Tshwane University of Technology

We empower people

Enhanced Bandwidth Allocation Model for Improving Quality of Service in Intelligent Home Networks

By

THEMBA LUCKY NKOSI

(206314460)

Submitted in partial fulfillment of the requirements for the degree

MASTERS IN COMPUTING: INFORMATION TECHNOLOGY

In the

Department of Information Technology

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

TSHWANE UNIVERSITY OF TECHNOLOGY

Supervisor: Prof M Mphahlele

Co-Supervisors: Prof S. Ojo & Dr. TE Mathonsi

July 2021

DECLARATION

I Themba Lucky Nkosi, hereby declare that this dissertation entitled, Enhanced Bandwidth Allocation model for improving Quality of Service in Intelligent Home Networks, submitted for the degree MASTERS IN COMPUTING: Information Technology (Communication Networks), at the Tshwane University of Technology, is my work and has not been previously submitted to any other institution of higher education. I further declare that all sources cited or quoted are indicated and acknowledged utilizing a comprehensive list of references.

……………………………………… ……**21/07/2021**……………

Signature Date

DEDICATION

I dedicate this work to all my family members who have supported and encouraged me to continue with my studies.

ACKNOWLEDGMENTS

I thank God, for giving me strength and guiding me to complete this task. Thereafter, I would like to express my sincere gratitude and appreciation to the generosity of my supervisors who contributed to the success of this study: Prof. M Mphahlele, Prof. S Ojo, and Dr. T.E Mathonsi for his continuous professional guidance in keeping me focused on the research topic, for his ongoing support, constructive feedback and motivation throughout the study. I have learned a lot of lessons from him throughout my research study, writing conference papers, and organizing a journal. It was great and an honor for me to work with them.

Finally, I would like to thank my whole family, friends, and colleagues for their invaluable support and motivation during this entire period.

PUBLICATIONS

Parts of this dissertation have been presented and published in the following conference proceedings and journals:

1. Nkosi, T.L., Mphahlele, M., Ojo, S.O. and Mathonsi, T.E., 2019, October. Dynamic Bandwidth Allocation Algorithm to improve Quality of Service in Intelligent Home Networks. In 2019 Open Innovations Conference, Cape Town, South Africa, 2 - 4 October 2019, pp. 83-88. IEEE.
2. Nkosi, T.L., Mphahlele, M., Ojo, S.O. and Mathonsi, T.E., 2020. Enhanced Dynamic Bandwidth Allocation Algorithm for Intelligent Home Networks. International Journal of Communication Networks and Information Security, 12(2), pp.227-234.

ABSTRACT

Over the past few years, the need for communication among devices has resulted in the connection of home devices, thus creating networks known as Intelligent Home Networks (IHNs). IHNs offer and ensure comfort, safety, healthcare, automation, energy conservation, as well as remote management to their users and devices. IHNs make use of intelligent systems to perform their daily operations. On the other hand, introducing the Internet of Things (IoT) has played a tremendous change in the Information Technology (IT) environments, and thus its importance has also been realized within IHNs. Consequently, more and advanced devices and services continue to be introduced and used in IHNs. Meanwhile, the current migration from Internet Protocol version 4 (IPv4) to IPv6 standards also plays its vital role. This has led to a competition of limited network resources, services, and bandwidth as well. Apart from that, the addition of more and advanced devices and network resources into IHNs leads to congestion problems. IHNs, therefore, continue to experience and suffer poor Quality of Service (QoS) when performing operations both locally and remotely leading to unavailable, unreliable, and inefficient bandwidth to the consumers of these networks. In this study, therefore, this study proposes enhanced bandwidth management and allocation algorithm, named Enhanced Dynamic Bandwidth Allocation (EDBA) algorithm for IHNs. This is mainly because none of the existing studies considered designing and developing bandwidth management and allocation algorithms for IHNs. The proposed EDBA allocates the available bandwidth according to the workflow procedures of each subnet in the network. Meanwhile, it ensures that each subnet is assigned or allocated bandwidth according to its workflow. Therefore, EDBA integrates two existing algorithms known as Offline Cooperative Algorithm (OCA) and Particle Swarm Optimization (PSO). The purpose of PSO is to define the priority limits for each subnet and node in the network. Meanwhile, the purpose of OCA is to facilitate the allocation of bandwidth dynamically. The Network Simulator-2 (NS-2) was used to simulate and evaluate EDBA against other traditional bandwidth allocation algorithms. The experimental evaluation was against that of Dynamic QoS-aware Bandwidth Allocation (DQBA) and Data-Driven Allocation (DDA). EDBA algorithm reduced congestions and bandwidth usage. Statement revised to read: The obtained results showed an average throughput of 92%, an average delay of 0.8 seconds, and savings on energy consumption by 95%, compared to Dynamic QoS-aware Bandwidth Allocation (DQBA) and Data-Driven Allocation (DDA).. Therefore, the proposed

EDBA outperforms these traditional algorithms in regards to average throughput, a reduced average delay, and energy consumption.

Keywords — QoS, IHNs, Dynamic Bandwidth Allocation, PSO, OCA.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

GLOSSARY

5G	Fifth Generation
AMI	Advanced Metering Infrastructure
AMR	Automatic Meter Reading
BAN	Body Area Subnet
DBA	Overview of Dynamic Bandwidth Allocation
DDA	Data-Driven Allocation
DHCP	Dynamic Host Control Protocol
DLB	Dynamic Load Balancing
DQBA	Dynamic QoS-aware Bandwidth Allocation
DSCP	Differentiated Services Code Point
DSL	Digital Subscriber Line
FCC	Federal Communications Commission
FiWi	Fiber-Wireless
GHz	GigaHertz
GPS	Global Positioning System
HDTV	High Definition Television
HNGs	Home Network Gateways
HPON	Hybrid Passive Optical Network
HTTP	Hypertext Transfer Protocol
ID	Protocol Identification
IEEE	Electrical and Electronics Engineers
IHNs	Intelligent Home Networks
IoT	Internet of Things
iPod	Portable Media Players
IPv4	Internet Protocol version 4
ISM	Industrial, Scientific, and Medical

ISO	International Organization for Standardization
IT	Information Technology
ITU-T	Telecommunication Standardization Sector
L2CAP	Logical Link Control and Adaptation Protocol
LMP	Link Management Protocol
MATLAB	MATrix LABoratory
MMS	Multimedia Messaging Services
MuPAD	Multi-Processing Algebra Data
EDBA	Enhanced Dynamic Bandwidth Allocation
NS	Network Simulator
OADR	Open Automated Demand Response
OCA	Offline Cooperative Algorithm
OMNET	Optical Micro-Networks
OPNET	Optimum Network Performance
OTcl	Object-oriented Tool command language
PAN	Personal Area Networks
PDAs	Personal Digital Assistants
PON	Passive Optical Network
PON	Passive Optical Network
PSO	Particle Swarm Optimization
QoS	Quality of Service
RBOCs	Regional Bell Operating Companies
RFID	Radio Frequency Identification Devices
SDP	Service Discovery Protocol
SHF	Super High Frequency
SMS	Short Message Sent
SNGs	Sub Network Gateways
SPB-DBA	Staged Priority-Based Dynamic Bandwidth Allocation

STBs	Set-Top Boxes
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UHF	Ultra-High Frequency
VINT	Virtual Inter-Network Testbed
VLAN	Virtual Local Area Network
VM	Virtual Machine
VPN	Virtual Private Networks
VPON	Virtual Personal Overlay Network
Wi-Fi	Wireless Fidelity
WMNs	Wireless Mesh Networks

CHAPTER 1 : INTRODUCTION

1.1. Background

The Internet of Things (IoT) technology establishes a connection between all things and the Internet through different sensing devices and implements intelligence to deal with the identification and management of these things. The information sensing devices include Radio Frequency Identification Devices (RFID), infrared sensors, Global Positioning System (GPS), and laser scanner devices. These devices are all connected to the Internet to implement remote perception and control. As a consequence, over the past few years, the research has seen many advances in physical-layer wireless communication theories and their implementation in both wired and wireless networks. Furthermore, this has led to the advent of computer networks, and thus, there has been a consistent need to have electronic devices in any network. This aids in communication among various devices and the sharing of the limited available network resources and services. In the recent past, the need for communication among these devices has resulted in the connection of home devices, thus creating Intelligent Home Networks (IHNs). IHNs provide and ensure comfort, safety, healthcare, automation, energy conservation, remote management, etc. to devices and users within it (K'Obwanga, Kogeda & Lall, 2016; Liu & Wang, 2017; Nkosi, Mphahlele, Ojo & Mathonsi, 2019; Zhang, Wu, Nie & Zhang, 2017)).

These networks use intelligent systems to ensure that their end-users can perform their daily operations. Intelligent systems assist in self-healing after faults, power outages, reconfigurations, and more (Elsayed, Elhoseny, Riad & Hassanien, 2017; Liu, Wu, Sun, Xu & Wen, 2019). Furthermore, these networks can be accessed and managed either locally or remotely, enabling monitoring, scheduling, and controlling the various devices and users.

In most cases, IHNs could be designed to comprise six sub-networks (subnets), namely: Wireless Fidelity (Wi-Fi), ZigBee, Smart Grid, Bluetooth, Body Area, and Ultra-Wide Band (UWB) (K’Obwanga, Kogeda & Lall, 2016). These subnets have different priorities based on the importance and workflow procedures. The devices in each subnet are connected to Sub Network Gateways (SNGs). On the other hand, the SNGs are connected to Home Network Gateways (HNGs). The responsibility of HNGs is to enable the integration of IHNs with other networks, the Internet, and more.

This study is mainly motivated by a continuous increase in the number of advanced IoT devices and services, which increases network bottlenecks. Apart from that, the current migration from Internet Protocol version 4 (IPv4) to IPv6 standards also plays its vital role in IHNs. This is because the growth of these devices means organizations need to start migrating to IPv6. Thus failing to migrate will restrict their ability to be flexible and innovative. IPv4 addresses have 32 bits, and this allows for only 4 billion addresses to be available for use. In comparison, IPv6 is 128 bits and supports 340 trillion addresses. Besides, unlike IPv4, IPv6 can handle packets more efficiently, deliver better performance, and increase security (Dawadi, Rawat & Joshi, 2019). These issues shall result in competition for the limited available network resources (such as address spaces) and services as well as available bandwidth. Hence, this network type experiences high network congestion leading to poor Quality of Service (QoS) when performing its operations both locally and remotely. This has resulted in unavailable, unreliable, and inefficient bandwidth to the consumers of these networks. The research further concurred that effective resource allocation algorithms are important for bandwidth management to improve QoS and to satisfy the demands of its customers as well (Rafiq, Hayat & Younus, 2018).

In this study, therefore, this study proposes enhanced bandwidth management and allocation algorithm for IHNs. The reason behind all this is that none of the existing research considered developing bandwidth management and allocation algorithms for IHNs. Through an extensive literature review, this study realized that the current existing bandwidth allocation algorithms have been developed for wireless networks such as Wireless Mesh Networks (WMNs) and more. The proposed algorithm allocates the available bandwidth according to the workflow procedures for every node in the network. The idea behind, it is that devices are assigned or allocated bandwidth according to their workflow procedure, and thus, the device with high workflow procedure than others are given high priority and assigned more bandwidth than other devices. Therefore, the proposed model integrates two algorithms, namely; Offline Cooperative Algorithm (OCA) and Particle Swarm Optimization (PSO). OCA bases itself on the game-theoretical framework to any bandwidth allocation problem, and thus, it has been applied in the proposed algorithm to be responsible for the allocation of bandwidth in the network. On the other hand, PSO is an optimization algorithm used to define priority limits for all nodes in the network.

In this study, the simulations were conducted to evaluate the performance of the proposed bandwidth allocation algorithm called the Enhanced Dynamic Bandwidth Allocation (EDBA) algorithm. Therefore, the proposed EDBA algorithm was simulated, tested, and evaluated using Network Simulator-2 (NS-2). NS-2 is a discrete-event network simulation tool built by the Virtual Inter-Network Testbed (VINT) research group from the University of California (Issariyakul & Hossain, 2009). This simulation tool simulates network protocols with varying network architectures and it can simulate both wired and wireless networks. The experimental evaluation of the proposed EDBA algorithm was against Dynamic QoS-aware Bandwidth Allocation (DQBA) proposed and developed by (Hussain, Ahmed, Saikia & Sarma, 2015) and Data-Driven Allocation (DDA) by (Fan, Leng & Yang, 2016). From the simulation results, it is clear that the EDBA algorithm can reduce congestions, saves bandwidth, and more. Consequently, there is an improved average throughput and reduced energy consumption, and average delay on the EDBA compared to DQBA and DDA. The obtained results show an average throughput of 92%, average delay of 0.8 seconds, and saves on energy consumption by 95% compared to Dynamic QoS-aware Bandwidth Allocation (DQBA) and Data-Driven Allocation (DDA). However, this study proposes future research on security issues affecting IHNs and also looks at how the introduction of the Fifth Generation (5G) would affect IHNs in terms of dynamic bandwidth allocation.

1.2. Motivation

Due to the growth of mobile cellular networks and mobile devices such as smartphones, downloading content from the internet using mobile has become a basic demand (Fan, Leng & Yang, 2016).

The internet of Things (IoT) can be defined as connectivity daily things like smartphones, the Internet, TVs, Sensors, and actuators to the internet where the devices are intelligently connected allowing new forms of communication between objects and persons (Imran, 2016).

Previous research has considered these possibility assignments and the resulting Fi-Wi network throughput delay performance for full overloaded traffic where all nodes are continuously back loaded (Chen & Reisslein, 2018).

Lately, the demand for bandwidth services such as video conferencing, video calls and high definition television has been gradually increasing and new network application scenarios (e. g IoT and cloud computing) are endlessly emerging. The visual access network, which allows users to access networks assets, has encountered plain stress on the network bandwidth and governor (Qian, Li, Zhang, Cao, Li & Wang, 2018).

Besides, it is also important to gain deep knowledge in the field of IHNs and bandwidth allocation algorithms. Meanwhile, it is important to make a significant research contribution to the field of IHNs by coming up with tailored solutions that could be industrially recognized by the management of bandwidth on IHNs. Developing research skills and learn how to solve research problems professionally and logically is also of importance.

1.3. Premises of Research

This research study was based on the International Organization for Standardization (ISO) 9001 model, Telecommunication Standardization Sector (ITU-T), and 802.11 standards. ITU-T was used to define how telecommunication networks operate and interwork. The purpose of ISO is to allow one to efficiently develop and implement an algorithm. The model was employed to ensure quality in all phases namely; algorithm design, implementation, testing, and inspection. The idea behind ISO 9001 guidelines is that all phases be planned, controlled, and verified. IEEE 802.11 standard was also used to design network architectures and develop communication models.

1.4. Problem Statement

Intelligent Home Networks (IHNs) have become increasingly popular and this is driven by the emergence of Internet-connected appliances that form part of the emerging breed of IoT devices. Nowadays users of IHNs can remotely monitor and manage their home networks through smart devices, and more. However, more and advanced devices continue to be introduced on IHNs which results in congestion problems and competition of the limited available network resources and services. With this continuous development and innovation of the Internet of Things (IoT) users require high speed and therefore available bandwidth should be managed and allocated accordingly to attain improved QoS. Otherwise, these networks will forever experience poor QoS performing operations both locally and remotely. On the other hand, existing works were more concerned about residential energy monitoring and controlling and also providing analysis on the characteristics of these networks. In addition, security issues were also a major concern ever since these networks were introduced. This study focuses on the fact that more and advanced continue to be introduced within IHNs and surely will yield to a competition of limited network resources and services.

1.5. Research Questions

The main research question from which the research sub-questions are derived focuses on how bandwidth can be managed and allocated efficiently within IHNs to solve the bottlenecks caused by the daily introduction of new and advanced devices.

The sub-questions are as follows:

1) What are the factors affecting QoS in IHNs?
2) What are the salient features of existing bandwidth allocation algorithms to improve QoS in IHNs?
3) How can a bandwidth allocation model for improving QoS in an IHN be developed, using a hybrid of OCA and PSO methods?
4) How can the model be experimentally validated?

1.6. Technical Objectives

The following objectives based on the research questions had to be achieved to fulfill the requirements set by this research:

1) To identify the factors affecting QoS in IHNs.
2) To develop a taxonomy of existing bandwidth allocation algorithms for improving QoS in IHNs.
3) To develop a bandwidth allocation model for improving QoS in an IHN, using a hybrid of OCA and PSO methods.
4) To experimentally validate the model.

1.7. Research Methodology

To produce effective answers to the research questions as well as to ensure that all the technical objectives are met, several processes, methods, and procedures had to be undertaken. A summary of the methodology that was followed to answer the research questions is given in Table 1.1.

1.7.1. Literature Review

The study was conducted using literature review research wherein peer-reviewed articles, journals, books, seminars, white papers, workshops, and conference proceedings were read to fully understand and learn what others have done, how, and what are the various gaps found in that literature.

1.7.2. Modeling

Mathematical models (scientific models) are sets of equations that take into account many factors to represent a phenomenon. Therefore, in this study, mathematical models were used when designing the proposed dynamic bandwidth allocation algorithm.

1.7.3. Design and Implementation

Different network simulation tools allow study and also do examinations of applications and protocols under varying network configurations. These tools further allow users to experiment and implement newly designed models in a controlled manner. There are many existing network simulation tools such as NS-2, Network Simulator-3 (NS-3), OMNET++, MatLab, and more. However, NS-2 will be used for the development and implementation of the proposed EDBA algorithm.

1.7.4. Simulation

The proposed algorithm will be simulated using the NS-2 platform. To evaluate the performance of the algorithm, NS-2 recorded its statistics on trace files which were used to calculate performance metrics such as average throughput, average delay, and energy consumption. The collected quantities were plotted in graphical format using R Programming. Furthermore, AWK scripts were created to calculate the total averages of the considered performance metrics.

Table I: Research Questions, Technical Objectives, and Methods

Research Questions	Technical Objectives	Methods	Deliverables
What are the factors affecting QoS in IHNs?	To identify the factors affecting QoS in IHNs.	Literature Review (books, journals, conference proceedings, and more)	Report on the QoS factors.
What are the salient features of existing bandwidth allocation algorithms to improve QoS in IHNs?	To develop a taxonomy of existing bandwidth allocation algorithms for improving QoS in IHNs.	Literature Review (books, journals, conference proceedings, and more) Taxonomic modeling	Report on existing algorithms.
How can a bandwidth allocation model for improving QoS in an IHN be developed, using a hybrid of OCA and PSO methods?	To develop a bandwidth allocation model for improving QoS in an IHN, using a hybrid of OCA and PSO methods.	Mathematical model design and development through the aid of NS-2.	Development of the proposed EDBA by hybridization of OCA and PSO.
How can the model be experimentally validated?	To experimentally validate the model.	Experimental evaluations conducted using NS-2, R Programming, AWK, and GREP scripts.	Evaluations of the developed EDBA. E.g. Delay

1.8. Contributions of the study

This section as indicated, and draw from the deliverables in Table 1.1 in writing this section with specific contribution:

The contributions of this study are as follows:

a) The EDBA algorithm to use on IHNs to optimize the performance and improve QoS in these networks. The proposed algorithm allocates the available bandwidth according to the workflow of each node device in the network.

b) The method of hybridization, the tool developed for practical use. This is the hybridization of OCA and PSO algorithms. The Offline Cooperative Algorithm is responsible for the allocation of bandwidth in the network. The PSO is responsible for defining priority limits for all nodes in the network. Thus, the nodes with high workflow are prioritized and assigned more bandwidth than others

During the process of undertaking this research, several accomplishments have been achieved including:

a) An enhanced dynamic bandwidth allocation (EDBA) algorithm for IHNs. The algorithm optimizes the performance of IHNs by improving QoS. This is achieved irrespective of the increase in numbers of home devices and services, resulting in congestion problems and competition of the limited available network resources and services.

1.9. Outline of the Dissertation

The remainder of this dissertation is organized as follows: Chapter 2 provides an overview of IHNs and discusses some related works. Chapter 3 discusses system design and architecture where the proposed IHN architecture consisting of six subnets, various devices, and connections are presented. Chapter 4 presents the implementation and simulation of the proposed EDBA algorithm wherein the description of the algorithm development in steps and the experimental setup is presented. Chapter 5 presents results, analysis, and discussions concerning three performance metrics considered in this study, namely: average throughput, average delay, and energy consumption. Finally, Chapter 6 provides a dissertation conclusion and future way forward.

1.10. Summary

The purpose of this chapter was to present an introduction to the proposed research study, stating its importance and relevance to the body of knowledge. The key motivational factors behind the area of study were highlighted; and also pointed out required regulatory standards which would ensure that the study conducted and the algorithm proposed and developed herein conforms to acceptable world-class standards. Most importantly, the chapter introduced the research goals in the form of technical objectives, which are to be met through the provision of relevant answers to the research questions posed in the chapter. Procedures followed to respond to the research questions as well as outputs that were obtained in the course of the research have also been outlined. The next chapter provides a deep discussion of the literature as well as related work.

CHAPTER 2 : LITERATURE REVIEW

2.1 Introduction

This chapter provides a deeper insight into the design and implementation of IHNs. In general, IHNs are examples of computer networks, which recently present an emerging research field that is important for the cohesive existence of users, animals, and devices within these networks. These networks were proposed to provide comfort, safety, energy conservation, healthcare, and security to its users both locally and remotely. This chapter provides discussions on the characteristics, benefits as well as drawbacks of these networks. The study begins by providing an overview, merits, and demerits of intelligent networks as well as intelligent home networks. The study also discusses how the network is proposed by looking at its components in detail.

The remainder of this chapter is organized as follows: Section 2.2 presents an overview of intelligent networks. Section 2.3, briefly discusses Intelligent Home Networks by including their characteristics, benefits, and drawbacks. Section 2.4, provides discussions of the systems of intelligent home networks. Section 2.5 presents a review of the intelligent home networks system design. Section 2.6 presents a review of dynamic bandwidth allocation algorithms. Meanwhile, related works and the various gaps found in the literature are discussed in Sections 2.7 and 2.8. Finally, this chapter is concluded in Section 2.9.

2.2 Overview of Intelligent Networks

The intelligent network refers to the standard network architecture and could be defined as networks that deal with providing specific technical capabilities or services outside the conventional network standard spectrum. It was intended for fixed as well as mobile telecom networks as it allows operators to differentiate themselves by providing value-added services in addition to the standard telecom. These networks are crucial in the building of telecom networks because of utilizing artificial intelligence technologies to improve the QoS (Gao, Zhang, Dong, Sun, Wang, Yang, Wu, Li & de Albuquerque, 2020). In general, these networks use smart or intelligent systems in performing their operations. This provides the network operator with the ability to develop and control services and resources more efficiently. These types of networks were introduced in the mid-1980s by regional Bell operating companies to meet the objectives, including; rapid deployment of services in the network, vendor independence, and standard interfaces, and allow none-Regional Bell Operating Companies (RBOCs) to offer services for increased network usage (Chorafas, 2019). For these networks to fully operate, some decisions are to be made whereby the decision model collects data, analyzed by intelligent systems before making a hypothesis (Berbec, 2009; Sadkhan, Jawad & Alshalah, 2017). Some of the examples include services such as televoting, cool screening, telephone number portability, toll-free calls free phone, prepaid calling, account card cooling, and more (Kniazieva, Shestopalov, Kunup & Kondratenko, 2018).

