes for UDP, RTP and TCP,” *research article,* pp. 1-2, 16 March 2016.

[49] S. Jianan, P. Dong, Y. Qin, T. Zheng, X. Yan and Y. Zhang, “Improving bandwidth utilization by compressing small-payload traffic for vehicular networks,” *International Journal of Distributed Sensor Networks,* vol. Vol. 15(4), pp. 1-14, 2019.

[50] Y. Niu, C. Wu, L. Wei, B. Liu and J. Cai, “Backfill: An Efficient Header Compression Scheme for OpenFlow Network with Satellite Links,” *International Conference on Networking and Network Applications,* pp. 1-4, 2016.

[51] T. Mate, D. Lucani, F. H. Fitzek and P. Ekler, “Unidirectional Robust Header Compression for Reliable Low Latency Mesh Networks,” *Department of Engineering, Communications Systems, Aarhus University, Denmark,* pp. 1-6, 2019.

[52] C. Feres, "Low Complexity Header Compression with Lower-Layer Awareness for Wireless Networks," *International Conference on Communications,* pp. 1-7, 2019.

[53] M. Tomoskozi, D. Lucani, F. H. P. Fitzek and P. Ekler, "Reliable Base Proposal for Header Compression," *EEE 90th Vehicular Technology Conference (VTC2019-Fall),* pp. 1-5, 2019.

[54] H. Cha, T. Shon, K. Kim and M. Hong, "Improving packet header compression with adaptive sliding window size," *Cha, Hyejin; Shon, Taeshik; Kim, Kangseok; Hong, Manpyo (2015). [IEEE 2015 International Conference on Information Networking (ICOIN) - CambInternational Conference on Information Networking (ICOIN),* 12-14 Jan. 2015.

[55] B. Hung, D. Defrancesco and B.-N. &. S. P. Cheng, "An Evaluation of IP Header Compression on the GIG Joint IP Modem System," *Hung, Brian; Defrancesco, Dana; Cheng, Bow-Nan; Sukumar, Prasanna (2014). [IEEE 2014 IEEE Military Communications Conference (MILCOM) - BaltiMilitary Communications Conference,* pp. 1-7, 2014.

[56] D. Chauhan and J. K. Jain and S. S. Jain, "An End-to-End Header Compression for Multihop IPv6 tunnels with varying Bandwidth," pp. 1-5, 2016.

[57] R. Measel, "IPv6 Compression Techniques and Perfomance (Doctoral dissertation, Drexel University)," 2011.

[58] C. S. Lubobya, M. E. Dlodlo, G. de Jager, and K. L. Ferguson, 2011, September. SIMD implementation of integer DCT and hadamard transforms in H. 264/AVC encoder. In IEEE Africon'11 (pp. 1-5). IEEE.

[59] C. R. Kothari, Research Methodology, methods and Techniques, New Delhi, New Delhi: New Age International (P) Ltd, 2019, pp. 7-8.

[60] A. M. Novikov and D. A. Novikov, Research Methodology, From Philosophy of Science to Research Design, 6000 Broken Sound Parkway NW, Suite 300, United States of America: Taylor & Francis Group, LLC, 2013, p. 3.

[61] R. Kumar, Research Methodology, a step by step guide for beginners, London: SAGE Publications Ltd, 2011, p. 18.

[62] P. D. Leedy and J. E. Ormrod, "Practical Research; Planning and Design (8th ed)," 2005.

[63] M. Vinita and S. Jangale, "Analysis and comparison of different network simulators," *International Journal of Application or Innovation in Engineering & Management (IJAIEM),* pp. 1-5, 2014.

[64] K. M. Humayun, S. Islam, M. J. Hossain and S. Hossain, "Detail Comparison of Network Simulators," *International Journal of Scientific & Engineering Research,* vol. 5, no. 10, pp. 1-16, 2014.

[65] S. Bilalb and M. Othmana, "A Performance Comparison of Network Simulators for Wireless Networks," *Faculty of Computer Science & Information Technology, University of Malaya, LP 50603, Kuala Lumpur, Malaysia.,* vol. arXiv:1307.4129, pp. 1-6, 2013.

[66] M. S. Shegokar and V. R. Tripathi, "Analysis Study and Comparison of Recent Trends in Network Simulator," *JETIR,* vol. Volume 6, no. Issue 4, pp. 1-7, 2019.

