Portfolio Peak Algorithms Achieving Superior Performance for Maximizing Throughput in WiMAX Networks

Mohammed Sabri Al-Aboodi

Department of Computer and Information Sciences

University of Strathclyde

A Thesis Submitted in Fulfilment of the requirements for the degree of

Doctor of Philosophy

February 2021

Declaration of Originality

This thesis is the result of the author’s original research. It has been composed by the author and has not been previously submitted for examination which has led to the award of a degree. The copyright of this thesis belongs to the author under the terms of the United Kingdom Copyright Acts as qualified by University of Strathclyde Regulation 3.50. Due acknowledgement must always be made of the use of any material contained in, or derived from, this thesis.

Signed: Mohammed Al-Aboodi
Date: February 2021

Dedication

To

My beloved parents for their support and encouragement.

My beloved wife Nordaan for her patience and support.

My lovely kids Alhasan, and Hajar for their encouragement.

My brothers and sisters.

My father-in-law and his family.

My uncle Sattar Hussein for his support and sacrifices.

My cousin Mustafa Samir and his family.

All of my friends.

My sponsor "Al-Nahrain University - Baghdad".

&

My beloved country Iraq.

Acknowledgments

In the name of Allah the most gracious and the most merciful

First of all, praise and glory be to Allah the Almighty God who gave me strength and patience to complete this journey. My grateful appreciation goes to my parents, wife, children, and family.

My sincere thanks go to my supervisor Dr John Levine, for his continuous support and encouragement during my PhD, and for his insightful and constructive suggestions that helped me to improve this thesis. His support and guidance were essential for this work. Indeed, it was my privilege to do my PhD under his supervision.

I would like to express my thanks to Dr Mark Dunlop, and the staff of Department of Computer and Information Science at the University of Strathclyde, Glasgow, UK, for their unlimited support and valuable feedback during my PhD journey.

Furthermore, special thanks to my friends and colleagues for the unfailing support, considerable guidance and continuous encouragement that has supported me throughout the course of this work. Thanks also go to my new colleague Dorothea Hinrichs for her feedback and suggestions that helped me to improve my thesis.

I would also like to express my sincere gratitude and appreciation to my friend Allan Watson for his English proofreading and support throughout all this Journey.

Without you all, this accomplishment would not have been possible.

Abstract

The Mobile WiMAX IEEE 802.16 standards ensure provision of last mile wireless access, variable and high data rate, point to multi-point communication, large frequency range and QoS (Quality of Service) for various types of applications.

The WiMAX standards are published by the Institute of Electric and Electronic Engineers (IEEE) and specify the standards of services and transmissions. However, the way how to run these services and when the transmission should be started are not specified in the IEEE standards and it is up to computer scientists to design scheduling algorithms that can best meet the standards. Finding the best way to implement the WiMAX standards through designing efficient scheduler algorithms is a very important component in wireless systems and the scheduling period presents the most common challenging issue in terms of throughput and time delay. The aim of the research presented in this thesis was to design and develop an efficient scheduling algorithm to provide the QoS support for real-time and non-real-time services with the WiMAX Network. This was achieved by combining a portfolio of algorithms, which will control and update transmission with the required algorithm by the various portfolios for supporting QoS such as; the guarantee of a maximum throughput for real-time and non-real-time traffic. Two algorithms were designed in this process and will be discussed in this thesis: Fixed Portfolio Algorithms and Portfolio Peak Algorithm. In order to evaluate the proposed algorithms and test their efficiency for IEEE 802.16 networks, the authors simulated the algorithms in the NS2 simulator. Evaluation of the proposed Portfolio algorithms was carried out through comparing its performance with those of the conventional algorithms. On the other hand, the proposed Portfolio scheduling algorithm was evaluated by comparing its performance in terms of throughput, delay, and jitter. The simulation results suggest that the Fixed Portfolio Algorithms and the Portfolio Peak Algorithm achieve higher performance in terms of throughput than all other algorithms.

Keywords: WiMAX, IEEE802.16, QoS, Scheduling Algorithms, Fixed Portfolio Algorithms, and Portfolio Peak Algorithms.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

1G	First generation
2G	Second generation
3G	Third generation
4G	Fourth generation
AAS	Adaptive Antenna Systems
AMC	Adaptive modulation and coding
ATM	Asynchronous Transfer Mode
AWRR	Adaptive Weighted Round Robin
BE	Best effort
BER	Bit error ratio
BFWA	Broadband Fixed Wireless Access
BPSK	Binary Phase-Shift Keying
BS	Base Station
BWA	Broadband Wireless Access
CAC	Connection admission control
CDMA	Code Division Multiple Access
CID	Channel Identifier
CS	Convergence 19 Sub-layer
DES	Data Encryption Standard
DFPQ	Deficit Fair Priority Queuing
DL	Downlink
DL-MAP	Downlink map
DRR	Deficit Round Robin
DSL	Digital subscriber line
EDF	Earlier Deadline First
ertPS	Extended Real-time Polling Service
FDD	Frequency Division Duplex
F-FDD	Full-Frequency Division Duplex
FFT	Fast Fourier Transform
FIFO	First-in, first-out