2.3 Overview of Intelligent Home Networks

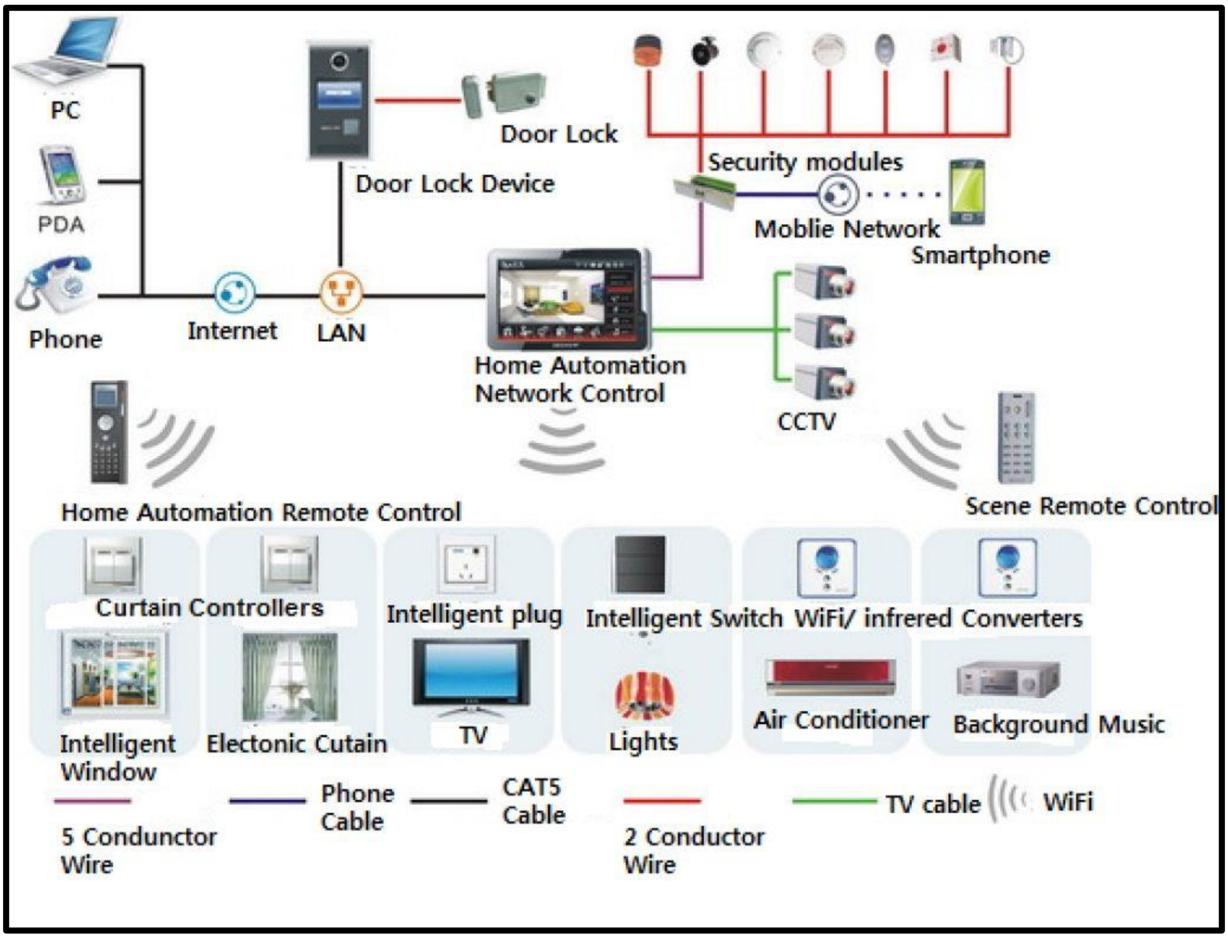

Figure 2:1: Intelligent Home Network (Intelligent Home Network, 2021)

As mentioned previously, Intelligent Home Networks (IHNs) are becoming a trend in which intelligence can be built-in to provide improved user comfort at an incremental cost (Kaiwen, Kumar, Xavier & Panda, 2016). The intelligent home network is an application of ubiquitous computing that can provide users automated or assistive services in the form of ambient intelligence, remote and local home control, management, and automation. Intelligent home networks provide comfort, safety, energy conservation, and healthcare to consumers both locally and remotely (Hou *et al.*, 2008; K'Obwanga, Kogeda & Lall, 2016; Liu & Ge, 2017). The intelligent network provides services independently of both networks owners and equipment vendors. The intelligent network enables users to introduce self-tailored services within their devices. Therefore, any intelligent network could be realized as a combination of both knowledge modules and intelligent models and structures. In most cases, intelligent networks have four basic information-processing models: control, data, demand, and information-driven.

2.4 Benefits of Intelligent Home Networks

Over the past few years, the design of the home intelligent gateway has become the focus of research (Yang, 2017). The benefits of Intelligent Home Networks include (Crossler & Choo, 2018; K'Obwanga, Kogeda & Lall, 2016):

a) Provision of information on a real-time basis, thus ensuring minimal delay between network and devices. Most intelligent networks are timed and carry out an operation within the configured time limit.
b) Permits dynamic configuration of several devices into a particular network.
c) Minimum human errors are experienced during network setup and configurations.
d) Provision of security features to users such as using CCTV cameras for monitoring purposes.
e) Aids in saving lives, wherein, devices in hospitals monitor patient health progress in real-time. The captured information is promptly transmitted to relevant doctors who affect treatment. Moreover, road lighting control systems help in preventing accidents that would have occurred at road junctions where several users compete for access and use.
f) Permits remote connectivity and monitoring of devices within a particular network.

2.5 Drawbacks of Intelligent Networks

The limitations of Intelligent Home Networks include (Gunge & Yalagi, 2016; K'Obwanga, Kogeda & Lall, 2016):

a) Network devices and relevant software used in setting up an intelligent network are slightly expensive to most consumers.

b) Setting up an intelligent network requires highly skilled experts who are expensive to hire and maintain.
c) Some intelligent networks are prone to errors and external attacks by intruders (hackers) thus, resulting in frustrations to the consumers.
d) Most intelligent networks require a constant power supply.

2.6 Characteristics of Intelligent Home Networks

The intelligent network supports services such as televoting, call transfers, call queuing, seamless Multimedia Messaging Services (MMS), mass calling services, private number plans, Virtual Private Networks (VPN), telephone number portability, call screening, toll-free calls, etc. Most telecommunication networks can dynamically locate a user, automatically update the SIM menu, allows SIM swapping, update users on weather changes, and permits self-healing. Roads on the other hand also have intelligent network devices mounted onto them e.g., road control lights (robots). Intelligent networks have also been deployed as a private network entity in homes, offices, airports, hospitals, meteorological departments, military bases, restaurants, and hotels. In military bases, intelligent networks help in monitoring the borders of a country from external attacks. This is done via the use of a wireless machine to machine communication. Machines at respective borders automatically communicate to central machines, thereby, providing all the needed information. Military officers can monitor the safety of the country at a central pool. In airports, intelligent networks assist in recording information of physically screened passengers and luggage automatically. The screened information is instantaneously sent to the analysis database within the airport premises, thereafter, accessibility is granted to luggage and passengers who have not shown any malfunctioning in the system, thereby enhancing the security of a country.

2.7 Applications of Intelligent Home Networks

Several systems help in home automation. Some of these applications include Loxone, Smart Things, Control4, Incontrol, Vivint, Creston, Savant, Via International, etc., and contains security, control, monitoring, and management features.

Some of the features of intelligent home networks together with their associated apps include among others:

a) Digital gate mounted with an inbuilt camera, button and voice recognition device, and system.
b) Digital doors mounted with biometric detection devices and numeric keypad. Each door is configured with a unique encrypted password and each user-provided authorization to specific rooms. When a given user walks in or out of a room, lights turn on or off respectively.
c) Lights control in which lights can be turned on/off in any room. Further, the preferences of the users' light dimming are enabled. Security lights are dynamically configured to turn on/off at specific times of the day.
d) Appliances control in which High Definition Television (HDTV), Amphitheatre, stove, fan, microwave, refrigerators, etc., can be monitored both locally and remotely. A user can regulate volumes of various sound systems, change channels or movies to watch in HDTV, shut down entire electricity usage at home, monitor foods in the fridge, and automatically order more food products from relevant store wares.
e) Window and door curtains control.
f) Temperature and ventilation control.
g) Vehicle detection system and plant and pet monitoring system.
h) Health monitoring devices and apps that check on the heart rate beats, pressure, weight, temperature, cardiac arrest, etc. This information is transmitted on a real-time basis to family doctors for effective actions. The elderly and home patients are monitored in real-time via objects mounted onto the body or at specific locations within the house/home.

2.8 Overview of Intelligent Home Network Systems

The intelligent home network is going to be segmented into six subnets, namely; ZigBee, Smart Grid, Body Area, Wi-Fi, UWB, and Bluetooth.

2.8.1 Wi-Fi Subnet

This is a local area wireless technology that permits electronic devices to exchange data or connect to the internet using 2.4 GigaHertz (GHz) Ultra High Frequency (UHF) and 5 GHz Super High Frequency (SHF) radio waves. Wi-Fi products are based on the Institute of Electrical and Electronics Engineers (IEEE) 802.11 standards, e.g., 802.11a operating at 5 GHz, 802.11b, 802.11e and 802.11g operating at 2.4 GHz, and 802.11i, 802.11n, and 802.11ac operating at both 2.4 and 5 GHz. Wi-Fi network devices operating on 802.11a and 802.11b standards are not backward compatible with each other. This is due to the different frequencies and bandwidths that they operate on. However, devices operating on 802.11n and 802.11ac standards are compatible due to the similar frequency range of operation.

Wi-Fi subnet has accommodated devices like smartphones, laptops, tablets, Personal Digital Assistants (PDAs), digital cameras, printers, video game consoles, digital audio players, etc., as shown in Figure 2.1. Wi-Fi subnet has been configured using Cisco wireless Linksys routers. However, the access distance of Wi-Fi subnet devices is determined by specific 802.11 standards it runs, device transmitter strength, and nature of physical obstructions or radio interference in the surrounding area. Brick walls and metal frames in homes reduce the range of Wi-Fi networks by 25% or more. Radio signal interference from microwave ovens, ZigBee devices, and other equipment also affects negatively the Wi-Fi network range. This is because 2.4 GHz radios are commonly used in consumer gadgets.

2.8.2 Bluetooth Subnet

This is a wireless technology standard for exchanging data over short distances. It uses short-wavelength UHF radio waves between the range of 2.4 to 2.485 GHz from fixed, mobile devices and building Personal Area Networks (PAN). Devices in the Bluetooth subnet do not need to be in a visual line of sight to each other since they use the radio communication system. However, a quasi-optical wireless path must be viable. Bluetooth is standardized as IEEE 802.15.1. Some of the devices contained in the Bluetooth subnet include smartphones, tablets, PDAs, laptops, hands-free headsets, printers, keyboards, mouse, and more.

Each Bluetooth version has its data rate, coverage distance, and maximum permitted power and application throughput. Version 1.2 was the first version to be implemented. It had a data rate of 1Mbit/s, a distance of approximately 1meter (m), and maximum application throughput >80kbit/s. Version 2.0 + EDR had a data rate of 3Mbit/s, a distance of approximately 10m, and maximum throughput >80kbit/s. Whereas, version 4.0, which is the current version in many Bluetooth devices, has a data rate of 24Mbit/s and a range of approximately 100m. Effective range varies due to propagation conditions, material coverage, production sample variations, antenna configurations, and battery conditions.

The Bluetooth stack contains protocols like:

a) Link Management Protocol (LMP): is used for setting up and controlling radio links between two devices. It is implemented on the controller.
b) Logical Link Control and Adaptation Protocol (L2CAP): is used to multiplex multiple logical connections between two devices using different higher-level protocols. It also provides segmentation and reassembly of the on-air packet.
c) Service Discovery Protocol (SDP): allows a given device to discover services offered by other devices and their associated parameters.

A Bluetooth device in discoverable mode transmits the device name, device class, list of services, and technical information like manufacturer, on-demand. A device that has got the address of the device it wants to connect to, responds to direct connection requests, and transmits automatically the data requested. Devices that are paired have their communications synchronized and automated as opposed to unpaired devices. Current Bluetooth devices have encryption keys that are automatically generated between two communicating devices. The keys assist in pairing and for security. Once devices are paired, they will not require any other key for their subsequent communications provided the settings are not changed.

2.8.3 Smart Grid Subnet

A smart grid subnet consists of three layers: Automatic Meter Reading (AMR), Advanced Metering Infrastructure (AMI), and AMI compatibility (AMI+). AMR layer is responsible for real-time power consumption measurement. AMI layer is responsible for consumer outage detection, remote turn on/off, hourly remote meter readings, and rate programming. AMI+ refers to the system that collects, monitors, controls, analyzes, and optimizes energy usage through various communication media either on-demand or on pre-defined schedules. It also integrates alternative energy resources(Al-Ali, El-Hag, Dhaouadi & Zainaldain, 2011).

Smart grid subnet devices allow for both local and remote monitoring by home users. In an intelligent home network, there is always an integration of both Smart Grid and ZigBee appliances. The energy consumption of different devices mounted on the ZigBee subnet is controlled via smart grid devices. The integration of these two subnets enables utility companies and homeowners to communicate using Short Message Sent (SMS) or web applications via public mobile networks. This enables monitoring and managing the power consumption of home appliances. Two-way communication is always developed and implemented to access the power consumption digital meters, detect peaks and outages at consumer premises, send hourly/daily tariff bills, and remotely interrupt or restore power supply.

Smart grid subnet uses different international communication standards in their operations: IEEE P2030, Open Automated Demand Response (Open ADR), ITU-T G.hn are some of the available standards for smart grid subnets and devices. Smart grid subnet devices include automatic door switch, water heater, power meter reader, water cooler, geyser, fire alarm devices, etc., as shown in Figure 2.1.

2.8.4 Body Area Subnet (BAN)

This is a system of devices close to the body of a person. The devices cooperate for the benefit of the user. It is defined by IEEE 802.15 standards. BAN can be used in cases like Body Sensor Network (BSN), Sports and Fitness Monitoring, Wireless Audio, Mobile Device Integration, and Personal Video Devices. Each of the cases has unique requirements in terms of bandwidth and signal distance.

BANs are always viewed in the provision of health services remotely, where different BAN devices are located at an approximate distance between zeros - to - two meters from the patients to aid in monitoring purposes. These devices can be used to measure heartbeat rate, pressure, cardiac arrest, or any other illness that can be monitored in a patient and directly relayed to remote doctors. Equally, information gathered from a patient can be stored at the home server and be used later, at a point of request, in making health decisions by practitioners. However, BAN is not commonly used in homes since it is still hard to effectively propagate a signal in and around the human body. Consequently, the development of Human-Computer Interaction (HCI) to make the technology usable is still a challenge (Karulf, 2008).

The current emergence in BANs has brought Wireless BAN (WBAN) which is based on IEEE 802.15.6 and IEEE 802.15.4j standards. These standards have the major aim of improving speed, accuracy, and reliability of communication between sensors and actuators within, on, and in the immediate proximity to the human body. This helps in improving the quality of life for humans. WBANs can be applied in both medical-related situations such as wearable WBAN, which is mounted around the human body for patient monitoring, implant WBAN, which is mounted onto the human body to detect cardiovascular and cancer diseases, remote control of medical devices, patient monitoring and telemedicine systems, and non-medical related situations such as real-time streaming, entertainment application, emergency, emotion detection, and secure authentication. Nodes present in a WBAN are based on functionalities and they include Personal Device (PD), sensors, actuators, implants node, body surface node, external node, coordinator, end nodes, and relay(Movassaghi, Abolhasan, Lipman, Smith & Jamalipour, 2014). Devices found and used within BAN are smartphones, emergency lights, automatic gates, and doors, Personal Digital Assistants (PDAs), and more.

2.8.5 ZigBee Subnet

This is a specification of a suite of high-level communication protocols used to create personal area networks built from small, low-power digital radios. It is based on IEEE 802.15 standard and IEEE 802.15.4 for low rate wireless PAN (WPAN). ZigBee is used in low data rate applications that require long battery life and secure networking, e.g., wireless light switches, electrical meters with in-home displays, traffic management systems, and other consumer and industrial equipment that require short-range low rate wireless data transfer.

ZigBee operates on Industrial, Scientific, and Medical (ISM) radio bands of 2.4GHz. Devices in the ZigBee subnet are categorized into three: ZigBee Coordinator, which forms the route of the network and might bridge to other networks, store information about the network. ZigBee Router/Gateway runs the applications and acts as a passage of data from other devices. ZigBee End Device, which has the functionalities to talk to the parent node and cannot relay data from other devices. The 802.15.4 has two operation modes, beacon-enabled, and non-beacon-enabled modes, and supports a maximum data rate of 250 kbps which at times is insufficient for the applications that generate heavier data traffic(Chen, Wan, Gonzalez, Liao & Leung, 2014). Devices/appliances mounted onto the ZigBee subnet include Refrigerator, Electrical and gas cooker, clothe and utensils washing machines, coffee maker, etc.

2.8.6 Ultra Wide Band (UWB) Subnet

It is a radio technology that can be used with frequencies from 3.1 GHz to 10.6 GHz. Each radio channel has a bandwidth of more than 500 MHz based on its center frequency. UWB operates at a very low energy level for short-range and high-bandwidth wireless personal area network communications. It uses a large portion of the radio spectrum and is based on IEEE 802.15.3a standards.

The wide signal in UWB allows for the support of high wireless data rates of between 480 Mbps - to - 1.6 Gbps at distances of few meters. High data rate UWB enables the use of wireless monitors, efficient transfer of data from digital camcorders/videos, wireless printing of digital pictures from a camera without the need for a personal computer, and file transfer between cell phone handsets and handheld devices such as portable media players (iPod). However, at longer distances, UWB data rates drop considerably. Devices supported by the UWB subnet include MP3 recorders and players, High-Definition TVs (HDTVs), Set-Top Boxes (STBs), gaming systems, PDAs, cell phones, etc. as shown in Figure 2.1.

However, UWB has some drawbacks to its full deployment for use within homes. These drawbacks include:

- Performance, including over-the-air data rate, power consumption, co-existence with other wireless devices, immunity to interference, and link robustness.
- Interoperability.
- Quality of service.
- Global spectrum allocation, etc.

2.9 Overview of Dynamic Bandwidth Allocation (DBA)

Bandwidth allocation is the process of assigning radio frequencies to different applications. The radio spectrum is a finite resource, which means there is a great need for an effective allocation process. In the United States, the Federal Communications Commission or FCC has the responsibility of allocating discrete portions of the spectrum, or bands, to various industries. Bandwidth allocation means how many portions of your internet speed you can set aside for the particular allocated app. This will help you to do another internet of all things while the other service is running. Dynamic bandwidth allocation is a technique by which traffic bandwidth in a shared telecommunications medium can be allocated on demand and fairly between different users of that bandwidth. This is a form of bandwidth management and is essentially the same thing as statistical multiplexing. Where the sharing of a link adapts in some way to the instantaneous traffic demands of the nodes connected to the link.

Dynamic bandwidth allocation takes advantage of several attributes of shared networks: (1) all users are typically not connected to the network at one time (2) even when connected, users are not transmitting data (or voice or video) at all times (3) most traffic occurs in bursts -- there are gaps between packets of information that can be filled with other user traffic

Different network protocols implement dynamic bandwidth allocation in different ways. These methods are typically defined in standards developed by standards bodies such as the ITU, IEEE, FSAN, or IETF. One example is defined in the ITU G.983 specification for the passive optical network (PON).

2.10 Related Works

Homes are becoming increasingly smart (thus called intelligent homes), driven by the emergence of Internet-connected appliances that form part of the emerging breed of Internet-of-Things (IoT) devices (Sivaraman, Gharakheili, Vishwanath, Boreli & Mehani, 2015). This enables home users to remotely monitor and manage their home environment. However, to date, nothing much has been done to manage bandwidth within this network.

(McEwen & Cassimally, 2013) proposed a home network QoS harmonization model that used a class-based communication protocol. The model assisted in differentiating priorities in the third and second layers. In the model, Differentiated Services Code Point (DSCP) was used in changing signal state, priority level, and protocol Identification (ID) of classified and marked packets in layer three from standardized to harmonize. Consequently, DSCP values in the second layer were recovered by protocol IDs from the mapping table. This aided in optimizing bandwidth allocation in the network. However, the proposed EDBA model has used all available bits in DSCP in classifying and prioritizing six subnets in the intelligent home network.

(Macone, Oddi, Palo & Suraci, 2013) developed a Dynamic Load Balancing (DLB) model to facilitate QoS and mobility management in next-generation home networks. The model responded to network changes whenever there were link failures or node mobility in home networks. The model used a combination of Langrangian Relaxation, Dijkstra algorithm, and Column Generation to achieve load balancing. This enabled development and use of the Inter MAC layer. Dijkstra aided in establishing the shortest path that a signal would traverse from source to destination. This was aided via multiple paths between nodes thus guaranteeing required bandwidth and minimum delay. Sessions that shared the same source, destination, and level of QoS were aggregated and used. Column Generation solved linear programming variable problems, yielding iteration. The research also addressed the problem of nonlinear programming complexities, thereby yielding an optimum solution. The model ensured efficient use of link bandwidth, the mobility of users, and reduction in delay.

(Choubey, Pateria, Saxena, SB, Jha & PM, 2015) proposed a localized data processing and decision-making framework for an IoT-enabled environment. It constituted a master unit that efficiently manages local sensor networks in Smart Homes. The master unit collected data from the network of sensors installed in different locations within and around the house and intelligently identifies the dependencies among them. Based on this, the set of sensors was classified as dependent whose output values can be predicted using data from the remaining of them. This knowledge extracted locally was shared with the Cloud Server, thereby providing a refined picture of the sensory data. Based on locally extracted knowledge, the sensors were tuned in real-time to minimize the redundancy in usage and power consumption. The assumption was that, in cases of failure of a sensor unit, approximate decisions can still be made based on previous patterns.