[67] M. A. Kaushik and M. Shekhar, "Network Simulators for next Generation Network: An Overview," *International Journal of Mobile Network Communications & Telematics (IJMNCT),* Vols. Vol. 4, No., pp. 1-13, 2014.

[68] Z. Ani and I. Alsmadi, “Evaluating network test scenarios for network simulators systems,” *International Journal of Distributed Sensor Networks,* vol. Vol. 13(10), pp. 1-17, 2017.

[69] R. Patel, M. Pathak and A. Nayak, “Survey on Network Simulators,” *International Journal of Computer Applications (0975 – 8887),* Vols. Volume 182 – No. 21, October 2018, pp. 1-8, 21 October 2018.

[70] Z. Lu, and H. Yang, 2012. Unlocking the power of OPNET modeler. Cambridge University Press..

[71] M. M. A Ghazala, M. F. Zaghloul, and M Zahra, 2009. Performance evaluation of multimedia streams over Wireless Computer Networks (WLANs). International Journal of Advanced Science and Technology, 13, pp.63-76.

[72] N. S. Chauhan, and L. Kaur, 2013. Implementation of QoS of different multimedia applications in WLAN. International Journal of Computer Applications, 62(8).

[73] T. Nimi, and P. Samundiswary, 2018, February. Comparative Performance evaluation on Priority based ZigBee Network with tree and mesh routing. In 2018 4th International Conference on Electrical Energy Systems (ICEES) (pp. 691-695). IEEE.

[74] L. Zhang, J. Xiao and P. Wang and P. X. Wang, “Implementation of QoS of different multimedia applications in WLAN.,” *International Journal of Computer Application,* Vols. Volume 62– No.8,, pp. 1-5, 2013.

[75] A. Bagwari, R. Jee and P. Joshi and B. S. Joshi, “Design and implementation of communication network for modular microgrid based power park,” *IEEE Innovative Smart Grid Technologies,* pp. 1-4, 2016.

[76] T. Nimi, and P. Samundiswary, 2018, February. Comparative Performance evaluation on Priority based ZigBee Network with tree and mesh routing. In 2018 4th International Conference on Electrical Energy Systems (ICEES) (pp. 691-695). IEEE.

[77] R. Madan, and P. S. Mangipudi, 2018, August. Predicting computer network traffic: a time series forecasting approach using DWT, ARIMA and RNN. In 2018 Eleventh International Conference on Contemporary Computing (IC3) (pp. 1-5). IEEE.

[78] A. Bagwari, R. Jee, P. Joshi, and S. Bisht, 2012, May. Performance of AODV routing protocol with increasing the MANET nodes and its effects on QoS of mobile ad hoc networks. In 2012 International Conference on Communication Systems and Network Technologies (pp. 320-324). IEEE.

[79] A. A. Atayero, A. S. Alatishe, and I. O. Iruemi, 2012. Modeling and Simulation of a University LAN in OPNET Modeller Environment. International Journal of Emerging Technology and Advanced Engineering, 2(1), pp.1-4.

APPENDIX A: CERTIFICATE OF PUBLICATION

Journal of Wireless Networking and Communications

Certificate of Publication

Date: November 16, 2022

Dear Colleague,

Thank you very much for your contribution to Scientific & Academic Publishing.

We are pleased to inform you that your paper has been published online.

Paper ID:	100300172
Paper Title:	Performance Evaluation and Compression of IP Packets in a Wireless Local Area Network (WLAN)
Authored by	Bernard Sentala, Charles S. Lubobya, Ackim Zulu

It is published in ***Journal of Wireless Networking and Communications (Volume 11, Number 1, 2022).***

We look forward to your continued support.

Thank you again.

Sincerely yours,

Charles Duke

Journal manager

Scientific & Academic Publishing (http://www.sapub.org/), USA

SCIENTIFIC & ACADEMIC PUBLISHING CO. LTD CORPORATE SEAL USA CALIFORNIA

APPENDIX B: PUBLISHED JOURNAL/ARTICLE

Journal of Wireless Networking and Communications 2022, 11(1): 1-10
DOI: 10.5923/j.jwnc.20221101.01

Performance Evaluation and Compression of IP Packets in a Wireless Local Area Network (WLAN)

Bernard Sentala*, Charles S. Lubobya, Ackim Zulu

Department of Electrical Engineering, University of Zambia, School of Engineering, Lusaka, Zambia