GPC	Grant Per Connection
GPRS	General Packet Radio Service
GPS	Generalized Processor Sharing
GPSS	Grant Per Subscriber Station
GPTS	Grant per Type-of- Service
GSM	Global System for Mobile Communications
GSM	Global System for Mobile Communications
H-FDD	Half-duplex
HoL	Head-of-Line
http	Hypertext transfer protocol
IEEE	Institute of Electrical and Electronics Engineers
IfQ	Interface Queue
IMT	International Mobile Telecommunications
ITU-R	International Telecommunication Union- Radiocommunicatio
LBS	Location Based Service
LL	Link Layer
LL-WRR	Low Latency-Weighted Round Robin
LOS	Line-of-sight
LTE	Long Term Evolution
MAC	Media Access Control
MAC-CPS	MAC Common Part Sublayer
MAN	Metropolitan area network
MIB	Management Information Base
MIMO	Multiple Input Multiple Output
MR	Multi-hop relay
MRTR	Minimum reserved traffic rate
MS	Mobile Station
MSDUs	MAC Service Data Units
MSTR	Maximum Sustained Traffic Rate
MWRR	Modified Weighted Round Robin
NLOS	Non-Line-of-Sight

nrtPS	Non-real-time polling Service
NS2	Network Simulator
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal frequency-division multiple access
PHY	PHYsical layer
PMP	Point-to-Multipoint
PMP	Point-to-Multipoint
PPA	Portfolio Peak Algorithm
PQ	Priority Queue
PTP	Point-to-Point
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadra-Phase-Shift Keying
RR	Round Robin
RS	Relay Station
rtPS	Real-time polling service
SAPs	Service Access Points
SCFQ	Self-Clocked Fair Queuing
SNR	Signal to noise ratio
SOFDMA	Scalable Orthogonal Frequency Division Multiple Access
SON	Self-organizing network
SP	Strict-Priority
SS	Subscriber stations
TDD	Time Division Duplex
TDM	Time Division Multiplexing
TDMA	Time Division Multiple Access
TLS	Two-level scheduling
UGS	Unsolicited Grant Service
UL	Uplink
UL-MAP	uplink map
VoIP	Voice over Internet Protocol

WF2Q	Worst-case fair Weighted Fair Queuing
WFQ	Weighted Fair Queuing
Wifi	Wireless Fidelity
WiMAX	Worldwide Interoperability for Microwave Access
WMAN	Wireless Metropolitan Area Network
WMAN-SC	Wireless Metropolitan Area Network-Single Carrier
WMN	Wireless mesh network
WRR	Weighted Round Robin

CHAPTER 1

INTRODUCTION

This chapter introduces the general background to this research and identifies the specific research problems and the key motivating considerations. It also presents the principal objectives and describes the scope of this research. Furthermore, this chapter highlights the research contributions, justifies the intended benefits, and clarifies the implications of this research. Finally, this chapter summarises the organisation of the thesis.

1.1 Background

Over the past two decades, the Internet has played a very important role in the way we work, play, communicate and learn. The proliferation of mobile devices, laptops, tablets and smart phones significantly drives the demand for ubiquitous wireless Internet access for diverse sets of applications ranging from simple email and social networking to real time and bandwidth intensive applications such as; Voice over Internet Protocol (VoIP), streaming and interactive multimedia such as gaming. This remarkable growth in the demand for high data rates has driven the infrastructure of current wireless communication and will continue to drive it in the future through the continued expansion of the capacity for innovation and the development of robust techniques for the provision of high data rates and better services.

To respond to this demand, the Institute of Electrical and Electronics Engineers (IEEE) developed the IEEE 802.16 set of standards known as Worldwide Interoperability for Microwave Access (WiMAX) for Wireless Metropolitan Area Networks (WMANs). It developed through several versions and has defined 4G network specifications. Mobile WiMAX enables several impressive features including support of a high data rate, coverage of large areas, corporate-grade security, dynamic Quality of Service (QoS) and good spectral efficiency. It also provides better support for Non-Line-of-Sight (NLOS) technologies, multiple services with different QoS policies, fast and inexpensive deployment of “last mile access” to public networking and being a cost-effective alternative to WiFi and 3G/4G cellular networks (IEEEStandard, 2018).

A full mobility management feature has been introduced in IEEE 802.16e called Mobile WiMAX. Although the basic handover procedure was introduced for IEEE 802.16e to support full mobility, the emerging standards of IEEE 802.16j and IEEE 802.16m also exploit the same handover principles of IEEE 802.16e with other amendments on the requirements of these new standards (Teo et al., 2007, Bacioccola et al., 2010).