(Hussain *et al.*, 2015) proposed a QoS-aware dynamic bandwidth allocation scheme to mitigate the congestion problem in gateway-based multi-hop Wi-Fi-based long-distance networks and thereby enhance QoS guarantees for real-time traffic. This study presented a dynamic slot scheduling scheme that efficiently distributes the unused TDMA time slots among the needy nodes. The distribution process was carried out hierarchically through the use of the parent-child relationship of a tree topology. By doing so, the end-to-end performance of real-time traffic was enhanced manifold. Furthermore, the proposed protocol provided assured packet forwarding for already admitted flows by allowing transmission of nodes according to the capacity of ancestral links. Simulation results showed that their proposed protocol achieved a significant performance improvement in terms of throughput and delay of real-time traffic.

(Guo, Liu, Lui & Jin, 2015) applied the game-theoretic framework to the bandwidth allocation problem in IaaS datacenters. Through a cooperative approach, the research presented the design of Falloc with both offline and online algorithms that solves the optimization of the Nash bargaining solution. Falloc guaranteed the bandwidth requirement based on the base bandwidth for each Virtual Machine (VM) and proportionally shares the residual available bandwidth according to VM's weight. The purpose of a VM is to virtualize or emulate a computer system. The experimental evaluation with prototype implementation showed that Falloc can provide flexible fairness for VMs by balancing the tradeoff between bandwidth guarantee and proportional bandwidth share. Meanwhile, the trace-driven simulations showed that Falloc can achieve high network utilization and good job completion time in data center networks.

Fan, Leng & Yang (2016) proposed a bandwidth allocation algorithm that uses the collected user data and network data to advance the number of cellular network users. This algorithm can be applied in circumstances where multiple users demanded the same content, such as popular TV Show or shared material, over cellular networks. With the huge amounts of data collected from different user devices, users that are close to each other can be grouped into one cluster to effectively share available services and resources. Since the cluster users share bandwidth, the saved bandwidth can be used to improve the throughput of other users. The simulation results showed that the proposed bandwidth allocation algorithm can hugely advance the throughput of users and drop the delay of the content getting. Furthermore, it could focus on distribution diversity for various types of traffic (such as video and voice) in terms of data rate and delay, since different QoS requirements impact the formation of a coalition.

(Mennicken, 2016) implemented a smart home Machine to Machine (M2M) architecture model. In the model, machines were used in gathering, storing, and uploading processed data into the cloud for analysis and processing. The stored information in the cloud was used in processing, scheduling, and controlling home network events and resources, thus providing real-time interactions and least bandwidth usage. However, the proposed EDBA model has used locally based HNG that gathers, stores, processes, schedules, and controls home network resources. Equally, the model has used Virtual Local Area Network (VLAN) to segment the network into six subnets and used DSCP in classifying, scheduling, and prioritizing supported services into six classes.

(Chen & Reisslein, 2018) developed an advanced universal delay-through model of dynamic bandwidth allocation (DBA) within Passive Optical Networks (PONs). The purpose was to accommodate non-heavy poison traffic. Based on their formalisms and insight attained from their throughput-delay analysis, they developed a novel traffic control approach to provide restrictions in terms of the source traffic inserted at the WMN nodes and into the Fiber-Wireless (FiWi) network. Also, novel node-level assignments of channel access probabilities for Wireless Mesh Network (WMN) nodes. Meanwhile, forwarding probabilities for relay packets at the WMN nodes. These researchers discussed the upstream bandwidth allocation technique and the bandwidth allocation mechanism in software-defined networks. However, Hybrid Passive Optical Network (HPON) was their primary research. By presenting the concept of Virtual Personal Overlay Network (VPON), an exact Staged Priority-Based Dynamic Bandwidth Allocation (SPB-DBA) mechanism for inter-VPON was presented. It contained a two-stage admission structure of bandwidth allocation for VPONs with distinguished importance levels and bandwidth demand priorities. Their simulation results showed that the SPB–DBA mechanism, provides effective bandwidth sustenance VPONs with importance level diversity and guarantees the admission fairness among VPONs, and also advances the performances of bandwidth utilization, middling satisfaction ratio, and packet average delay.

(Qian *et al.*, 2018) proposed a software-defined HPON architecture, in which multiple Passive Optical Network (PON) systems using various PHY multiplexing technologies can coexist. As the core of this architecture, the controller performs some functions, including information collection, network abstraction, VPON networking, and bandwidth management. The upstream bandwidth allocation procedure is discussed, and the dynamic bandwidth allocation mechanism in software-defined HPON is primarily researched. By introducing the concept of VPON, a precise SPB-DBA mechanism for inter-VPON is presented, which consists of the two-stage admission scheme of bandwidth allocation to avoid frequent bandwidth reallocation, and a detailed weighted bandwidth allocation algorithm to allocate bandwidth for VPONs with differentiated priority levels and bandwidth demand priorities. Simulation results suggest that the SPB-DBA mechanism, providing effective bandwidth support for VPONs with priority level differentiation and ensuring the access fairness among VPONs, improves the performances in bandwidth utilization, average satisfaction ratio, and packet average delay.

(Zouaoui, Boussaid & Mtibaa, 2019) developed a priority scheduling model for real-time systems. The model scheduled five processes and assigned fixed priority to each. In the model, lower priority processes had larger waiting and response time and got interrupted by incoming higher priority processes. This made them starved. Moreover, the least waiting time and packet loss were observed in higher priority processes than lower priority processes. Their model is similar to our proposed model since they did classify and prioritize processes (services). However, in our proposed model, this study has used DSCP to classify, prioritize, and schedule six subnets in the network. Consequently, this study has iteratively scheduled each subnet in the media which increases the number of packets transmitted to each. This further reduces the delay duration between two successive transmitting subnets in the media. This also eliminates starvation and ensures the optimized performance of intelligent home networks irrespective of the increase in several devices and services into it.

However, in the enhanced (EDBA) algorithm, this study has used the PSO algorithm to aid in classifying, prioritizing, and scheduling subnets in the network, which increases the number of packets that can be transmitted to each subnet. This assisted us to achieve network load balancing irrespective of the addition of more devices. Meanwhile, it facilitated us to realize an optimized performing home network irrespective of the increase in several devices and services.

2.11 The gap in the Literature

Over the past few years, smart homes also known as Intelligent Home Networks (IHNs) have become increasingly popular, driven by the emergence of Internet-connected appliances that form part of the emerging breed of IoT devices. This has facilitated the users of IHNs to remotely monitor and manage their home network environments using things such as mobile devices, and more. On the other hand, nothing much has been done to efficiently manage and allocate bandwidth on these networks. Most of the existing works were more focusing on residential energy monitoring and controlling. Meanwhile, providing analysis on the characteristics of these networks. In addition, security issues were also a major concern ever since these networks were introduced. This work is also motivated by the fact that more and advanced devices continue to be introduced within IHNs which results in limited network resources and services. Besides, the addition of more and advanced IoT devices into these networks further causes congestion problems. The IHNs, therefore, are seen to continue experiencing and suffering from poor Quality of Service (QoS) when performing operations both locally and remotely leading to unavailable, unreliable, and inefficient bandwidth to the consumers of these networks. In this study, therefore, an enhanced bandwidth management and allocation algorithm for IHNs called Enhanced Dynamic Bandwidth Allocation (EDBA) algorithm is proposed.

2.12 Summary

In this chapter, this study provided a detailed discussion of intelligent home networks. This study presented an overview, characteristics, and various existing environments, besides home networks, where intelligent networks may be applied and used. This study also presented various benefits and limitations of intelligent networks. Consequently, this study discussed the overview of intelligent home networks. This study provided various benefits that intelligent home networks offer to consumers. On the other hand, this study presented various features and detailed composure, subnets, of intelligent home networks. Furthermore, this study has presented an overview of bandwidth allocation. Equally, this study discussed in detail, the existing dynamic bandwidth allocation algorithms and various gaps available in those algorithms. Finally, this study provided a discussion of some research works done previously on optimizing performance as well as the dynamic bandwidth allocation algorithm for intelligent home networks. In the next chapter, this study presents contributions made by this dissertation. Furthermore, this study illustrates the design of intelligent home networks by introducing the necessary network components and protocols.

CHAPTER 3 :SYSTEM DESIGN AND ARCHITECTURE

3.1. Introduction

In this chapter, this study presents the system design and architecture of the proposed EDBA algorithm. This study begins by discussing all the various system components and configurations required. Furthermore, this study presents and also discusses the physical connections done in the network to achieve the objective of this study. Besides, this study provides discussions on the importance that each device, configuration, and connection it does within the network in optimizing and improving the performance of IHNs. Moreover, this study present and discuss the design of the proposed algorithm, Enhanced Dynamic Bandwidth Allocation (EDBA). In the proposed EDBA algorithm, optimizing the network performance and improving QoS is the principle. On the other hand, this study also defines various sub-optimization procedures involved in designing the proposed bandwidth management algorithm.

The rest of this chapter is organized as follows: In Section 3.2, this study presents system building blocks and architecture. In Section 3.3, this study presents and discusses the modeling of the system and the various assumptions made to achieve the expected results. In Section 3.4, this study summarizes this chapter.

3.2. System Architecture

Over the past few years, it has been realized that the addition of more home devices causes traffic load, competition, and depletion of the limited network resources. This results in poor QoS of IHNs. This study proposes to use the system architecture designed by K'Obwanga, et al. (2016). The proposed architecture in the form of star topology to ensure centralized management and configurations is shown in Figure 3.1. The proposed topology has segmented the intelligent home network into six subnets: ZigBee, Wi-Fi, Bluetooth, UWB, Smart Grid, and Body Area.

This has aided in subnets prioritization, thereby, ensuring adequate distribution of the available limited intelligent home network resources. In the adopted architecture, the Multi-Layer 3

Switch is configured as a Home Network Gateway (HNG). Some of the available Multi-Layer 3 Switches include NETGEAR M4100, Cisco SG300 series, SG500-28P, Cisco 3650, 4500, and 6500. Multi-Layer 3 Switch can handle large amounts of traffic, supports hardware-based packet switching, permits routing and prioritization of VLANs, prioritizes packets by the six bits in IP DSCP, and implements QoS DiffServe. Multi-Layer 3 Switch also doubles up as the VLAN Trunking Protocol (VTP) server. Consequently, the architecture has used VLAN configuration in specifying port bandwidth capacity and allocation of interfaces to respective SNGs from HNG.

Furthermore, it has been configured and used seven VLANs at the HNG. It has assigned VLANs 60, 50, 40, 30, 20, 10 to respective subnets based on pre-determined priority, and VLAN 100 for intelligent home network management purposes. Moreover, Layer 2 switches have been configured as VTP clients, thus making them SNGs. Some of the available Layer 2 switches include DES – 3200 series, S3610 series, S3100 series, etc. Each SNG has a VLAN number and name to aid in easier identification. On the other hand, the Dynamic Host Control Protocol (DHCP) pool has been used, which dynamically allocates IP addresses to subnet devices, in the design.

This limits the direct involvement of users in allocating IP addresses. Equally, this study has configured wireless sub-networks to use Linksys wireless routers as SNGs. Looking at Figure 3.1, the Linksys routers are named with subnetwork names and provided with passwords to aid in authentication and authorization to intended users only. This limits the wastage of network bandwidths and enhances security. The internet interfaces of the Linksys wireless routers have an assigned IP address that aids in connection to the rest of the network and remote locations. Also, the network has provided an IP address, enabled DHCP, configured starting IP address, and left open maximum IP addresses that can be supported by each wireless subnet. This permits the introduction of more devices into the wireless subnets.

The secure server is configured to house databases, files, emails, configurations, management, security, etc., which aids in the provision of timely and available intelligent home network resources and operations. The IP address of the server doubles up as a DNS address. This address is used by home devices in accessing server resources. For remote connectivity, the use of the Secure Router and Virtual Private Network (VPN) is applied. There is a connection between HNG and a secure router. The secure router further connects to remote sites via Internet Service Provider (ISP) or Digital Subscriber Line (DSL). The VLAN database of the secure router is reconfigured with the assigned VLAN names and numbers to aid in the identification and prioritization of incoming remote signals. The single gateway at the secure router helps in traffic management from and/ or to the VLANs.

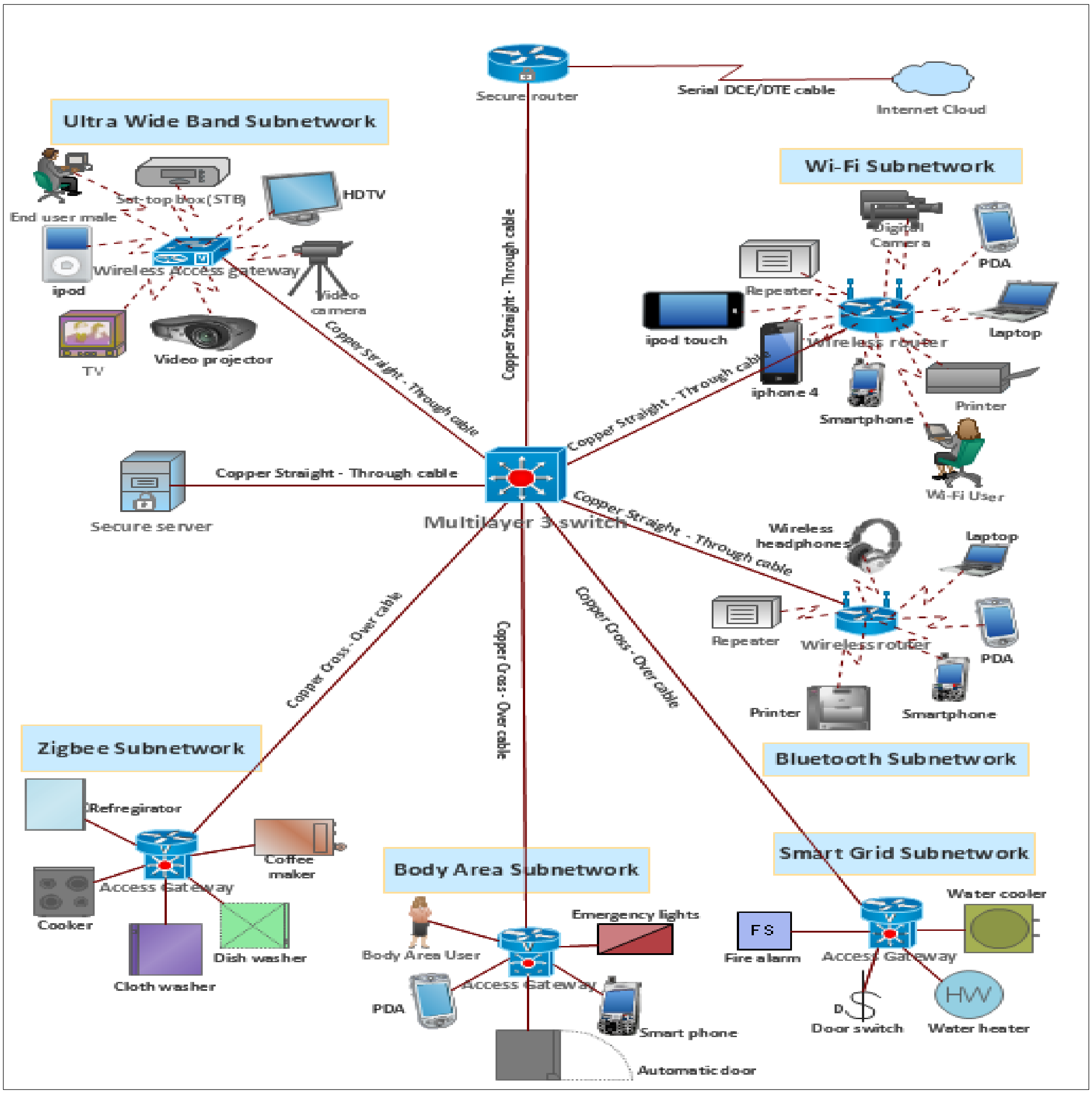

Figure 3:1: Typical Intelligent Home Network (K'Obwanga, Kogeda & Lall, 2016)

3.3. System Design

In implementing the proposed model, this study has modeled the network by a directed graph which is represented as a tree of $T = (V, E)$ whereby V is the set of nodes and E is the set of links in the network. In the network $V = (v_1, v_2, \dots, v_n)$ and $E = (e_1, e_2, \dots, e_m)$. Let $n \in V$ be wired and wireless nodes. Moreover, for each link $l = (s, n, t) \in E$ be wired and wireless links.

Furthermore, the model also distinguishes three different types of nodes in the network and gives them different responsibilities. The nodes are categorized as follows: i) root node, ii) relay node, and iii) child node. The root node is special and acts as the central coordinator of the network. It carries out the task of gateway nodes. On the other hand, the relay node receives the external requests and forwards them to their child nodes. Also, it forwards node join requests to its root node. These nodes perform dynamic slot assignments on receipt of the traffic demands from their children nodes. The child nodes always have exactly one adjacent link. The child nodes are the end-points of the network which carry out the task of receiving control packets, generating node join request, and sending bandwidth demands to their parent nodes.

In the proposed scheme, this study logically divides the proposed IHN into 1-hop clusters as shown in Figure 1. All the clusters exhibit a parent-child relationship and have identical behavior. As a result, in this study, therefore, this study has introduced the process of relay nodes optimization, which solves the bottleneck with bandwidth allocation. Furthermore, this study has applied the Offline Cooperative Algorithm (OCA), which bases itself on the game-theoretical framework to any bandwidth allocation problem, and thus, it has been applied in the proposed algorithm to solve the bottleneck with bandwidth allocation in the network.

3.3.1 Particle Swarm Optimization (PSO) Algorithm

The Particle Swarm Optimization (PSO) algorithm was proposed as a stochastic optimization technique that bases itself on a swarm (Kennedy & Eberhart, 1995). Its' purpose is to simulate the animals' social behavior, including insects, herds, birds, and fishes (Wang, Tan & Liu, 2018). The algorithm was proposed to define behaviors of flocks in a swarm such as cohesion which means they stick together, the separation which means they do not come too close, and alignment which means they follow the general heading of the flock (Ndlovu, Kogeda & Lall, 2018). This algorithm simulates a random search in the design space for the maximum value of the objective. (Wang, Tan & Liu, 2018) stated that Eberhart and Kennedy implemented PSO based on two paradigms, namely; one globally-oriented (GBEST) and one locally-oriented (LBEST). PSO functions as much as Genetic Algorithms (GAs) besides that PSO facilitates only GBEST and LBEST entities to share information with other entities meanwhile GAs facilitate all entities to have the ability to share information (Kumar, Mehta & Kulkarni, 2019).

In its most general design, a particle moves according to its own best-known position in the search space as well as the best-known position of its neighbors. At each instant, a particle p takes a new position vector, $V_p(t+1)$, which is normally calculated as follows:

$$V_p(t+1) = wV_p(t) + c_1R_1\left(Xbest_p(t) - X_p(t)\right) + c_2R_2(Xbest_p(t) - X_p(t)) \quad (1)$$

$$X_p(t+1) = X_p(t) + V_p(t+1) \quad (2)$$

Whereby w represents a parameter called inertia weight. Meanwhile, c1 and $c_1$and c_2 are two parameters called, respectively, the cognitive factor or (self-recognition factor) and social factor (or social-component factor); and R_1 and R_2 are two square diagonal matrices in which the entries on the main diagonal are random numbers in the interval [0,1]. On the other hand, $Xbest_p(t)$ is the best position reached by a particle p up to time t. Therefore, $Xbest_p(t)$ which is normally referred to as the neighborhood best, represents the best position that is found by the neighbors of p. In normal cases, the size of the swarm, the topology, and values assigned to the PSO parameters influence the relative importance of exploration versus exploitation such as the ability to test various points in the search space during the exploration as well as the ability to focus the search around a promising area during the exploitation to have a refined candidate solution (Lynn & Suganthan, 2015). In this study, it is used to classify and provide prioritization to the defined subnetworks also named subnets.

3.3.2 Offline Cooperative Algorithm (OCA) Algorithm

The algorithm was proposed based on the game-theoretical framework to present the bandwidth allocation algorithm for datacenter networks (Guo *et al.*, 2015). The focus is on two interesting results in the bargaining game approach: 1) to ensure that the dual variables (λ_m^{E} and λ_m^{I}) of each server m can be updated independently with local information. Meanwhile, the iteration of each rate ($r_{i,j}$) only requires the bandwidth information of the server hosting Virtual Machine $(VM)_j$ and the server hosting VM_j. This motivated the design of the Offline Cooperative Algorithm, which can obtain the optimal rate in a distributed cooperative manner.

Update Weight and Base Bandwidth: the model first considers how to update the base bandwidth and weight of each VM-pair when a VM establishes a connection or disconnects with another VM, the number of its connected VMs changes. Thus, the base bandwidth and weight of existing VM-pairs are supposed to be changed accordingly. The function was triggered by the change of connections to a VM: When it connects to or disconnects from another VM_j, it obtains the weight and base bandwidth of VM_x, and then updates the weight and base bandwidth of all its VM pairs. To reduce the overhead of recalculation, each VM-pair can be managed by its source VM. The outputs of **Algorithm 1**, i.e., the base bandwidth and weight, are used as input for calculating the optimal rate allocation for VM-pairs.

Distributed Iteration Process: Since the calculation of rate $(r_{i,j})$ for each VM-pair involves two servers, it can be executed either on the source server or the destination server. The iteration process of each server is presented in Algorithm 1. The model let $r_{Pm}^{E} = \sum_{i=1}^{N}(w_{m,i}\sum_{j=1}^{N} r_{i,j}) - C_m$ and $r_{Pm}^{1} = \sum_{i=1}^{N}(w_{m,i}\sum_{j=1}^{N}) r_{i,j}) - C_m$ represent the allocated egress and ingress bandwidth of server m. For the VM-pair whose bandwidth demand $(D_{i,j})$ is less than the base bandwidth $(B_{i,j})$, the algorithm sets $(r_{i,j})$ to be $(D_{i,j})$. For other VM-pairs that belong to J, the iteration of $(r_{i,j})$ begins with the initial lower bound $L_{i,j}$. During each iteration, server m updates its dual variables $(\lambda_m^{\mathrm{E}}\ and\ \lambda_m^{\mathrm{I}})$ by calculating its residual bandwidth $(r_{Pm}^{\mathrm{E}}\ and\ r_{Pm}^{\mathrm{I}})$. For all the VM-pairs $(\boldsymbol{r_{i,j}})$ with VM i on server m, the server should request the other dual variable λ_j^i from server $l(j \in V_i)$. The rate of the VM-pair is then updated based on these two dual variables. Also, the model makes a judgment in case the rate exceeds the upper-bound bandwidth $U_{i,j}$. The step size is chosen according to the step-size rules and updated locally within each iteration.

Stopping Rules: Note that the convergence speed of $(r_{i,j})$ depends on the step-size ξ and the gradient of the dual variables, which are exactly the residual bandwidth of serve m and l. Since the step-size and residual bandwidth are both maximal initially and decrease as the algorithm performs, the dual variables will quickly converge to approximate optimal values and then slowly approach the optimal values. Hence, the algorithm can finish within an acceptable number of steps if it does not need strict optimal rates for all VM pairs. The model can define two stopping rules to balance the trade-off between algorithm overhead and precision.