Abstract The study focuses on the Performance Evaluation and Compression of IP Packets in a Wireless Local Area Network (WLAN). WLAN has many network challenges such as network delay, buffering requirement, bandwidth, and network congestion. The aim of the study is to design and develop a WLAN model that would assist test the Performance Evaluation and Compression of IP Packets in a WLAN. The key specific objectives are to assess the effectiveness and efficiency of per-interface compression in a WLAN during IP packet transmission; to analyze the effects of TCP/IP header compression and per-virtual circuit compression on performance and/or quality of service (QoS) in a WLAN, and to design and develop the best compression scheme in transmitting IP Packets in a WLAN. OPNET simulation software tool has been used to simulate the four (4) IP Packet Compression Schemes that include; No compression, Per-interface Compression, TCP/IP Header Compression, and Per-virtual Circuit Compression. The results indicate that Per-virtual Circuit Compression Scheme performed below Per-Interface and TCP/IP Header Compression Schemes. No Compression Scheme is utilized solely for comparative reasons and the scheme recorded the lowest score of all the compression schemes. In terms of overall performance, the Per-Interface Compression algorithm excelled over all other compression schemes tested in the study.

Keywords Compression, OPNET, WLAN, Efficiency, and performance

1. Introduction

In general, IP packet compression is the process that reduces the packet size or file size of digital signals without changing the amount of digital file signals during data transmission without affecting the signal amount of digital files in a Wireless Local Area Network (WLAN) [1] [2]. To simulate and come up with a better solution, the study employed Optimized Network Engineering Tool (OPNET) software for computer network simulation, modeling, and analysis. This was done in order to create optimized and/or mitigated IP Packet delivery challenges, the study aim was to design and develop a model that would help in testing the Performance Evaluation and Compression of IP Packets in a Wireless Local Area Network (WLAN) using four (4) compression schemes while taking performance and Quality of Service into consideration. The key specific objectives were: to assess the effectiveness and efficiency of the per-interface compression in a WLAN during IP packet transmission; to analyze the effects of TCP/IP header and per-virtual circuit compression on performance and/or quality of service (QoS) in a WLAN, and to design and develop the best compression scheme in transmitting IP Packets in a WLAN. The network problems such as Queuing, transmission, propagation, and processing delays, including buffering needs and network congestion of IP packets, could be reduced and/ or mitigated when the available resources are employed as efficiently and effectively as possible [1] [2] [3] [4].

* Corresponding author:
bsentala@gmail.com (Bernard Sentala)
Received: Oct. 13, 2022; Accepted: Oct. 24, 2022; Published: Nov. 14, 2022
Published online at http://journal.sapub.org/jwnc

The study focused on the Performance Evaluation and Compression of IP Packets in a Wireless Local Area Network (WLAN) as a strategy to optimize and reduce packet network delay problems in connection with IP Packet delivery [2]. Data compression is a coding system that allows characters to be eliminated from data frames on the sending side of a transmission device and appropriately substituted on the receiving side [3]. Compression is classified into two (2) major categories: lossy and lossless. Lossy compression occurs when data is compressed to reduce the file size by permanently deleting unnecessary information, such as a picture, and image quality is lowered to a minimum. A lossless compression system is one in which X and Y are identical, whereas a lossy compression technique gives substantially better compression than lossless compression but enables Y to diverge from X [3] [4] [5].

The study found that the Per-Virtual Circuit Compression technique continuously trailed behind the Per-Interface Compression and TCP/IP header compression schemes. None, no compression scheme is used merely for comparison purposes, and this scheme obtained the lowest

APPENDIX C: ETHICS APPROVAL LETTER

THE UNIVERSITY OF ZAMBIA

DIRECTORATE OF RESEARCH AND GRADUATE STUDIES

Great East Road Campus | P.O. Box 32379 | Lusaka 10101 | Tel: +260-290 258/291 777
Fax: (+260) 211 290 258/253 952 | Email: director.drgs@unza.zm | Website: www.unza.zm

APPROVAL OF STUDY

IORG No. 0005376
HSSREC IRB No. 00006465

30th May, 2022

REF NO. NASREC-2022-MAY-001

Mr. Bernard Sentala
The University of Zambia
School of Engineering
Department of Electrical Engineering
<u>LUSAKA</u>

Dear Mr. Sentala

RE: "EVALUATION OF THE EFFECTIVENESS AND EFFICIENCY OF THE IP PACKET COMPRESSION AND DECOMPRESSION IN A WIRELESS LOCAL AREA NETWORK (WLAN)"

Reference is made to your protocol dated as captioned above.