- *Step-based mechanism*: Use *S* as the total iteration step in the convergence process. As the execution time of the algorithm is in proportion to the iteration rounds, one can manage the cost of the algorithm by adjusting the iteration rounds.
- *Precision-based mechanism*: Stop the convergence process if the variation of each $(r_{i,j})$ is less than within two consecutive iterations.

When the iteration stops with the control of the stopping rules, outputs can be applied in the hypervisors by enforcing a bandwidth limitation for each VM-pair. The allocated bandwidth, which is always less than (or equal to) the bandwidth demand, will be fully utilized due to the aggressiveness of transport-layer flows.

3.3.3 Enhanced Dynamic Bandwidth Allocation Algorithm

Based on Figure 1, in the proposed EDBA algorithm, relay nodes are classified and prioritized using the PSO. The relay nodes are represented by s.

In which, **equation (1) and (2)** are redesigned to define priority limits for all relay nodes as shown in **equation (3)**:

$$X_s^L \leq X_s \leq \cdots \leq X_s^H \ \forall_s \quad (3)$$

$$s = 1, 2, 3, \ldots, 6 \ X_s \in R$$

Whereby X represents the priority given to each relay node depending on the workflow procedures. L represents the least prioritized relay node and H represents the highest prioritized relay node. R represents that priority for each subnet is real.

This study predetermines the priority of relay nodes in which:

If min s ∈ {1, 2… 6} = 1 (4)

Relay node 1 becomes the least prioritized.

Else if max s ∈ {1, 2… 6} = 6 (5)

Relay node 6 becomes the highest prioritized.

Since each relay node, s, contains child nodes within it. Based on **equation (3)**, this study prioritizes and schedules each child node, d, in the subnets using weighing factor numbers as shown in **equation (4)**.

$$X_d^L \leq X_s \leq \cdots \leq X_d^H \ \forall_d \quad (6)$$

$$d = 1, 2, 3, \ldots, \infty \ X_d \in R$$

X represents the priority given to each child node. L represents the least prioritized child node in a relay node and H represents the highest prioritized child node in a relay. R represents that priority for each child node in the relay node is real.

This study predetermines the priority of child nodes in which:

If min d ∈ {1, 2, 3…∞} = 1 (7)

Child node 1 becomes the highest prioritized.

If max d ∈ {1, 2, 3…∞} = ∞ (8)

Child node ∞ becomes least prioritized.

In computing total bandwidth, b, used in the network, the bandwidth for relay nodes was sum up as shown in **equation (9)**.

$$b_6 + b_5 + b_4 + b_3 + b_2 + b_1 = \sum_{i=1}^{n} b_i \quad (9)$$

Considering that, not all packets would successfully be transmitted in the network. This study, therefore, computes packet loss using **equation (10)**.

$$\text{Packet loss} = \left\{\frac{\sum_{t_1}^{t_n} \text{dropped packet}}{\sum_{t_1}^{t_n} \text{sent packet}}\right\} * 100 \quad (10)$$

In which $t_1$represents the initial dispatch time of the first packet in each child node and t_n represents the final dispatch time of the last packet in each child node.

This study proposed to apply the OCA to dynamically enforce rate allocation relying on the requiring the traffic demand of each node. Based on OCA, the design is comprised of two components: the bandwidth allocation service on each server and the communication protocols between servers. The service is responsible for updating the base bandwidth/weight of VM-pairs and calculating the rates to be allocated. Meanwhile, when the connections of VMs change, it updates the base bandwidth and the weight of VM-pairs (Servers) as depicted in **Algorithm 1**.

Algorithm 1: Offline Cooperative Algorithm (OCA)

Input:

Define the number of relay nodes

The step-size: ξ

Server bandwidth capacity: C_m,$\forall m \in \mathcal{M}$

Bandwidth demand matrix: $\left[D_{i,j}\right]$ N x N, $\forall_i \in \mathbf{N}$

VM placement: $\left[w_{m,i}\right]$ M x N, $\forall_m \in \mathcal{M}, \forall_i \in \mathbf{N}$

The total number of iteration rounds: S

The gap between two consecutive iterations: Δ

Output:

1: **Run equations (3), (4), and (5)** to define priority limits for all relay nodes.

2: **Run equations (6), (7), and (8)** to prioritizes and schedule each child node.

3: **Run equation (9)** to compute the total bandwidth, which is then used below.

4. **while** $s < S$ or $r_{i,j}^{(s)} - r_{i,j}^{s-1} > \Delta$ **do**

5: Update allocated bandwidth from **equation (9)**

$r_{Pm}^{E} = \sum_i w_{m,i}\, r_i^{E}, r_{P_m}^{\mathrm{I}} = \sum_i w_{m,i}\, r_i^{\mathrm{I}}$

Update dual variables as (3)

$\lambda_m^E = \max(0, \lambda_m^E - \xi(C_m - r_{P_m}^{\mathrm{E}}))$

$\lambda_m^{\mathrm{I}} = \max(0, \lambda_m^{\mathrm{I}} - \xi(C_m - r_{P_m}^{\mathrm{I}}))$

for all $r_{i,j}, i \in V_m$ **do**

Update $\lambda^E = \lambda_m^E$

Obtain $\lambda^{\mathrm{I}} = \lambda_\iota^{\mathrm{I}}$ from server $\iota, j \in V_\iota$

if $\frac{K_{i,j}}{\lambda^E + \lambda^{\mathrm{I}}} > U_{i,j} - L_{i,j}$ **then**

$r_{i,j}^{(s)} = U_{i,j}$

else

$r_{i,j}^{(s)} = L_{i,j} + \frac{K_{i,j}}{\lambda^E + \lambda^{\mathrm{I}}}$

end if

end for

Update step-size: ξ

Update iteration round: $s + s = 1$

end while

Run equation (10) to compute packet loss.

3.4. Summary

In this chapter, this study has presented and discussed the intelligent home network architecture and proposed system algorithm. This study proposed a star topology for the design of our architecture. Furthermore, this study showed an increase in the number of devices that are supported by each subnet. This study has modeled the proposed system to accommodate the addition of more devices, services, and technological evolutions. These changes often result in competition and depletion of the available network resources and services. Apart from that, the changes lead to the overloading of network traffic. In the EDBA algorithm, this study used the principle of optimality which does sub-optimization to/of each subnet. The proposed EDBA algorithm has established three performance metrics: average throughput, average delay, and energy consumption. In the next chapter, this study presents and discusses the implementation of the EDBA algorithm which has been introduced and designed in this chapter. This study presents the used simulation tool, NS-2, and discusses the features that make it suitable for utilization in this study. Lastly, this study presents the environment of the used simulation tool.

CHAPTER 4 : IMPLEMENTATION & EXPERIMENTATION

4.1. Introduction

In this chapter, this study presents a discussion on the implementation and experimental evaluations of the proposed Enhanced Dynamic Bandwidth Allocation Algorithm (EDBA). The configuration of the proposed intelligent home network is comprised of connected multiple different layer switches, routers, devices, and communication links. In general, simulation is a helpful ingredient before doing the real network implementation on the ground. The network simulation tools allow researchers to examine the behavior of networks under various conditions, thereby, allowing the modification of the network models to accomplish the set objectives. This study has implemented the proposed EDBA using NS-2 version 2.35. The NS-2 is an object-oriented discrete-event network simulation tool targeted by many networking researchers, which was originally developed in 1989 (Bakare & Enoch, 2019; Balfaqih, 2020; Kumar & Bansal, 2020; Nayyar, 2018) by Marc Greis. Also disused the configuration and design IHN

The rest of this chapter is structured as follows: In Section 4.2, this study provides and briefly discusses the existing network simulation tools. In Section 4.3, this study presents and discusses the comparison between network simulation tools. In Section 4.4, this study provides into detail, the NS-2 simulation environment and also discusses its benefits and drawbacks. In Section 4.5, this study deeply discusses how the proposed algorithm was implemented and evaluated. Lastly, in Section 4.6, this study provides a conclusion to the chapter.

4.2. Overview of Network Simulation Tools

Different network simulation tools allow researchers to perform experimental evaluations of applications and protocols under varying network configurations. These tools further allow users to experiment and implement newly designed models in a controlled manner. Therefore, in this Section, this study provides a discussion on different most used network simulation tools within the research niche. This study also presents a comparison of the discussed network simulators thereby, giving the reason behind our choice of NS-2 2.35 as the simulation tool as opposed to others.

4.2.1. IHN Configuration and Design

We combine largely two kinds of technology for design and realization of NS-2 based on IHN which is proposed in this study. We monitor the amount of network come in the house hold on NS-2 by communication with IHN to manage which device need more Bandwidth at a certain time.

4.2.2. Network Simulator–2 (NS-2)

This is a discrete-event network simulation tool built by the Virtual Inter-Network Testbed (VINT) research group from the University of California. NS-2 simulates network protocols with varying network architectures and it can simulate both wired and wireless networks. This tool uses C++ in carrying out core communication functions and uses an Object-oriented Tool command language (OTcl) to create simulation interfaces. NS-2 includes a network animator whose interface consists of control features that allow users to play, forward, pause, and stop the simulation. NS-2 further provides substantial support for the simulation of protocols such as Transmission Control Protocol (TCP), User Datagram Protocol (UDP), File Transfer Protocol (FTP), Hypertext Transfer Protocol (HTTP), etc., over wired and wireless (local and satellite) networks. Moreover, NS-2 is one very popular network simulation tool used by many researchers. This is because it offers online support wherein many users could assist each other with the simulation projects. The interface of the NS-2 is given in Figure 4.1.

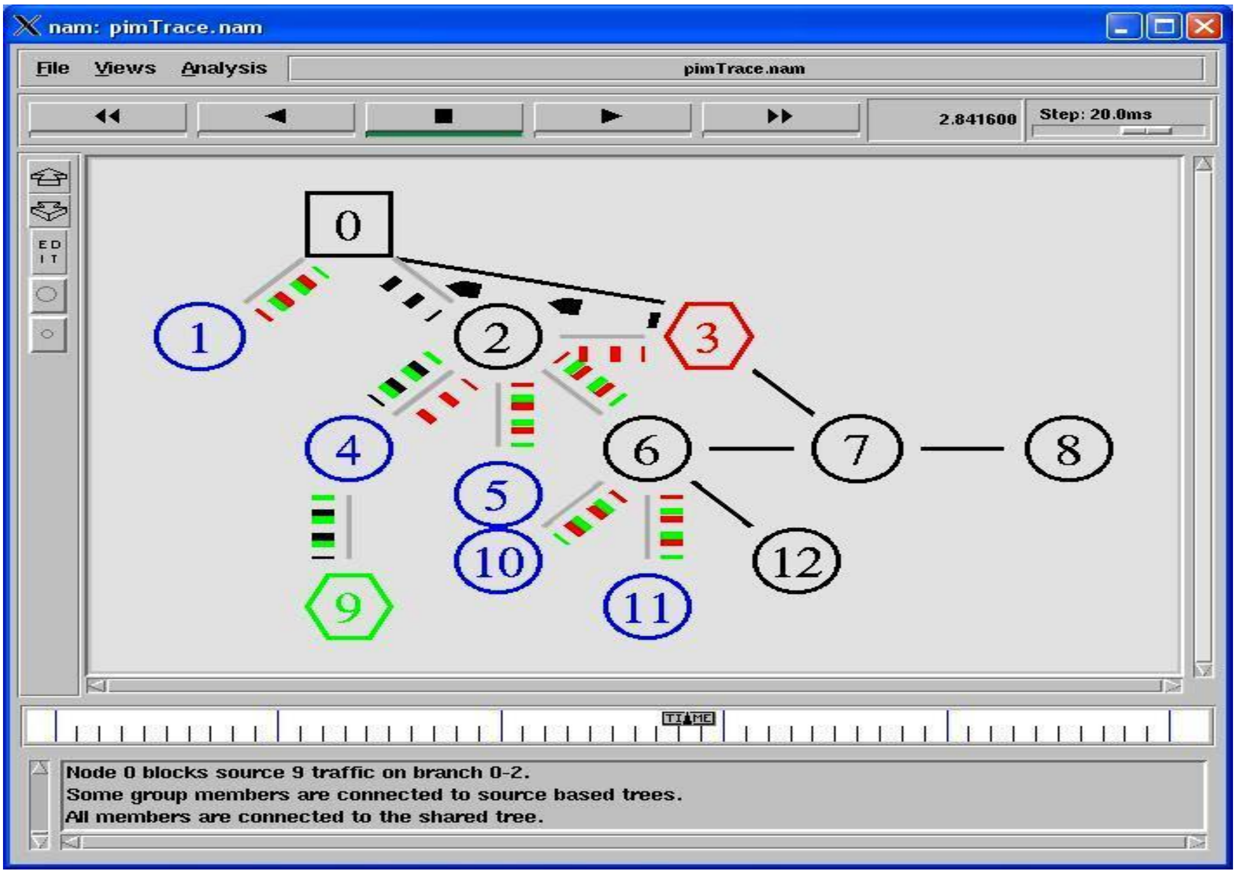

Figure 3:2: NS-2 simulation interface

4.2.3. Optical Micro-Networks (OMNET) ++

This is a simulation tool used in observing the behavior and performance of both wired and wireless networks. The main simulation frameworks for OMNET++ include INET, INETMANET, MiXim, and Castalia. INET Framework is considered the standard protocol model library of OMNeT++. It contains models for the Internet stack (TCP, UDP, IPv4, IPv6, OSPF, BGP, etc.), wired and wireless link layer protocols (Ethernet, PPP, IEEE 802.11, etc), support for mobility, MANET protocols, DiffServ, MPLS with LDP and RSVP-TE signaling, several application models, etc. MiXiM is an OMNeT++ modeling framework created for mobile and fixed wireless networks (wireless sensor networks, body area networks, ad-hoc networks, vehicular networks, etc.). It concentrates on the lower layers of the protocol stack and offers detailed models of radio wave propagation, interference estimation, radio transceiver power consumption, and wireless MAC protocols. Castalia is a simulator for Wireless Sensor Networks (WSN), Body Area Networks (BAN), and generally networks of low-power embedded devices. The benefits of this tool include the following; It is used for building large network communities and it is platform-independent. However, creating the simulations using this tool requires more effort as compared to other tools. Just like NS-2, the OMNET++ interface consists of control features that allow users to play, forward, fast forward, stop, and record the simulations. This study has given the OMNET++ interface in Figure 4.2.

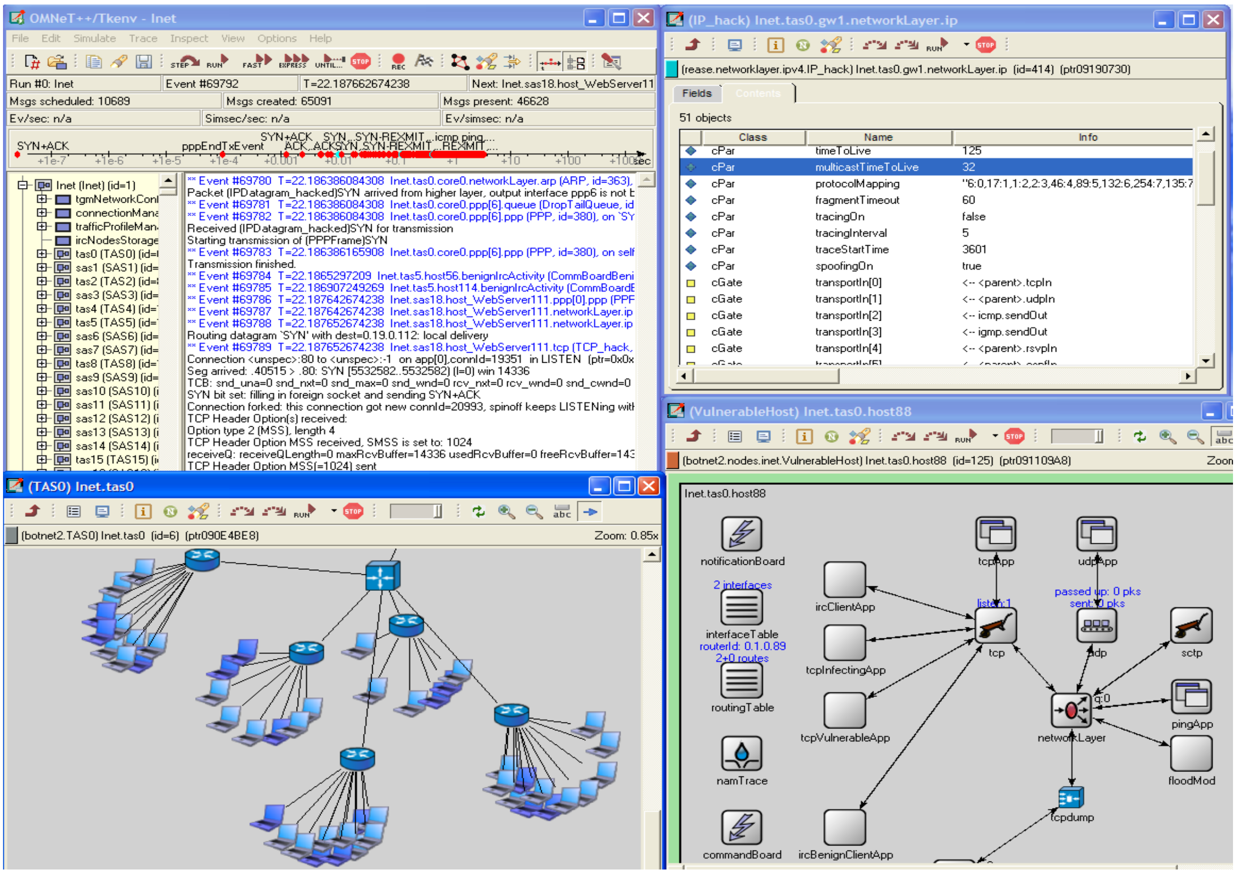

Figure 3:3: OMNeT++ simulation interface

4.2.4. Optimum Network Performance (OPNET)

This is a network simulation tool used in simulating the behavior and performance of both wired and wireless networks. OPNET has satellite communication as a building component. OPNET uses hierarchical modeling which is divided into three domains, namely; network, node, and process models. OPNET further allows the possibility to simulate complete heterogeneous networks with various protocols that run on various layers of the OSI model. OPNET uses C and C++ source code in assembling high-level user interfaces. Furthermore, OPNET categorizes node mobility into three different ways, namely; mobility trajectory, mobility vector, and manipulation of node coordinates. OPNET is commercial software and expensive to purchase. A single purchase permits usage by a single user. A licensed OPNET simulator cannot be distributed among computers in a network. This study shows the interface of OPNET in Figure 4.3.

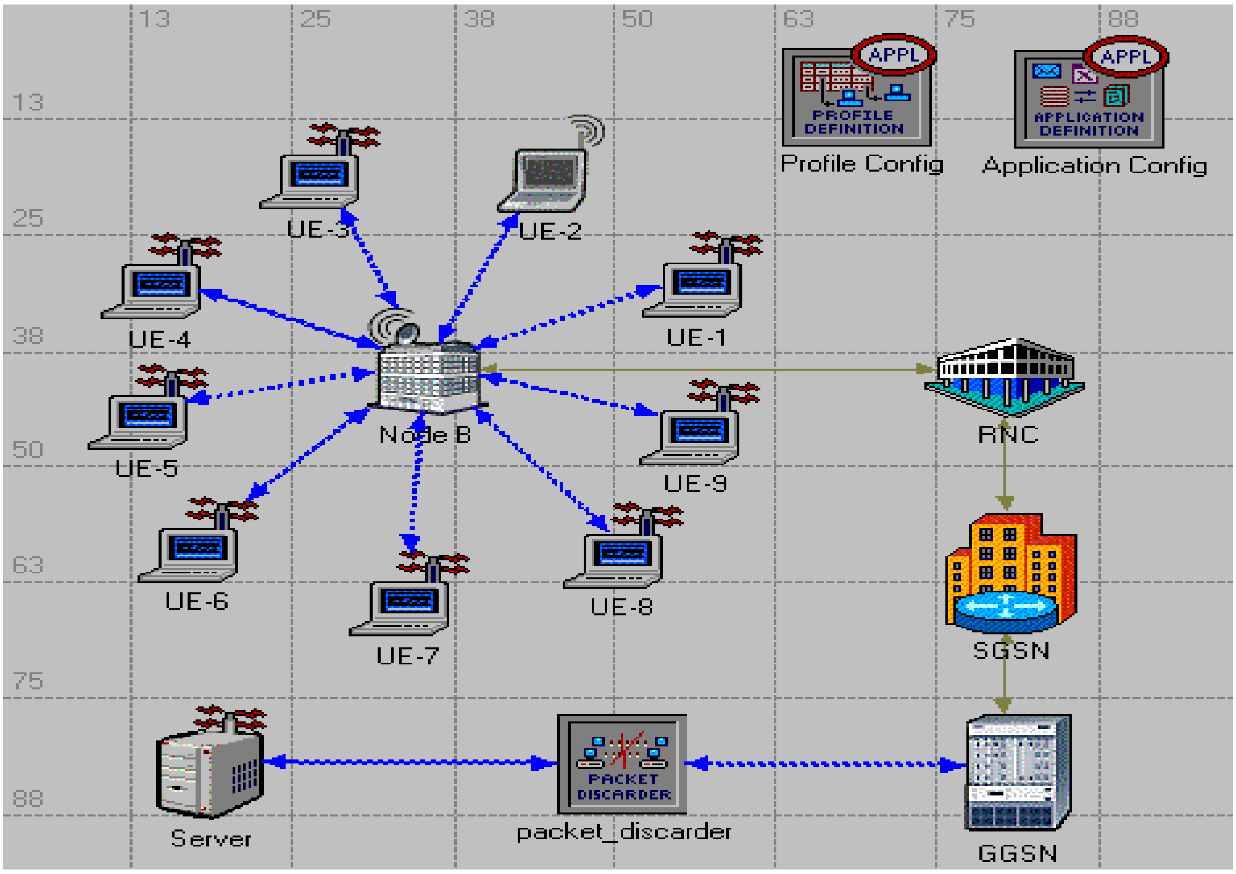

Figure 3:4: OPNET simulation interface

4.2.5. MATrix LABoratory (MATLAB)

MATLAB is a programming language that facilitates the manipulation of the matrix, plotting of functions and data, implementation of new algorithms as well as the improvement of existing ones, and the creation of simple user interfaces. This is a multi-paradigm numerical computing environment and fourth-generation programming language developed by Cleve Moler in the 1970s. MATLAB is developed and maintained by MathWorks to allow matrix manipulations, the creation of user interface, and the implementation of new and existing algorithms, and the plotting of data results. Many researchers use this tool to analyze and design their systems for purposes of machine learning, communications, design control, processing of signals, computer visions, processing of images, robotics, as well as computational finances. MATLAB uses C, C++, Java, FORTRAN, and Python. MATLAB uses Multi Processing Algebra Data (MuPAD) symbolic engine in allowing access to computing capabilities. Equally, Simulink adds graphical multi-domain simulation. MATLAB is used in industries, academic, and research institutions and runs on cross-platforms like Windows, Linux, and Mac OS X. However, this tool was designed to solve numerical problems. This study shows the interface of the MATLAB simulator in Figure 4.4.

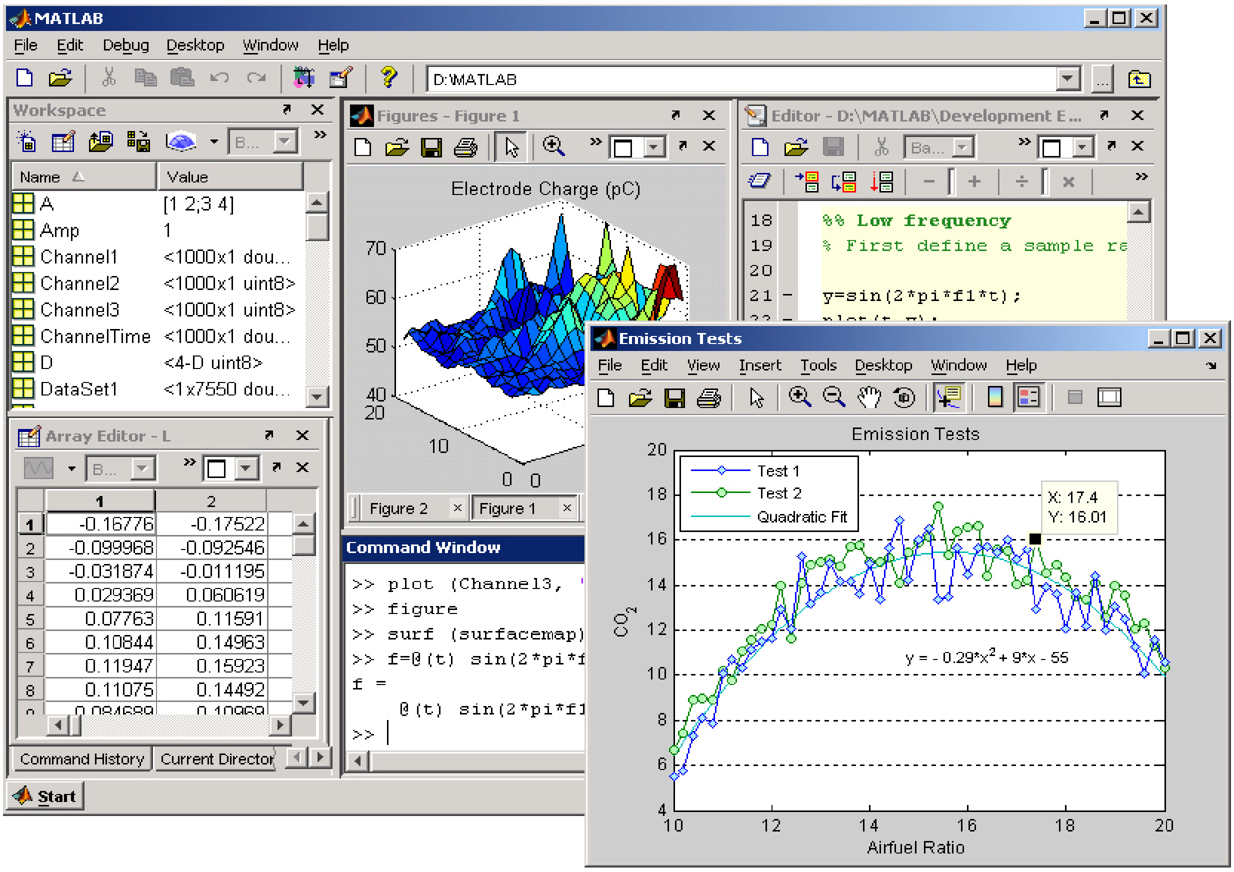

Figure 3:5: MATLAB simulation interface

4.2.6. QualNet

QualNet is also a network simulation tool mainly designed to plan, test, and also to model the behavior of large real computer networks (Gong *et al.*, 2014). Consequently, this tool provides a comprehensive environment aiding in the design of protocols, creating and animating real computer network scenarios, and analyzing the network performances before implementation. This tool is comprised of various components, namely; QualNet Architect - graphical scenario design and visualization tool; QualNet Analyser - the tool used to graphically display data collected in the simulations; QualNet Packet Trace - tool meant to provide a visual representation of packet trace files generated in the simulations; QualNet File Editor - the tool used for editing text, and QualNet Command Line Interface - used to access the simulator.

Additionally, this tool enables users to design new models and facilitate the optimization of new and existing models. Further, QualNet supports the design of large either wireless or wired networks and can be simulated using pre-configured or user-designed models. As its benefit, QualNet supports real-time speed, facilitating users to run multiple simulations simultaneously and in a very short period. However, this tool is not easy to install and very complex to use compared to others. This study shows the simulation interface of QualNet in Figure 4.5.

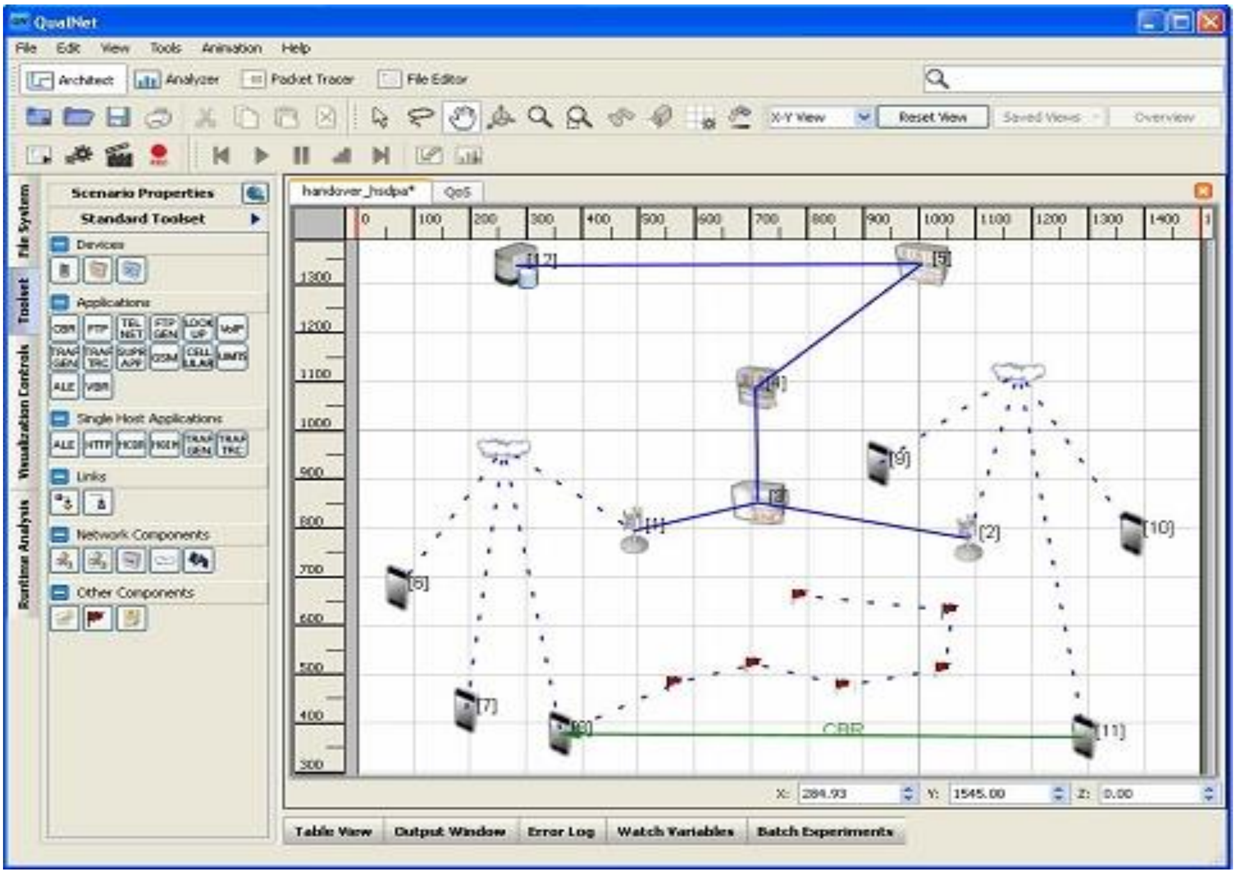

Figure 3:6: QualNet simulation interface

4.3. Comparison of Network Simulation Tools

This study provides characteristics of the discussed network simulators in Table II. NS-2 has been the tool to implement the proposed model. This is because NS-2 is much favorable for simulating large networks as shown in the table. Further, it is a free source and scalable. Moreover, setting up and using NS-2 is very easy especially for beginners. Consequently, many academic research institutions use them. Therefore, NS-2 was our preference as it allows simulation of both wired and wireless networks which were building components of the architecture proposed in this study. Equally, it required not to be purchased compared to other tools such as OPNET which was also our preference. The purchasing of a licensed OPNET simulator proved futile. A licensed OPNET possesses all the functionalities for simulating any research model. Apart from that, NS2 offers online support to its users, which enables users to help each other with their NS-2 simulation project challenges. Moreover, this tool is GUI-based and supports the simulations of IEEE 802.11 models and, thus, supports the simulation of different routing protocols, queuing protocols, and multicast protocols for both wired and wireless networks.

Table II: Comparison of Network Simulation Tools

Feature	NS-2	OMNET++	OPNET	MATLAB	QualNet
Ease of setup	Easy	Moderate	Moderate	Moderate	Hard
Ease of use	Moderate	Moderate	Moderate	Hard	Hard
Scalability	Poor	High	High	Poor	High
Programming Language	C++	C++	C & C++	C,C++, Java	C
Larger Communication Networks	No	Yes	Yes	Yes	Yes
Continuous development	Yes	Yes	Yes	Yes	Yes
GUI	Yes	Yes	Yes	Yes	Yes
Obstacles	No	No	Yes	Yes	Yes
Accessibility	Yes	Yes	No	Yes	No

4.4. Network Simulator-2 (NS-2) Environment

In this section, this study provides insight and briefly discusses the NS-2 simulation environment. This includes its basic architecture, directory structure, class hierarchy, and how the trace files are formatted.

4.4.1. NS-2 Basic Architecture

As been discussed in Section 4.2.1, NS-2 comprises two programming languages, namely; C++ and OTcl. The reasoning behind these two languages in-cooperated is that C++ runs faster but takes time to code compared to OTcl, which runs slowly but very easily, and faster to code. In general, the protocol implementation requires powerful languages such as C++ for faster per-packet processing, meanwhile, the use of script language makes the writing and change of simulation configuration faster to adjust with desired parameters (Karim & Ashraful, 2009). Apart from that, OTcl is used to design the network structure and, thus, it is used to build and configure network structures and different topologies. On the other hand, C++ forms part of the kernel of NS-2, making it very important in NS-2 simulations. This is because C++ carries out the core communication functions of simulation objects including the implementation of the protocol designs. This study shows NS-2 basic architecture in Figure 4.6.

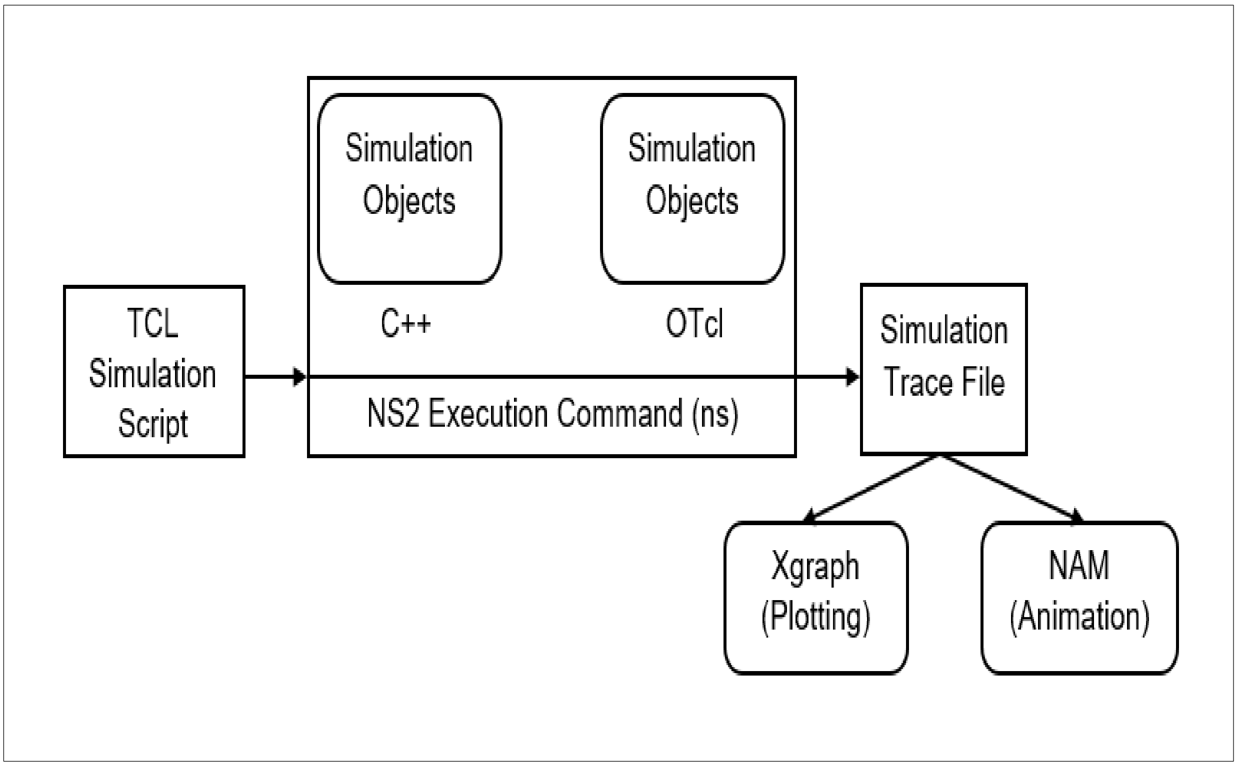

Figure 3:7: NS-2 basic architecture

4.4.2. NS-2 Directory Structure

Generally, all the versions of NS-2 are normally installed from one package. This one package configuration simplifies the installation process of NS-2. In this work, this study used NS-2 version 2.35 from a package mostly known as ns-allinone-2.35. This study shows the NS-2 directory structure of the package that was installed and run on Linux Ubuntu 16.04.5 LTS. This study runs the OS on the Virtual Machine (VM). The simulation modules are defined and found in the subdirectory called NS2.35. The directories for TCL and C++ source code are also found in the NS2.35 directories. Apart from that, the commands “. /configure, make, and make install” must be executed in the terminal to compile and run C++ source codes. This study shows a typical NS-2 directory structure in Figure 4.7.

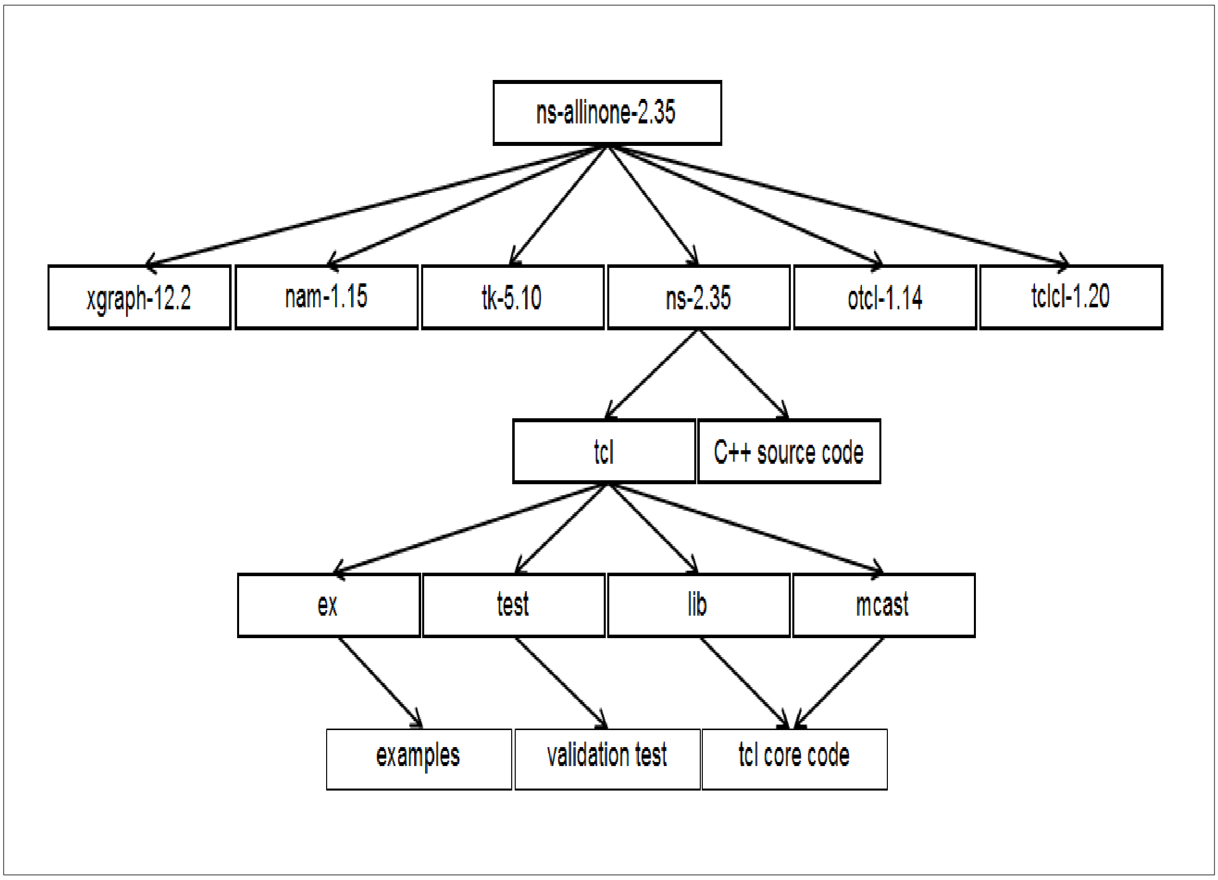

Figure 3:8: NS-2 directory structure

4.4.3. NS-2 Class Hierarchy

As discussed in Section 4.2.1, NS-2 uses C++ in carrying out core communication functions and uses Object-oriented Tool command language (OTcl) to create simulation interfaces. Therefore, this class hierarchy represents and defines the relationships between C++ classes. As shown in Figure 4.8 below, the NsObject and its sub-classes extend the super-class called TclObject and thus, it is always the root class. This super-class has many sub-class and its responsibility is to provide basic functionality such as providing the ability to retain and release objects.

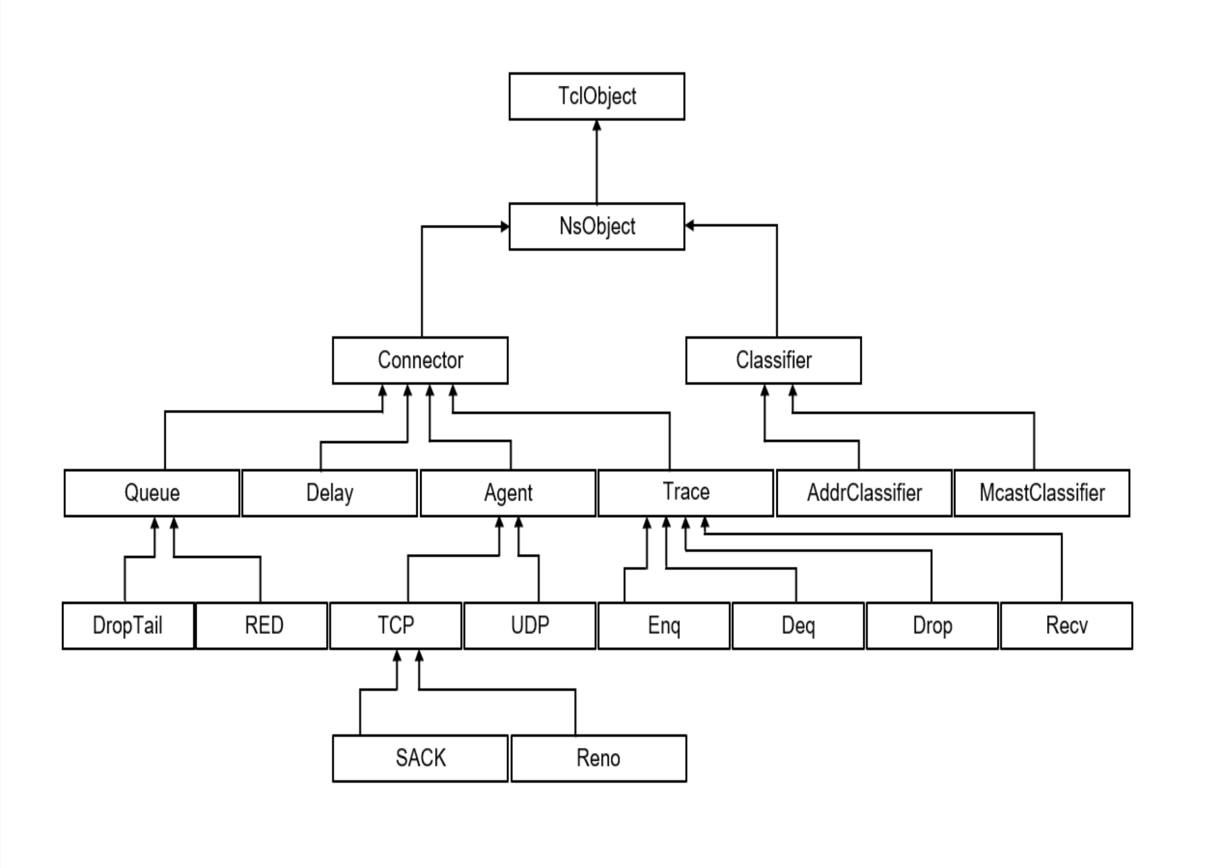

Figure 3:9: NS-2 class hierarchy

4.4.4. Trace Files Format

In most cases, trace files are files written by the application and contain coverage information or overall network information that has been transmitted between nodes. The trace file format for wireless networks is defined in the cmu-trace.cc file in the NS2.35 directories. However, as mentioned above, these traces are auto-generated or created during the simulation. This study has written AWK scripts for interpreting or analyzing the collected data by the trace files. This helped us to get useful statistics such as the number of packets received and dropped and further calculate network throughput, packet loss, and more. The AWK is a programming language designed to process the content of text files. The AWK scripts are given in Appendix C. This study shows the basic format of trace files in Table III.