NASREC resolved to approve this study and your participation as Principal Investigator for a period of one year.

REVIEW TYPE	ORDINARY REVIEW	APPROVAL NO. NASREC-2022- MAY-001
Approval and Expiry Date	Approval Date: 30th May, 2022	Expiry Date: 29th May, 2023
Protocol Version and Date	Version - Nil.	30th May, 2022
Information Sheet, Consent Forms and Dates	• English.	To be provided
Consent form ID and Date	• Version - Nil	To be provided
Recruitment Materials	• Nil	Nil
Other Study Documents	• Questionnaire.	

Towards Improving Service and Excellence in High Education Beyond Fifty Years

Specific conditions will apply to this approval;

As Principal Investigator it is your responsibility to ensure that the contents of this letter are adhered to. If these are not adhered to, the approval may be suspended. Should the study be suspended, study sponsors and other regulatory authorities will be informed.

Conditions of Approval

- No participant may be involved in any study procedure prior to the study approval or after the expiration date.
- All unanticipated or Serious Adverse Events (SAEs) must be reported to NASREC within 5 days.
- All protocol modifications must be approved by NASREC prior to implementation unless they are intended to reduce risk (but must still be reported for approval). Modifications will include any change of investigator/s or site address.
- All protocol deviations must be reported to NASREC within 5 working days.
- All recruitment materials must be approved by NASREC prior to being used.
- Principal investigators are responsible for initiating Continuing Review proceedings. NASREC will only approve a study for a period of 12 months.
- It is the responsibility of the PI to renew his/her ethics approval through a renewal application to NASREC.
- Where the PI desires to extend the study after expiry of the study period, documents for study extension must be received by NASREC at least 30 days before the expiry date. This is for the purpose of facilitating the review process. Documents received within 30 days after expiry will be labelled "late submissions" and will incur a penalty fee of K500.00. No study shall be renewed whose documents are submitted for renewal 30 days after expiry of the certificate.
- Every 6 (six) months a progress report form supplied by The University of Zambia Natural and Applied Sciences Research Ethics Committee as an IRB must be filled in and submitted to us. There is a penalty of K500.00 for failure to submit the report.
- When closing a project, the PI is responsible for notifying, in writing or using the Research Ethics and Management Online (REMO),both NASREC
- and the National Health Research Authority (NHRA) when ethics certification is no longer required for a project.
- In order to close an approved study, a Closing Report must be submitted in writing or through the REMO system. A Closing Report should be filed when data collection has ended and the study team will no longer be using human participants or animals or secondary data or have any direct or indirect contact with the research participants or animals for the study.
- Filing a closing report (rather than just letting your approval lapse) is important as it assists NASREC in efficiently tracking and reporting on projects. Note that some funding agencies and sponsors require a notice of closure from the IRB which had approved the study and can only be generated after the Closing Report has been filed.
- A reprint of this letter shall be done at a fee.

2

- All protocol modifications must be approved by NASREC by way of an application for an amendment prior to implementation unless they are intended to reduce risk (but must still be reported for approval). Modifications will include any change of investigator's or site address or methodology and methods. Many modifications entail minimal risk adjustments to a protocol and/or consent form and can be made on an Expedited basis (via the IRB Chair). Some examples are: format changes, correcting spelling errors, adding key personnel, minor changes to questionnaires, recruiting and changes, and so forth. Other, more substantive changes, especially those that may alter the risk-benefit ratio, may require Full Board review. In all cases, except where noted above regarding subject safety, any changes to any protocol document or procedure must first be approved by NASREC before they can be implemented.

Should you have any questions regarding anything indicated in this letter, please do not hesitate to get in touch with us at the above indicated address.

On behalf of NASREC, we would like to wish you all the success as you carry out your study.

Yours faithfully,

Dr. M. Kaonda

VICECHAIRPERSON
THE UNIVERSITY OF ZAMBIA NATURAL AND APPLIED SCIENCES RESEARCH ETHICS COMMITTEE - IRB

cc: Director, Directorate of Research and Graduate Studies
Assistant Director (Research), Directorate of Research and Graduate Studies
Assistant Registrar (Research), Directorate of Research and Graduate Studies
Acting Senior Administrative Officer (Research), Directorate of Research and Graduate Studies