Table III: Trace files format

Type Identifier	Time	Source Node	Dest Node	Name	Size	Flags	ID	Source Address	Dest Address	Seq Number	Uniq ID

In its most general form, trace files are comprised of twelve fields and these are discussed as follows:

- Type Identifier – normally represented by **+** to indicate enque operations, **-** to indicate deque operations, **r** to indicate receive events, **d** to indicate drop events, and **c** to indicates packet collisions at the MAC level.
- Time - indicates the simulation time at which each event occurred. This is always measured in seconds.
- Source Node - the node that created and sent the packet.
- Destination Node - the node that received the packet.
- Name - a descriptive name for the type of packet transmitted.
- Size - the size of the packet transmitted.
- Flags - flags that are defined in the flags array in the trace.cc file.
- ID – the identifier for the IP flow.
- Source and Destination addresses – packet's source and destination node addresses, respectively.
- Seq - sequence number.
- Uniq ID - unique packet identifier assigned to each packet created.

4.4.5. NS-2 Configuration

As mentioned in Section 4.3.1, NS-2 is a discrete-event network simulator and is platform-independent. Consequently, this tool can be installed and used in any OS such as Windows, Mac, UNIX OS, etc. Though, it was specifically designed and developed for UNIX-based OS. Consequently, this makes it easy to install on UNIX-based systems as compared to other environments. It is very crucial to be aware that, NS-2 is quite big and needs a hard disk with space of approximately 300 MB for installation. Herein, this study provides the steps to be followed to configure and install NS-2 version 2.35 on Ubuntu 18.04.1 LTS.

1. Download the ns-allinone-2.35.tar.gz, which is a free source package from the Internet.
2. Put the downloaded package in the /home/Documents/ directory and thereafter extract it by right-clicking it and select the command of extracting, otherwise, open the terminal and change to /home/Documents/. Type the following command:
   - ✓ *tar –zxvf ns-allinone-2.35.tar.gz*

3. Open the terminal and type the following commands below to update the OS with the latest software and updates:
   - ✓ *sudo apt-get update*
   - ✓ *sudo apt-get dist-upgrade*
   - ✓ *sudo apt-get update*
4. Thereafter, this study has to install the essential packages required by NS-2 by typing the following commands in the terminal:
   - ✓ *sudo apt-get install build-essential autoconf automake*
   - ✓ *sudo apt-get install tcl8.5-dev tk8.5-dev*
   - ✓ *sudo apt-get install perl xgraph libxt-dev libx11-dev libxmu-dev*
5. Change the directory path in the terminal to /home/Documents/ns-allinone-2.35 by typing the command below:
   - ✓ *cd /home/Documents/ns-allinone-2.35*
6. Open the following folders (ns-allinone-2.35, NS2.35, and linkstate) and then open the ls.h file. In this file, go to line 137 and replace erase with this->erase. Save the file after editing.
7. Go back to the terminal and type the command below to start installing NS-2:
   - ✓ *./install*
8. Once the installation is successful, the message below will be displayed in the terminal: "*After these steps, you can now run the ns validation suite with cd NS2.35; ./validate"*
9. This study then has to set up some environment variables which is achieved by opening a new terminal and type the command below:
   - ✓ *sudo gedit .bashrc*.
10. Once the bashrc is opened for editing, at the end of it, paste the following few lines:

```
#LD_LIBRARY_PATH
OTCL_LIB=/home/Documents/ns-allinone-2.35/otcl-1.14/
NS2_LIB=/ home/Documents/ns-allinone-2.35/lib/
USR_Local_LIB=/usr/local/lib/
export
LD_LIBRARY=$LD_LIBRARY_PATH:$OTCL_LIB:$NS2_LIB:$USR_Local_LIB
#TCL_LIBRARY
TCL_LIB=/ home/Documents/ns-allinone-2.35/tcl8.5.10/library/
USR_LIB=/usr/lib/
export TCL_LIBRARY=$TCL_LIBRARY:$TCL_LIB:$USR_LIB
```

```
#PATH
XGRAPH=/home/Documents     /ns-allinone-2.35/xgraph-12.2/:/home/Documents/ns-allinone-2.35/bin/:/home/Documents/ns-allinone-2.35/tcl8.5.10/unix/:/home/Documents/ns-allinone-2.35/tk8.5.10/unix/
NS=/ home/Documents/ns-allinone-2.35/NS2.35/
NAM=/home/Documents /ns-allinone-2.35/nam-1.15/
export PATH=$PATH:$XGRAPH:$NS:$NAM
```

11. Save the file and go back to the terminal to reload the bash file by typing the command below:
    - ✓ source ~/.bashrc
12. Thereafter, the installation needs to be validated by typing the command below:
    - ✓ ./validate.
13. Lastly, reboot the machine; open the terminal, and type ns, if the message shows "%" sign, it means the installation was successful. Therefore, open the Network Animator (NAM) by type nam in the terminal. The NAM will be opened as illustrated in Figure 4.9.

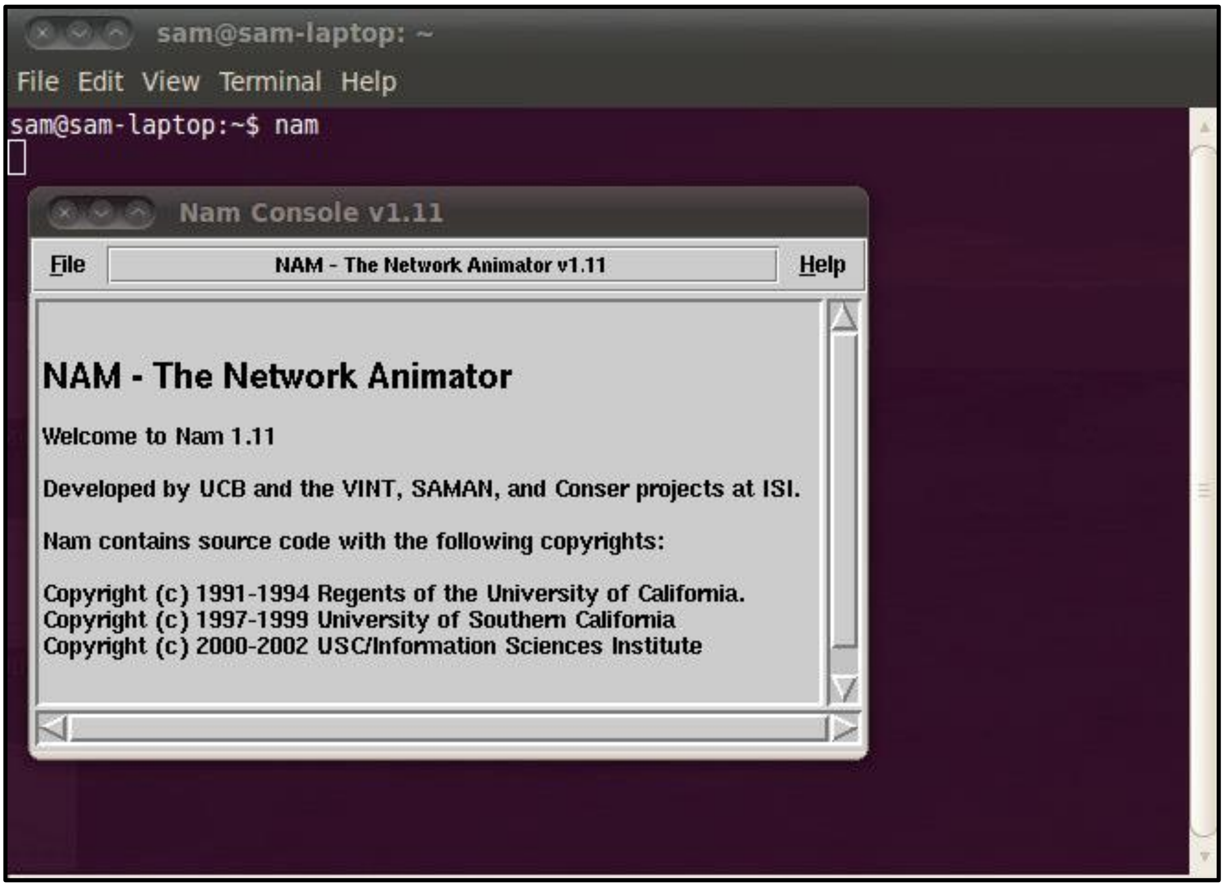

Figure 3:10: Sample NAM interface

4.5. Simulation of Enhanced Dynamic Bandwidth Allocation (EDBA) Algorithm

The main purpose of the simulation is to observe the improvement that EDBA brings in the management and allocation of bandwidth dynamically. This section provides details on how to configure NS-2 on a Linux machine. This section also presents the simulation environment and discusses the simulation topology as configured in NS-2. The performance evaluation of the proposed bandwidth allocation scheme has been done through extensive simulations in NS-2. The simulations were done by using a model of IEEE 802.16g developed in NS-2 version 2.35 (NS-2.35). In the simulations, this study decided to have a minimum number of 8 nodes in total. These nodes are randomly placed in the network and assigned numbers from 0-7. In the simulation, there are no mobile nodes. However, this study could have applied a random waypoint if there were mobile nodes wherein each node would room around until its mobility-time expires. Equally, this study considered 2 relay nodes each labeled “Relay Node” with a 600-meter coverage area. However, the relay nodes were randomly placed in the network. Furthermore, this study had 6 client nodes given names such as Cooker, Coffee maker, Clothing machine, Lights, etc. These client nodes were randomly placed and each connected to a particular relay node. Moreover, the proposed EDBA algorithm was implemented at the distribution layer of the network and that is the root node 0. This layout has been chosen to make the simulations with multiple vehicles to make a more realistic physical layer response for different vehicle loads. The TCL script given in Appendix B was written to specify the network topology of the proposed EDBA.

Each simulation ran for 300 seconds to ensure enough results. Also, this study ensured that the simulations were performed multiple times to ensure quality and convincing results. The reason behind this is that NS2 is not scalable, and thus, the results might not be reliable or scalable enough. The results were recorded in trace files and used those trace files to calculate average throughput, delay, and energy consumption. On the other hand, this study used R Programming to graphically display the results which were obtained and analyzed. Furthermore, this study created AWK programs, compiled and run them using the simulation tool. The purpose of the AWK programs was to calculate the average throughput, delay, and energy consumption. Apart from that, packet losses were observed in the NAM during the experimental evaluations. Lastly, this study evaluated the proposed EDBA algorithm against DQBA and DDA.

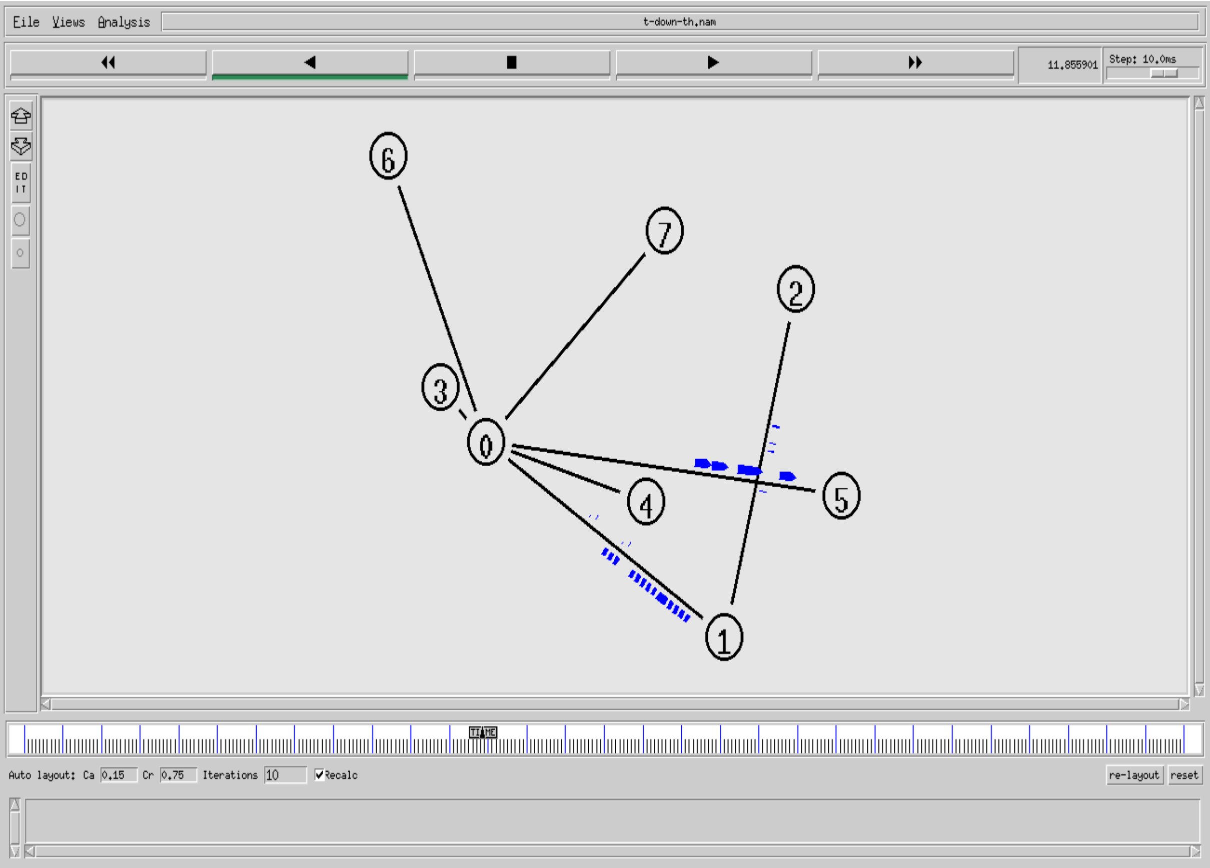

Figure 3:11: Proposed IHN simulation scenario

In the proposed algorithm, this study integrated two traditional algorithms, namely; Offline Cooperative Algorithm and PSO. Offline Cooperative Algorithm bases itself on the game-theoretical framework to any bandwidth allocation problem, and thus, it has been applied in the proposed algorithm to be responsible for the allocation of bandwidth in the network. Furthermore, in this study, PSO is applied as an optimization algorithm. This algorithm is used to define priority limits for all nodes in the network. The primary reason is to give different priorities to different relay nodes to assure a certain level of performance during client requests in the network. Looking at Figure 4.10, node 0 is the root node. The root node is special and acts as the central coordinator of the network. Its purpose is to carry out the task of gateway nodes, and thus, it connects a home or enterprise network to the internet. Therefore, the other nodes are relay nodes. Each relay node receives the external requests from the internet or other networks and forwards them to the child nodes. Also, it forwards node join requests to its root node. These nodes perform dynamic slot assignments on receipt of the traffic demands from their children nodes. As mentioned in chapter 3, the child nodes are the end-points of the network responsible for carrying out the task of receiving control packets, generating node join requests, and sending bandwidth demands to their parent nodes. As shown in Figure 4.10, this research logically divides the proposed Intelligent Home Network into 1-hop clusters. As the Figure shows, all the clusters exhibit a parent-child relationship and have identical behavior. Consequently, this study decided to introduce the process of relay nodes optimization through PSO, which solves the bottleneck with bandwidth allocation. Moreover, this study used LTEQueue/ULAirQueue/DLAirQueue to give different priorities to child nodes based on the priorities assigned to nodes by the PSO algorithm. This aided in improving QoS. On the other hand, this study used the bandwidth of 2 Mbps with a delay of 1.8 ms, and a data rate of 32 Kbps. Also, this study used ErrorModel which is a packet sink that keeps on observing packet losses. In conclusion, the simulations were run for 300 seconds and executed several times to obtain the most accurate and convincing results. The simulation parameters and values used in the simulations are presented in Table IV.

Table IV: Simulation Parameters

Experimental Parameters	
Data rate in the network	32 Kbps
Delay in wired links	1.8 ms
IGW hello packet broadcast interval	200 ms
IP queue scheduler	Strict priority
No. of source nodes	0 and 1
Number of Nodes (Min.)	8
Packet size	512 bytes
Propagation channel frequency	2.4 GHz
Propagation model	Two-ray ground reflection model
Queue Type	LTEQueue/ULAirQueue/DLAirQueue
Raw channel bandwidth	2 Mbps
Routing protocol	Fixed routing protocol
Simulation area	1800m X 840m
Simulation duration	300 s
Wired network bandwidth	100 Mbps

The implementation is not complex, hence the choice has been to implement it directly on the NS-2 configuration script file with no need to use any third-party software. AWK scripts presented in Appendix E were used to find the values of desired performance metrics and the results were graphically displayed via R Programming, a language, and environment for statistical computing and graphics. Both EDBA and traditional dynamic bandwidth allocation algorithms were compared. Table VI further presents simulation parameter settings for the algorithm.

Radio propagation model –the two-ray-ground reflection model chosen because it is more suitable for this experiment as it makes use of both the direct path and the ground reflection path for communication, which for IHNs simulation is realistic than the free-space model without adding significant simulation time.

Link Layer – this study used Link Layer (LL) not oriented for satellite wireless and the queue for the simulation is Queue/DropTail/Priqueue to give priority to routing packages so that the simulation can have an overall improvement in the bandwidth allocation performance.

4.6. System Configuration and Design.

We combine largely two kinds of technology for design and realization of NS-2 based on IHN which is proposed in this study. We monitor the amount of network come in the house hold on NS-2 by communication with IHN to manage which device need more Bandwidth at a certain time.

4.7. Summary

In this chapter, this study provided a detailed discussion regarding the implementation and simulation of the algorithm (EDBA) proposed in this study. In this chapter, this study began by presenting and discussing the most popular used network simulation tools. Thereafter, this study presented a comparison between these most popular network simulation tools. Also, this study presented the motivation behind NS-2 being the chosen tool compared to other tools. Furthermore, this study provided a discussion on the NS-2 environment. This discussion included the basic architecture of NS-2, its directory structure, and class hierarchy wherein the relationships between its objects were presented. Further, this study presented the procedure followed in the configuration of NS-2. In addition, this study presented the simulation set-up of the EDBA algorithm proposed in this research work. Furthermore, this study discussed how the simulation tool recorded the results and further presented the plotting tool used to graphically display the results which were obtained in all the experimental evaluations. Lastly, this study discussed how the average throughput, delay, and energy consumption were calculated. The next chapter, therefore, presents and discusses the results obtained in all the experimental evaluations which were conducted to evaluate the performance of the EDBA algorithm.

CHAPTER 5 : RESULTS & DISCUSSIONS

5.1 Introduction

As briefly discussed in Section 4.2, simulation tools have helped several researchers to study and examine applications and protocols under varying network configurations. The reason behind simulations is to model the behavior of real-time communication networks. This is because it is time-consuming and somehow expensive to set up a real-time communication network. In this chapter, therefore, this study discusses the experimental evaluations of the proposed EDBA algorithm. The experimental evaluations rely on the results obtained and analyzed using NS-2 concerning three performance metrics considered in this study, namely; average throughput, average delay, and energy consumption.

The reason behind these metrics is that the research has concurred that to improve QoS in any network, these performance metrics are to be considered very important. During simulations of the proposed algorithm, trace files were automatically created and used to record all the results obtained. Furthermore, those results were used to calculate average throughput, average delay, and energy consumption. On the other hand, R Programming Language was used to display the analyzed results which were obtained through extensive simulations in a form of graphs. Dr. Robert I. Kabacoff designed and developed R Programming Language as a data plotting program in 2014. The motive behind this language is the creation of interactive graphs. This language is maintained by the R Foundation for Statistical Computing. A sample R Programming code is shown in Appendix D. Apart from that, AWK programs were also written to calculate the averages of the metrics defined in this study.

The remainder of this chapter is organized as follows: In Section 5.2, this study presents the experimental evaluations of the proposed algorithm. In Section 5.3, the summary of this chapter is presented.

5.2 Experimental Evaluations

As shown in Figure 4.10, this study used TCL scripts to create the proposed simulation topology in NS-2. The reason behind this was to evaluate the influence of integrating the PSO and Offline Cooperative Algorithm to come up with the proposed EDBA algorithm. During the simulations, the NAM interface was used to observe packet losses, node locations, and the transmission of packets. The sample NAM interface created and used for all the evaluations is shown in Figure 4.10. Looking at Figure 4.10, the topology is comprised of eight nodes, wherein each is statically configured with a unique IP address.

The proposed algorithm is implemented at the distribution (root node) layer of the network. The root node acts as the central coordinator of the network. On the other hand, the parent nodes offer different services to clients by defining the service to offer, unique identifiers of each service, and the bandwidth each service will consume. As presented in Table IV, all the simulations were performed using a propagation model of the Two-ray ground reflection model which is already defined in the NS-2 simulation environment.

The simulation tool was installed and run on Linux Ubuntu 16.04.5 LTS, which was also running on an Oracle VM VirtualBox Manager Version 4.3.20 developed by Oracle Corporation in 2014. The proposed EDBA algorithm and network topology configurations were implemented using C++ and OTcl code. This study had extensive simulations that were performed several times to obtain the most promising and reliable results. The benefit of NS-2 is that it automatically creates trace files to record results such as packet transmissions, and more. As discussed previously, this study applied R Programming to graphically display the results obtained and analyzed. On the other hand, this study had to write AWK scripts to calculate the averages of all the metrics.

5.3 Performance Metrics

The proposed model was examined against DQBA and DDA. The motivation behind choosing these two bandwidth allocation algorithms was that both were proposed to improve QoS in communication networking environments. Also, the focus of these algorithms was on network throughput, delay, as well as bandwidth usage which brought us to energy consumption. Furthermore, this study was also motivated by the following similarities and differences; DQBA integrated two bandwidth allocation algorithms, namely; static slot allocation and dynamic slot allocation.

The slot allocation algorithm deals with equally distributing the time slots of a parent node among its children nodes in a given cluster. While the dynamic slot allocation algorithm ensures that bandwidth demands of children nodes are placed to the parent node through sending traffic indication map (TIM). However, in the DQBA algorithm, the uses of other parameters like jitter and packet loss are left as a part of future work, and thus, the efficiency of this algorithm cannot be guaranteed since the evaluation was not thoroughly done.

On the other hand, the benefit of the DDA algorithm is that it can be applied in situations where multiple users request the same content, such as a popular TV show or shared material, through cellular networks. However, this study did not focus on delivery differentiation for different types of traffic (such as video and voice) in terms of data rate and delay, since different quality of service (QoS) requirements impact the formation of a coalition. Furthermore, looking at these two algorithms, the issue of defining different priority levels to all the nodes which is a major concern that leads to poor QoS was not taken into attention.

The research concurs that defining different priority levels to network nodes guarantees a certain level of performance during data transmissions. Apart from that, DDA is a data-driven algorithm, and therefore, relies on bandwidth and energy consumption to have sufficient data communication. Moreover, these algorithms did not consider permitting a maximum number of clients, in priority order, to access the resources and services offered by the network, which leads to network congestions.

The performance metrics defined below were considered to evaluate the performance of the proposed EDBA algorithm against DQBA and DDA.

a) **Average throughput** – the measurement of the amount of data that can be transmitted in a given time between two or more nodes within a network.
b) **Average delay** – the measurement of the time it takes data to travel from one node to another within a network.
c) **Energy consumption** – the measurement of the amount of energy consumed during data transmission within a network.

5.3.1 Average Throughput

To calculate the average network throughput, the AWK program was created and used. It is given in Appendix E. In the proposed EDBA algorithm, this study has observed a promising average network throughput compared to DQBA and DDA algorithms. This has been clearly shown in Figure 5.1. The reason behind this, the EDBA algorithm defines define priority limits for all relay nodes wherein each node is given a certain priority level depending on the workflow procedures. On the other hand, in the proposed algorithm, this study permits a maximum number of clients, in priority order, to access the resources and services offered by the network, which provides improvements on QoS.

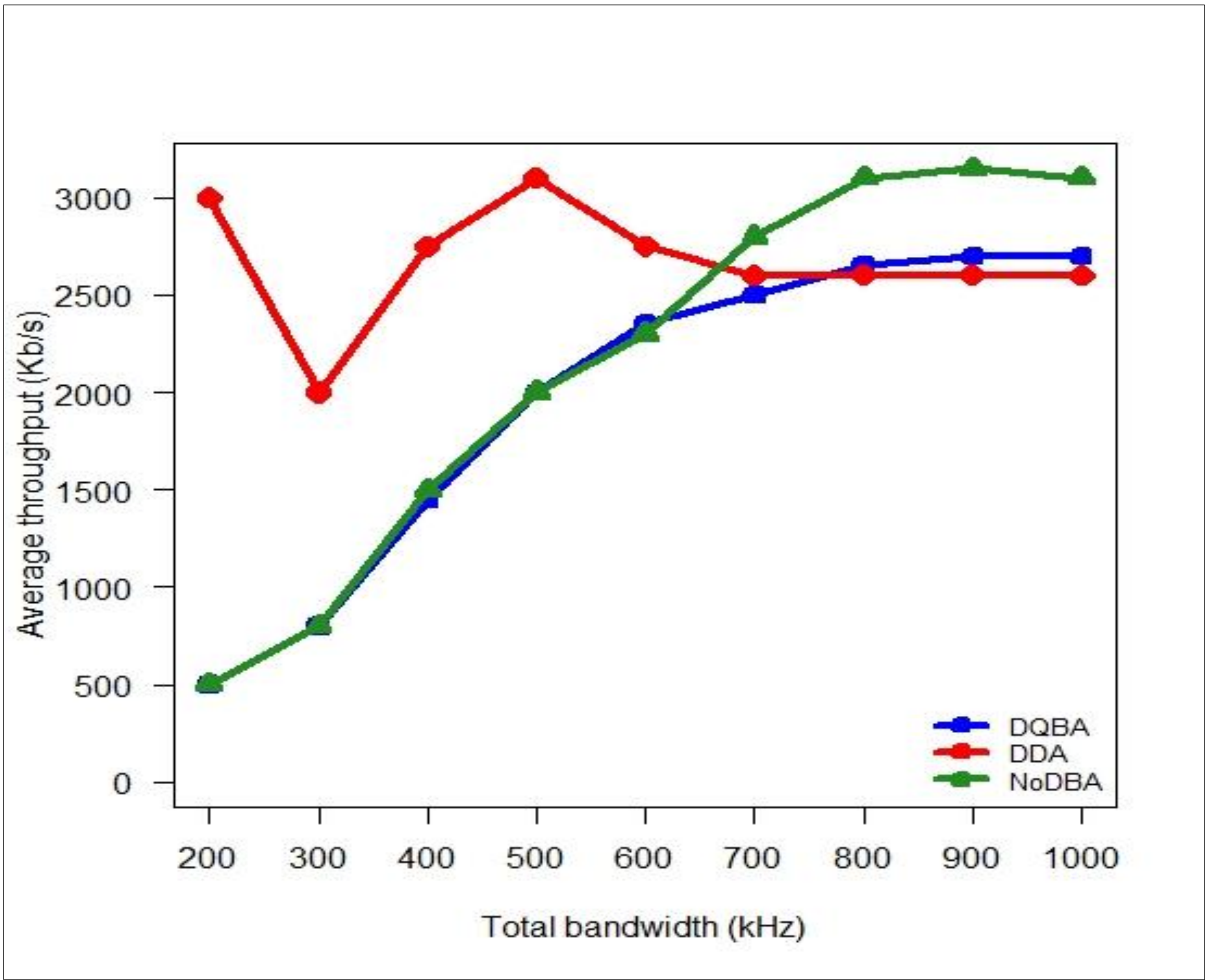

Figure 5:1: Average Throughput

5.3.2 Average Delay

Herein, this study presents and discusses the average network delay obtained by EDBA through multiple extensive simulations compared to DQBA and DDA algorithm. Looking at Figure 5.2, it is very clear that the EDBA algorithm reduces average network delay compared to other algorithms, which has been achieved by using the PSO algorithm to define priority limits to all the nodes in the network. The reduced delay was also achieved because of the minimized congestion, which is realized by permitting a maximum number of clients to access the network in priority order resulting in a minimized bandwidth usage.

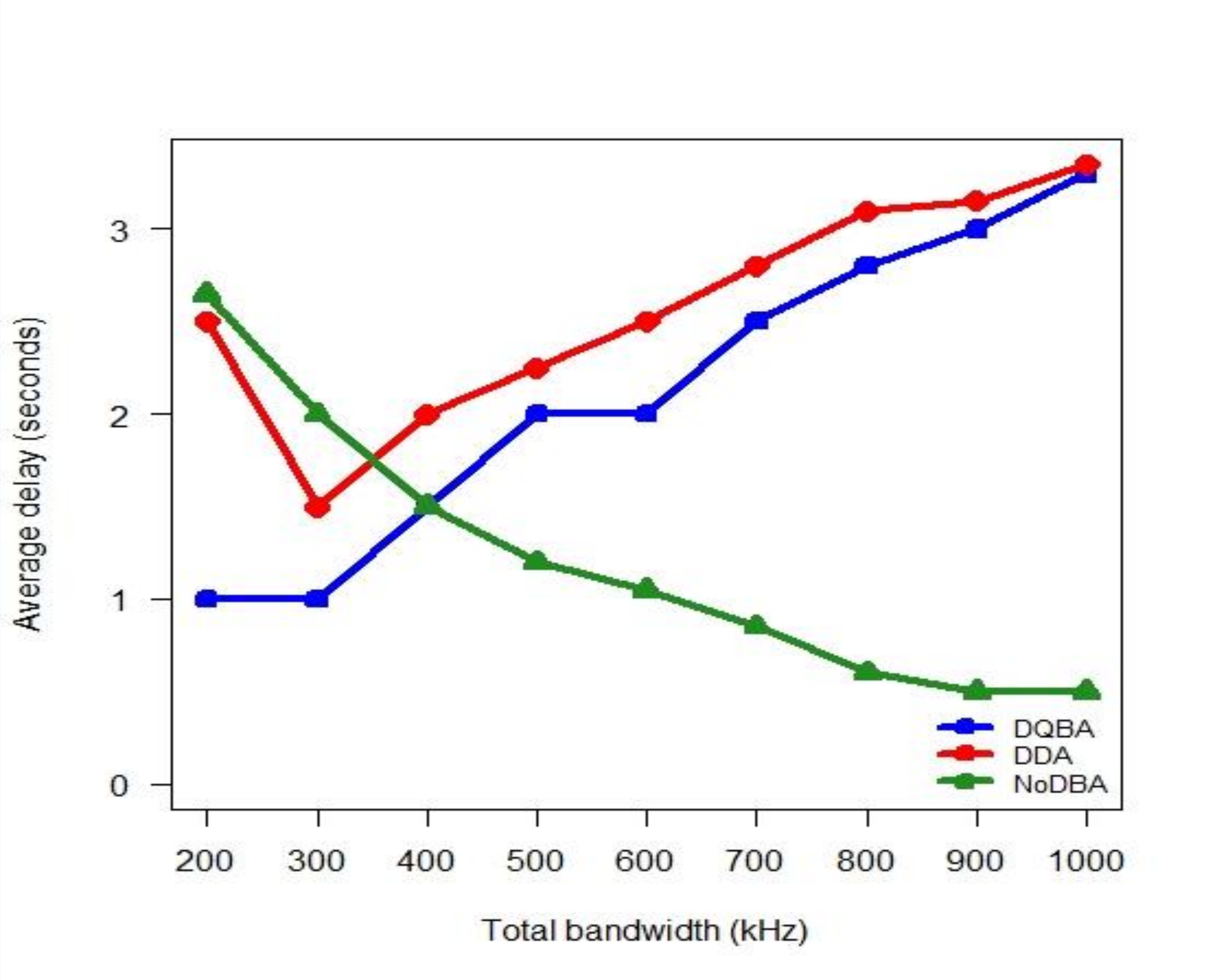

Figure 5:2: Average delay

5.3.3 Energy Consumption

Energy consumption, which is defined as the measurement of the amount of energy consumed during data transmission in a network was also improved. Looking at Figure 5.3, energy consumption has been well improved to the EDBA compared to DQBA and DDA algorithm. The reason behind this good improvement is that the EDBA algorithm integrated PSO and Offline Cooperative Algorithm, which provide a promising and minimized bandwidth usage as compared to other existing bandwidth allocation algorithms. Furthermore, by using the PSO algorithm, the proposed EDBA algorithm reduces congestions which on the other hand also saves on energy consumption.

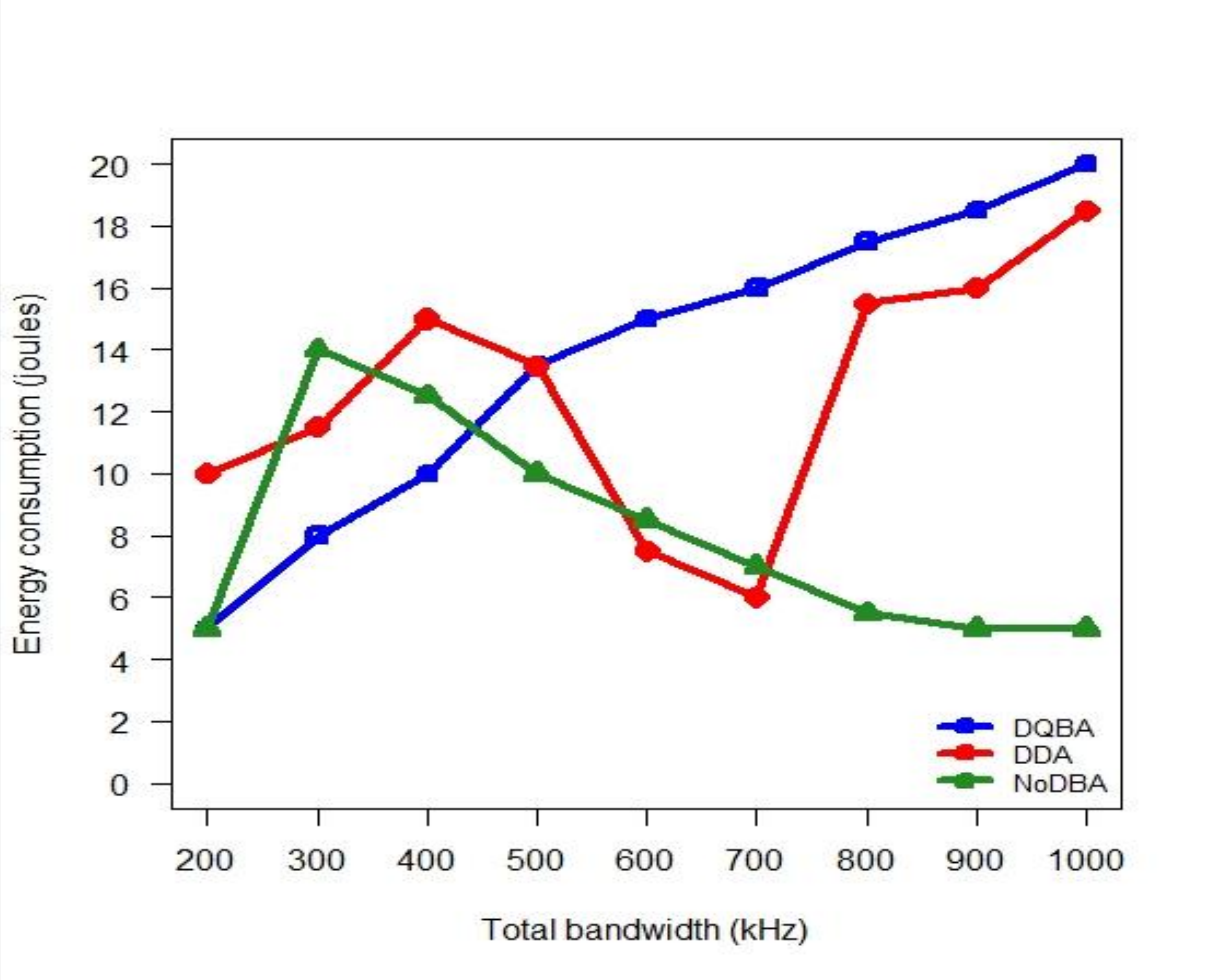

Figure 5:3: Energy consumption

Furthermore, the rate of lost packets is calculated as the number of lost packets over the total transmitted packets as given in equation (10). It is very important to realize from Figures 5.1 and 5.2 that the rate of lost packets decreases with the increase in total bandwidth. The proposed solution gave the lowest packet loss rate compared to DQBA and DDA algorithms because it resulted in improved average throughput and average delay, hence minimized the number of lost packets. For that reason, the services and resources that use the EDBA to communicate with each other and eventually with the internet are unlikely to be cut off which reduces packet loss rate as depicted in equation 10.

5.4 Summary

In this chapter, the main purpose was to present and discuss the experimental evaluations of the proposed EDBA algorithm using the simulation topology presented in Section 4.5. Furthermore, this study discussed the improvements offered by the EDBA algorithm in optimizing the performance and QoS in IHNs. The recorded results by NS-2 using trace files were analyzed and discussed in terms of three performance metrics mainly considered in this study. The metrics included average throughput, average delay, and energy consumption. Moreover, this study discussed the reason behind this choice, which is that QoS improvements rely on these performance metrics in any network. On the other hand, this study introduced the R Programming language and briefly mentioned that it is used to graphically display the results. In Section 5.2, this study presented a deep behavioral comparison concerning the proposed EDBA, DQBA, and DDA. In the next chapter, the objective achievements, conclusion, and future work of this study are presented wherein the discussion of the issues not addressed are briefly discussed.

CHAPTER 6 : SUMMARY, CONCLUSION & FUTURE WORKS

6.1 Summary

The purpose of this study was to design and implement an enhanced dynamic bandwidth allocation algorithm (EDBA) to optimize the performance of IHNs. The purpose of the proposed algorithm is to dynamically allocate bandwidth depending on the workflow procedures on each node. To achieve the objectives of this study, the algorithm integrated PSO and Offline Cooperative algorithms. Offline Cooperative Algorithm bases itself on the game-theoretical framework to any bandwidth allocation problem, and thus, it has been applied in the proposed algorithm to be responsible for the allocation of bandwidth in the network.

Furthermore, in this study, PSO is applied as an optimization algorithm. This algorithm is used to define priority limits for all the nodes in the network. The primary reason is to give different priorities to different relay nodes to assure a certain level of performance during client requests in the network. In the proposed algorithm, consider permitting a maximum number of clients, in priority order, to access the resources and services offered by the network, which in turn reduces network congestions. As a result, the proposed EDBA showed promising results by saving on bandwidth usage and improving QoS.

In this section, this study discusses how the set of technical objectives outlined in Section 1.6 of chapter 1 was achieved.

1. To identify the factors affecting QoS in IHNs.

In IHNs, users are found competing for the available services and resources, which leads to poor QoS. This, therefore, means users also compete for the available bandwidth. In Section 2.6, this study discussed the existing dynamic bandwidth allocation algorithms and the challenges associated with those algorithms. This study realized that most of these algorithms were developed for wireless networks, and thus, these have been ignored for IHNs. As a result, these models do not scale well in IHNs.

The reason behind all this is that in these algorithms, dynamic bandwidth allocation cannot be supported for wireless networks as well as IHNs simultaneously. Furthermore, this study identified that most of the recent bandwidth allocation algorithms have been developed only to focus on bandwidth allocation while not considering issues such as energy consumption and message dissemination. In this work, therefore, these algorithms are not scalable enough for IHNs.

In Section 2.8, this study presented and discussed related works wherein it was also discovered that none of the traditional algorithms were developed for IHNs but only for wireless networks. These algorithms were not concerned about energy consumption and message dissemination. Also, the traditional algorithms were not taking into account issues such as defining different priority levels to network nodes as well as permitting a maximum number of clients, in priority order, to access the resources and services offered by the network, which leads to network congestions.

2. To develop a taxonomy of existing bandwidth allocation algorithms for improving QoS in IHNs.

In Section 2.7 and 2.8, this study presented and discussed the existing dynamic bandwidth allocation algorithms for computer networks. Consequently, this study realized that none of the existing algorithms was developed for IHNs. On the other hand, these algorithms were only meant for the allocation of bandwidth, and thus, issues such as energy consumption and message dissemination were not taken into account. Also, this study identified that these algorithms did not consider assigning different priority levels to network nodes. This does not guarantee a certain level of QoS in any network. This study also permitted a maximum number of clients to have access to the resources and service of IHNs which was not taken into consideration by the traditional bandwidth allocation algorithms.

3. To develop a bandwidth allocation model for improving QoS in an IHN, using a hybrid of OCA and PSO methods.

This study developed the EDBA algorithm to address the performance of the bottleneck incurred because of bandwidth allocation in IHNs. The proposed EDBA algorithm optimizes the performance of IHNs by dynamically allocating bandwidth to all the nodes depending on the workflow procedures. In the algorithm, the nodes given more priority than others are assigned more bandwidth. In chapter 3, this study presented the design and model of the proposed IHNs architecture. The proposed architecture is a heterogeneous type of network, and thus, it is comprised of various types of network components. This study briefly explained the importance of each device, configuration, and connection it does within the network in optimizing and improving QoS in IHNs. In chapter 4, this study presented and discussed the implementation of the proposed EDBA algorithm. The implementation was carried out using the NS-2 simulation tool. This study had to modify some C++ files under the ns-2.35 directory. These modifications were specifically meant for message handling. On the other hand, this study made changes to some header, tcl files as well as the Makefile.in file.

4. To experimentally validate the model.

In chapter 5, this study shows how the proposed EDBA algorithm was evaluated against DQBA and DDA algorithms. As mention in Section 5.2, the evaluations were based on the three performance metrics considered in this study and those are average throughput, average delay, and energy consumption as the focus of this study is bandwidth allocation. This study used NS2 to simulate, test, and perform the evaluations of the EDBA, DQBA, and DDA algorithms. The sample simulation topology is illustrated in Figure 4.10. The topology was created using TCL scripts. This study had extensive simulations were multiple times to obtain the most convincing and promising results. Those results were automatically recorded in trace files. This study used those results to analyze average throughput, average delay, and energy consumption. As shown in Appendix C, this study has AWK programs to calculate the average of each of these performance metrics. On the other hand, R Programming was used to graphically display the analyzed results obtained during the simulations.

Moreover, IHNs have experienced an increase in the number of users, which has led to competition for available services and resources. These competitions have resulted in poor QoS because of issues such as a shortage of bandwidth, congestions, increased energy consumption, and more. In this research study, therefore, this study has proposed and developed the EDBA algorithm to solve the performance bottlenecks incurred in IHNs. The main reason behind this performance bottleneck has to do with bandwidth allocation. In the proposed EDBA algorithm, this study has integrated PSO and Offline Cooperative Algorithm.

PSO is used as an optimization algorithm used to define priority limits for all the nodes in the network. The PSO algorithm deals with assigning different priorities to different nodes to assure a certain level of performance during data communication in the network. On the other hand, Offline Cooperative Algorithm is responsible for the dynamic allocation of bandwidth in the network. By integrating the two algorithms, the proposed EDBA has obtained promising results by improving QoS.

Also, the EDBA algorithm optimizes the performance of IHNs by permitting a maximum number of clients, in priority order, to access all the network resources simultaneously. As mentioned in Section 3.2, the proposed network architecture was designed through pooling together various components such as Layer 3 switch, secured MG, server, Linksys wireless routers, and clients. This study implemented, tested, and evaluated the proposed EDBA algorithm using NS-2. NS-2 could be defined as an object-oriented discrete-event network simulator targeted at networking research.

Looking at Section 4.3, NS-2 is the most popular used simulation tool because it is less complex than other free simulation tools. In chapter 5, this study presented and discussed the results and analysis concerning three performance metrics considered in this work, namely; average throughput, average delay, and energy consumption. This study used the statistical results from the trace files to calculate average throughput, average delay, as well as energy consumption. On the other hand, this study wrote AWK scripts to calculate the average of each of the three metrics.

Also, this study wrote R Programming Language programs to graphically display the results after analysis. As a result, this study obtained an improved average throughput, average delay, and reduced energy consumption. These promising results yielded a better QoS compared to traditional algorithms. The contributions of this study, therefore, a dynamic bandwidth allocation algorithm, expected to optimize the performance of IHNs by improving QoS. This is expected to be achieved irrespective of the increase in numbers of home devices and services which results in congestion problems and competition of the limited available network resources. Moreover, this study plan to look into security issues for IHNs as a result of the Internet of IoT which was introduced recently.

6.2 Future Works

It was very difficult to create the proposed network topology in NS-2. The reason behind this is that this study did not know the OTcl language. Fortunately, that issue was solved by using the free source NS-2 topology generator tool. This tool is very great and allows one to drag and drop when defining the network nodes. Moreover, the files generated by the generator tool can be modified at any and have your own desired network topology. Apart from that, it was a very challenging experience to work on two programming languages (OTcl and C++) at the same. On the other hand, it was a bit challenging to come up with a mathematical model to design the proposed EDBA algorithm.

In this research study, this study did not take into account issues such as security and, thus, the network resources were not protected against unauthorized access, misuse, malfunction, modification, destruction, or improper disclosure by unknown users. Consequently, in the future, this study shall consider security as an important area of research for IHNs. Also, this study will look at more issues concerning bandwidth allocation. Furthermore, this study proposes extensive experiment evaluations of the proposed EDBA in much bigger and complex IHNs with more network resources. On the other hand, this study proposes research on the introduction of 5G, wherein more bandwidth will be required and a bandwidth allocation algorithm will be expected to have a quick response. Lastly, looking into the explosion of IoT, it could also be interesting to investigate and evaluate bandwidth allocation algorithms.

6.3 Conclusion

In this work, this study presented and discussed how the proposed model was developed and evaluated. The purpose was to present and discuss the experimental evaluations of the proposed EDBA algorithm using the simulation topology presented in Section 4.5. This study provided discussions on the improvements offered by the EDBA algorithm in optimizing the performance of IHNs in terms of and QoS. The results were recorded by NS-2 using trace files which were then analyzed and discussed in terms of three performance metrics mainly considered in this study. These metrics included average throughput, average delay, and energy consumption. The reason behind this choice of metrics is that QoS improvements rely on these performance metrics in any network. Looking at the result discussions and analysis, the proposed EDBA algorithm has yielded promising results because this study has realized an improved average throughput, a reduced average delay, and energy consumption. This, in turn, has resulted in an improved QoS in the proposed EDBA compared to DQBA and DDA algorithms. Consequently, the proposed EDBA algorithm has successfully attained the objectives of this study. This study has presented a deep behavioral comparison concerning the proposed EDBA against the performance of DQBA, and DDA in Section 5.2.

REFERENCES

AL-ALI, A., EL-HAG, A., DHAOUADI, R. & ZAINALDAIN, A. 2011. Smart Home Gateway for Smart Grid. International Conference on Innovations in Information Technology:90.

BAKARE, B. & ENOCH, J. 2019. A Review of Simulation Techniques for Some Wireless Communication System. International Journal of Electronics Communication and Computer Engineering, 10(2).

BALFAQIH, Z. 2020. Design and development of network simulator module for distributed mobility management protocol. Journal of Ambient Intelligence and Humanized Computing:1-15.

BERBEC, M. 2009. Intelligence Networks–The model based methodologies problem. Calitatea vieţii, 20(1-2):203-214.

CHEN, M., WAN, J., GONZALEZ, S., LIAO, X. & LEUNG, V. 2014. A survey of recent developments in home M2M networks. IEEE communications surveys & tutorials, 16(1):98 - 115.

CHEN, P.-Y. & REISSLEIN, M. 2018. FiWi network throughput-delay modeling with traffic intensity control and local bandwidth allocation. Optical Switching and Networking, 28:8-22.

CHORAFAS, D.N. 2019. INTELLIGENT NETWORKS: Telecommunications Solutions for the 1990s. CRC Press.

CHOUBEY, P.K., PATERIA, S., SAXENA, A., SB, V.P.C., JHA, K.K. & PM, S.B. 2015. Power efficient, bandwidth optimized and fault tolerant sensor management for IoT in Smart Home. 2015 IEEE International Advance Computing Conference (IACC). IEEE:366-370.

CROSSLER, R.E. & CHOO, K.-K.R. 2018. Intelligent Home Assistant Use in the Home Environment.

DAWADI, B.R., RAWAT, D.B. & JOSHI, S.R. 2019. Software-Defined IPv6 Network: A New Paradigm for Future Networking. Journal of the Institute of Engineering, 15(2):1-13.

ELSAYED, W., ELHOSENY, M., RIAD, A.M. & HASSANIEN, A.E. 2017. Autonomic self-healing approach to eliminate hardware faults in wireless sensor networks. International conference on advanced intelligent systems and informatics. Springer:151-160.

FAN, B., LENG, S. & YANG, K. 2016. A dynamic bandwidth allocation algorithm in mobile networks with big data of users and networks. IEEE Network, 30(1):6-10.

GAO, Z., ZHANG, H., DONG, S., SUN, S., WANG, X., YANG, G., WU, W., LI, S. & DE

ALBUQUERQUE, V.H.C. 2020. Salient object detection in the distributed cloud-edge intelligent network. IEEE Network, 34(2):216-224.

GONG, P., LI, M., KONG, J., LI, P. AND KIM, D.K. 2014. An interactive approach for QualNet-based network model evaluation and testing at real time. In 16th International Conference on Advanced Communication Technology. IEEE:978-982.

GUNGE, V.S. & YALAGI, P.S. 2016. Smart home automation: a literature review. International Journal of Computer Applications, 975:8887.

GUO, J., LIU, F., LUI, J.C. & JIN, H. 2015. Fair network bandwidth allocation in IaaS datacenters via a cooperative game approach. IEEE/ACM Transactions on Networking, 24(2):873-886.

HOU, J., WU, C., YUAN, Z., TAN, J., WANG, Q. & ZHOU, Y. 2008. Research of intelligent home security surveillance system based on ZigBee. 2008 International Symposium on Intelligent Information Technology Application Workshops. IEEE:554-557.

HUSSAIN, I., AHMED, Z.I., SAIKIA, D.K. & SARMA, N. 2015. A QoS-aware dynamic bandwidth allocation scheme for multi-hop WiFi-based long-distance networks. EURASIP Journal on Wireless Communications and Networking, 2015(1):160.

IMRAN, M. 2016. Intelligent home control and monitoring system via the internet. International Journal of Scientific Development and Research (IJSDR), 1(4):82-87.

Intelligent Home Network. 2021. [Online]. Available from: https://i.pinimg.com/originals/02/b0/e4/02b0e46310c9e0a7f49772bfb3e10e80.jpg [Accessed: 03 May 2021].

ISSARIYAKUL, T. & HOSSAIN, E. 2009. Introduction to network simulator 2 (NS2). In: Introduction to network simulator NS2. Springer:1-18.

K'OBWANGA, M.K., KOGEDA, O.P. & LALL, M. 2016. Performance Optimization of Intelligent Home Networks. In: Connectivity Frameworks for Smart Devices. Springer:209-234.

KAIWEN, C., KUMAR, A., XAVIER, N. & PANDA, S.K. 2016. An intelligent home appliance control based on WSN for smart buildings. 2016 IEEE International Conference on Sustainable Energy Technologies (ICSET). IEEE:282-287

Karim, S. and Ashraful, M.D., 2009. Simulation of New Security Elements in an Ad Hoc Network.

KARULF, E., 2008. BODY AREA NETWORKS (BAN). A survey paper written under guidance of Prof. RAJ JAIN.

KENNEDY, J. & EBERHART, R. 1995. Particle swarm optimization. Proceedings of

ICNN'95-international conference on neural networks, 4:1942-1948.
KNIAZIEVA, N., SHESTOPALOV, S., KUNUP, T. & KONDRATENKO, A. 2018. Research of intelligent network services traffic in NGN. 2018 14th International Conference on Advanced Trends in Radioelecrtronics, Telecommunications and Computer Engineering (TCSET). IEEE:995-1000.
K'OBWANGA, M.K., KOGEDA, O.P. & LALL, M. 2016. An Improved-Cross Layer Scheduling model for intelligent home networks. 2016 IST-Africa Week Conference. IEEE:1-9.
KUMAR, A., MEHTA, P. & KULKARNI, R.V. 2019. Particle Swarm Optimization for Solving Polynomial Equations. SASTech-Technical Journal of RUAS, 18(1):17-20.
KUMAR, S. & BANSAL, A. 2020. Performance Investigation of Topology-Based Routing Protocols in Flying Ad-Hoc Networks Using NS-2. In: IoT and Cloud Computing Advancements in Vehicular Ad-Hoc Networks. IGI Global:243-267.
LIU, L. & WANG, D. 2017. Design of intelligent home system based on internet of things. Proceedings of the International Conference on Graphics and Signal Processing.111-115.
LIU, R. & GE, Y. 2017. Smart home system design based on Internet of Things. 2017 12th International Conference on Computer Science and Education (ICCSE). IEEE:444-448.
LIU, X., WU, Q., SUN, J., XU, X. & WEN, Y. 2019. Research on Self-healing Technology for Faults of Intelligent Distribution Network Communication System. 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). IEEE:1404-1408.
LYNN, N. & SUGANTHAN, P.N. 2015. Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation. Swarm and Evolutionary Computation, 24:11-24.
MACONE, D., ODDI, G., PALO, A. & SURACI, V. 2013. A dynamic load balancing algorithm for Quality of Service and mobility management in next generation home networks. Telecommunication systems, 53(3):265-283.
MCEWEN, A. & CASSIMALLY, H. 2013. Designing the internet of things. John Wiley & Sons.
MENNICKEN, S. 2016. Designing Inhabitant-centered Experiences for Future Smart Homes. University of Zurich.
MOVASSAGHI, S., ABOLHASAN, M., LIPMAN, J., SMITH, D. & JAMALIPOUR, A. 2014. Wireless Body Area Networks. IEEE communications survey & tutorials:1658.
NAYYAR, A. 2018. Flying adhoc network (FANETs): simulation based performance

comparison of routing protocols: AODV, DSDV, DSR, OLSR, AOMDV and HWMP. 2018 International Conference on Advances in Big Data, Computing and Data Communication Systems (icABCD). IEEE:1-9.

NDLOVU, L., KOGEDA, O.P. & LALL, M. 2018. Enhanced service discovery model for wireless mesh networks. Journal of Advanced Computational Intelligence and Intelligent Informatics, 22(1):44-53.

NKOSI, T.L., MPHAHLELE, M., OJO, S.O. & MATHONSI, T.E. 2019. Dynamic Bandwidth Allocation Algorithm to improve Quality of Service in Intelligent Home Networks. 2019 Open Innovations (OI). IEEE:83-88.

QIAN, C., LI, Y., ZHANG, Q., CAO, B., LI, Z. & WANG, M. 2018. Staged priority-based dynamic bandwidth allocation in software-defined hybrid passive optical network. Optical Engineering, 57(12):126101.

RAFIQ, A., HAYAT, M.F. & YOUNUS, M.U. 2018. Efficient Bandwidth Management (EBM) Algorithm for NG-EPON. 2018 IEEE 4th International Conference on Computer and Communications (ICCC). IEEE:663-667.

SADKHAN, S.B., JAWAD, S.F. & ALSHALAH, H.A. 2017. Intelligent network used in distributed computer systems: An evaluation aspect. 2017 International Conference on Current Research in Computer Science and Information Technology (ICCIT). IEEE:169-173.

SIVARAMAN, V., GHARAKHEILI, H.H., VISHWANATH, A., BORELI, R. & MEHANI, O. 2015. Network-level security and privacy control for smart-home IoT devices. 2015 IEEE 11th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). IEEE:163-167.

WANG, D., TAN, D. & LIU, L. 2018. Particle swarm optimization algorithm: an overview. Soft Computing, 22(2):387-408.

YANG, B. 2017. Design and implementation of intelligent home wireless gateway based on STM32. 2017 4th International Conference on Information Science and Control Engineering (ICISCE). IEEE:258-260.

ZHANG, Y., WU, S., NIE, X. & ZHANG, C. 2017. The Design of Intelligent Home Furnishing System Based on ZigBee Technology. 2017 International Conference on Applied Mathematics, Modeling and Simulation (AMMS 2017). Atlantis Press.

ZOUAOUI, S., BOUSSAID, L. & MTIBAA, A. 2019. Priority based round robin (PBRR) CPU scheduling algorithm. International Journal of Electrical & Computer Engineering (2088-8708), 9(1).

APPENDIX A: TABLE OF NOTATIONS

Notations	Meaning
V	Set of nodes
E	Set of links
J	Set of gateways
S	Set of services
X	The priority level for the node
H	Highest prioritized node
L	Lowest prioritized node
∞	The least prioritized child node
P	The probability of choosing the next transmitter
C_m	Set of allocated egress
L_i	Lower bound

APPENDIX B: TCL SCRIPT

```
#=====================================================================
#       Copyright © 2016 – 2018 Tshwane University of Technology
#       Department of Information Technology
#       Author: Themba Lucky Nkosi
#       A TCL script proposed for experimental evaluations of the proposed EDBA
#=====================================================================
# Simulation parameters
# Define the multicast mechanism
set ns [new Simulator -multicast on]

# Predefine tracing
set nf [open t-down-th.nam w]
$ns namtrace-all $nf
set f [open t-down-th.tr w]
$ns trace-all $f

#set up color flow
$ns color 1 Blue
# Define the LTE topology
# UE(i) <--> eNB <--> aGW <--> server
# Other configuration parameters see ~ns/tcl/lib/ns-default.tcl
# step 1: define the nodes, the order is fixed!!
set eNB [$ns node];#node id is 0
set aGW [$ns node];#node id is 1
set server [$ns node];#node id is 2
set UE1 [$ns node]
set UE2 [$ns node]
set UE3 [$ns node]
set UE4 [$ns node]
set UE5 [$ns node]
# step 2: define the links to connect the nodes
```

```
$ns simplex-link $UE1 $eNB 75Mb 2ms LTEQueue/ULAirQueue
$ns simplex-link $eNB $UE1 150Mb 2ms LTEQueue/DLAirQueue
$ns simplex-link $UE2 $eNB 75Mb 2ms LTEQueue/ULAirQueue
$ns simplex-link $eNB $UE2 150Mb 2ms LTEQueue/DLAirQueue
$ns simplex-link $UE3 $eNB 75Mb 2ms LTEQueue/ULAirQueue
$ns simplex-link $eNB $UE3 150Mb 2ms LTEQueue/DLAirQueue
$ns simplex-link $UE4 $eNB 75Mb 2ms LTEQueue/ULAirQueue
$ns simplex-link $eNB $UE4 150Mb 2ms LTEQueue/DLAirQueue
$ns simplex-link $UE5 $eNB 75Mb 2ms LTEQueue/ULAirQueue
$ns simplex-link $eNB $UE5 150Mb 2ms LTEQueue/DLAirQueue
$ns simplex-link $eNB $aGW 250Mb 2ms LTEQueue/ULS1Queue
$ns simplex-link $aGW $eNB 300Mb 2ms LTEQueue/DLS1Queue
# The bandwidth between aGW and server is not the bottleneck.
$ns duplex-link $server $aGW 1Gb 2ms DropTail
# step 3.4 define the background traffic
# no parameters to be configured by FTP
# This study can configure TCP and TCPSink parameters here.
set tcp1 [new Agent/TCP/Linux]
$tcp1 set timestamps_ true
$tcp1 set fid_ 1
$ns attach-agent $server $tcp1
set sink1 [new Agent/TCPSink/Sack1]
$ns attach-agent $UE1 $sink1
$ns connect $sink1 $tcp1
set ftp1 [new Application/FTP]
$ftp1 attach-agent $tcp1
$ftp1 set type_ FTP
set tcp2 [new Agent/TCP/Linux]
$tcp2 set timestamps_ true
$tcp2 set fid_ 1
$ns attach-agent $server $tcp2
set sink2 [new Agent/TCPSink/Sack1]
$ns attach-agent $UE2 $sink2
```

```
$ns connect $sink2 $tcp2
set ftp2 [new Application/FTP]
$ftp2 attach-agent $tcp2
$ftp2 set type_ FTP

set tcp3 [new Agent/TCP/Linux]
$tcp3 set timestamps_ true
$tcp3 set fid_ 1
$ns attach-agent $server $tcp3
set sink3 [new Agent/TCPSink/Sack1]
$ns attach-agent $UE3 $sink3
$ns connect $sink3 $tcp3
set ftp3 [new Application/FTP]
$ftp3 attach-agent $tcp3
$ftp3 set type_ FTP

set tcp4 [new Agent/TCP/Linux]
$tcp4 set timestamps_ true
$tcp4 set fid_ 1
$ns attach-agent $server $tcp4
set sink4 [new Agent/TCPSink/Sack1]
$ns attach-agent $UE4 $sink4
$ns connect $sink4 $tcp4
set ftp4 [new Application/FTP]
$ftp4 attach-agent $tcp4
$ftp4 set type_ FTP
set tcp5 [new Agent/TCP/Linux]
$tcp5 set timestamps_ true
$tcp5 set fid_ 1
$ns attach-agent $server $tcp5
set sink5 [new Agent/TCPSink/Sack1]
$ns attach-agent $UE5 $sink5
$ns connect $sink5 $tcp5
```

```
set ftp5 [new Application/FTP]
$ftp5 attach-agent $tcp5
$ftp5 set type_ FTP
#loss module
set loss_module1 [new ErrorModel]
$loss_module1 set rate_ 0.02
$loss_module1 ranvar [new RandomVariable/Uniform]
$loss_module1 drop-target [new Agent/Null]
$ns lossmodel $loss_module1 $eNB $UE1
set loss_module2 [new ErrorModel]
$loss_module2 set rate_ 0.02
$loss_module2 ranvar [new RandomVariable/Uniform]
$loss_module2 drop-target [new Agent/Null]
$ns lossmodel $loss_module2 $eNB $UE2
set loss_module3 [new ErrorModel]
$loss_module3 set rate_ 0.02
$loss_module3 ranvar [new RandomVariable/Uniform]
$loss_module3 drop-target [new Agent/Null]
$ns lossmodel $loss_module3 $eNB $UE3
set loss_module4 [new ErrorModel]
$loss_module4 set rate_ 0.02
$loss_module4 ranvar [new RandomVariable/Uniform]
$loss_module4 drop-target [new Agent/Null]
$ns lossmodel $loss_module4 $eNB $UE4
set loss_module5 [new ErrorModel]
$loss_module5 set rate_ 0.02
$loss_module5 ranvar [new RandomVariable/Uniform]
$loss_module5 drop-target [new Agent/Null]
$ns lossmodel $loss_module5 $eNB $UE5
set loss_module6 [new ErrorModel]
$loss_module6 set rate_ 0.0002
$loss_module6 ranvar [new RandomVariable/Uniform]
$loss_module6 drop-target [new Agent/Null]
```

```
$ns lossmodel $loss_module6 $aGW $eNB
set loss_module7 [new ErrorModel]
$loss_module7 set rate_ 0.00002
$loss_module7 ranvar [new RandomVariable/Uniform]
$loss_module7 drop-target [new Agent/Null]
$ns lossmodel $loss_module7 $server $aGW
puts "Starting the simulation, please be patient....."
$ns at 1.0 "$ftp1 start"
$ns at 30 "$ftp1 stop"
$ns at 2.0 "$ftp2 start"
$ns at 30 "$ftp2 stop"
$ns at 3.0 "$ftp3 start"
$ns at 30 "$ftp3 stop"
$ns at 4.0 "$ftp4 start"
$ns at 30 "$ftp4 stop"
$ns at 5.0 "$ftp5 start"
$ns at 30 "$ftp5 stop"

# finish tracing
$ns at 30 "finish"
proc finish {} {
	#global ns f log
	global ns nf f
	$ns flush-trace
	close $f
	close $nf
	puts "running nam..."
	exec nam t-down-th.nam &
	exit 0
}

# Finally, start the simulation.
$ns run
```

APPENDIX C: SCRIPTING CODE

As discussed in chapter 5, to analyze or interpret the recorded trace files to extract useful information such as the number of packets received, dropped, or even throughput of the network, NS-2 permitted us to make use of AWK and GREP scripts. The AWK program is given by throughput.awk source code below.

```
#=============================================================
#	AWK script to calculate the average throughput
#=============================================================
BEGIN {
	recvdSize = 0
	startTime = 100
	stopTime = 0
}{
	event = $1
	time = $2
	node_id = $3
	pkt_size = $8
	level = $4
	#	recording the start time
	if (level == "AGT" && event == "s" && pkt_size >= 512) {
		if (time < startTime) {
			startTime = time
		}
	}

	#	updating the total received packet size and store packets arrival time
	if (level == "AGT" && event == "r" && pkt_size >= 512) {
		if (time > stopTime) {
			stopTime = time
		}
```

```
			hdr_size = pkt_size % 512
			pkt_size -= hdr_size

	#		recoding the received packet size
			recvdSize += pkt_size
	}
}
END {
	printf ("Average Throughput [kbps] = %.2f\t\t StartTime=%.2f\t
StopTime=%.2f\n", (recvdSize / (stopTime - startTime)) * (8 / 1000), startTime, stopTime)
}
```

APPENDIX D: R PROGRAMMING CODE

Herein this study provides the sample code used to plot the average throughput obtained by the proposed EDBA algorithm against DQBA and DDA algorithms as shown in Figure 5.1.

```
#=============================================================
#   R code to graphically display the average throughput
#=============================================================
#read DQBA, DDA    and EDBA values from tab-delimited SD_throughput.dat
autos_data <- read.table("C:\\R\\Throughput.dat", header=T, sep="\t")

max_y <- max(autos_data)
#define colours to be used for DQBA, DDA, and NTRP lines
plot_colors <- c("blue","red","forestgreen", "orange")

#graph autos using y axis that ranges from 0 to max_y.
#turn off axes and annotations (axis labels) so this study can specify them ourselves
plot(autos_data$DQBA, type="o", pch=22, lwd=4, col=plot_colors[1], ylim=c(0, max_y),
axes=FALSE, ann=FALSE)

#make x axis using 1-6 labels, which was the percentage of throughput
axis(1, at=1:9, lab=c(200, 300, 400, 500, 600, 700, 800, 900, 1000))
#making the y axis with horizontal labels that display ticks at every 10 seconds.
10*0:max_y is equivalent to c(0, 10, 20, 30, …).
axis(2, las=1, at=500*0:max_y)
#create box around plot
box()

#graph DDA with red dashed line and square points
lines(autos_data$DDA, type="o", pch=23, lwd=4, col=plot_colors[2])
#graph EDBA with green dotted line and diamond points
lines(autos_data$EDBA, type="o", pch=24, lwd=4, col=plot_colors[3])
#label the x and y axes with black text
title(xlab= "Total bandwidth (kHz)", col.lab=rgb(0,0.0,0))
title(ylab= "Average throughput (Kb/s)", col.lab=rgb(0,0.0,0))
```

```
#create a legend in the top-left corner that is slightly smaller and has no border
legend("bottomright", names(autos_data), cex=0.8, col=plot_colors, lwd=4, pch=22,
bty="n");
#turn off device driver (to flush output to PDF)
dev.on()
```

Appendix E: AWK Script

Herein this study provides the sample code used to calculate the average throughput obtained by the proposed EDBA algorithm against DQBA and DDA algorithms as shown in Figure 5.1.

```
#==============================================================
#   R code to graphically display the average throughput
#==============================================================
BEGIN {
        recvdSize = 0
        startTime = 400
        stopTime = 0
}
{
        event = $1
        time = $2
        node_id = $3
        pkt_size = $8
        level = $4

        # Store start time
        if (level == "MAC" && event == "s" && pkt_size >= 300) {
                if (time < startTime) {
                        startTime = time
                }
        }

        # Update total received packets' size and store packets arrival time
        if (level == "MAC" && event == "r" && pkt_size >= 300) {
                if (time > stopTime) {
                        stopTime = time
                }
                # Rip off the header

                hdr_size = pkt_size % 512
                pkt_size -= hdr_size
```

```
			# Store received packet's size
			recvdSize += pkt_size
		}
}
END {
		print "==============================================+\n";
		printf("\t\tThroughput[kbps] : %.2f\n",(recvdSize/(stopTime startTime))*(8/1000))
		print "\n+============================================+\n";
}
```