IEEE 802.16j introduces a relay station (RS) entity to provide larger coverage and better performance, especially around blind alleys (Genc et al., 2008). In contrast, the recent standard of IEEE 802.16m promises to meet the International Mobile Telecommunications-Advanced (IMT-Advanced) requirements to provide high data rates of at least 1 Gbps for fixed subscribers and 100 Mbps for mobile stations (MSs) at a vehicular speed of up to 350 km/h (Srinivasan, 2008). In addition, IEEE 802.16m supports the Media Access Control (MAC) and physical layer (PHY) features with the Location Based Service (LBS) solution, where the Base Station (BS) can track the Mobile Station (MS) movement.

The IEEE also developed one of these frameworks identified as 802.16 standards (IEEEStandard, 2018). This stands for Worldwide Interoperability for Microwave Access (WiMAX) offering by adopting 802.16. Mobile WiMAX offers peak data rates of 128 Mbit/s downlink and 56 Mbit/s uplink over 20 MHz wide channels.

Therefore, the question of how to fulfil the QoS requirement and service differentiation in IEEE 802.16 networks is one of the most important and open issues in the WiMAX algorithms.

In such a multi-service environment, the challenge is to ensure that Broadband Wireless Access (BWA) networks meet the diverse QoS requirements in terms of bounded delay, jitter, packet loss rate, guaranteed throughput, and spectral efficiency. In order to achieve QoS provisioning, several key modules including adaptive resource allocation, packet scheduling, and queue management, must be carefully designed. Consequently, new scheduling approaches are needed to control, organise, and coordinate such networks. Thus, within WiMAX Networks, there is a compelling need for scheduling algorithms to control the allocation of bandwidths.

Packet scheduling is a key mechanism in WiMAX that is used to support the required QoS. The mechanism must be effectively designed to maximise efficient utilisation of the spectrum and of systems resources. Scheduling is defined as the problem of determining which users will be given priority for the use of the bandwidth. This thesis focuses on packet scheduling algorithms for QoS provisioning in WiMAX networks, especially in the MAC and PHY layers.

In this thesis, two algorithms for the WiMAX Networks are proposed to guarantee the best QoS among the MSs, reduce delay and jitter and increase system throughput. These are:

(i) *Fixed Portfolio Algorithms* which is a fixed series of algorithms, and

(ii) *Portfolio Peak Algorithms (PPA)* which is described as an algorithm for data traffic control in high-speed WiMAX networks by switching algorithms based on peak performance of each existing algorithm.

The rest of the Chapter is organised as follows: First, the problem statement of this thesis is discussed. This is followed by a statement of the objectives, the scope of the thesis, and its contributions to research and further work in the field of study. Finally, the organisation of the thesis is presented.

1.2 Problem Statement

The IEEE 802.16 standard recognise the need to provide high transmission data and QoS for the diverse demand of users. However, the IEEE 802.16 standard does not specify which scheduling algorithm(s) should be used, which allows for vendors and researchers to innovate in this area. The algorithms in 802.16 are concerned with maintaining traffic scheduling continuity without much service disruption. However, for high priority flows, system throughput degradation may occur due to bandwidth disorder. Also, instability in real-time and non-real-time services leads to degradation in Jitter guarantee and deterioration of overall system utilisation. Prioritising non-real-time service could potentially impact real-time service negatively. Therefore, an efficient and fair scheduling algorithm should be developed towards improving the throughput and providing seamless QoS.

At the same time, algorithms must maximise spectral efficiency, reduce outage probability and maximise the utilisation of the system. Normally, algorithm selection for WiMAX is based on a single scheme. Due to the inefficient procedure of this scheme, the single algorithm is insufficient to meet the required QoS for different application services. Consequently, this issue will affect the real-time continuity of multimedia application sessions.

In the associated literature, the algorithms proposed for the Point-to-Point (PTP) and Point-to-Multipoint (PMP) models with specific performances are not suitable to meet the highest QoS requirements.

However, results in a number of published works (Hamza et al., 2010, Barooah et al., 2013) have shown that the throughput capacity of the WiMAX networks can be increased by developing an algorithm. The biggest challenge in the network is designing scheduling architecture that overcomes the problems of increased delay and signalling overhead in order to achieve guaranteed throughput and reduce delay under different circumstances to result in highly improved network performance.

1.3 Aim and Objectives

The principal aim of this thesis is to investigate and define the most efficient scheduling algorithm for achieving superior performance, maximizing throughput in WiMAX networks and providing the best QoS support for real-time and non-real-time services.

The following are the specific objectives to be obtained by using a portfolio of algorithms;

1. To optimise the selection of algorithms with the target of a portfolio of algorithms to meet the QoS requirements by means of reducing the delay and increasing the throughput.
2. To enhance the scheduling algorithms by combining a portfolio of algorithms, which will control and update transmission with the required algorithm by the various portfolios for supporting QoS such as; the guarantee of a maximum throughput for real-time traffic and reducing the delay.
3. To efficiently gather the results of the scheduling algorithm corresponding to different numbers of MS in all of circumstances, including real and non-real time applications.
4. To evaluate the behaviour of the proposed portfolio of algorithms by comparing and contrasting their performances under different scenarios using the Network Simulator (NS2).

1.4 Thesis Scope

This thesis covers scheduling algorithms at the MAC and PHY layers in the 802.16 standards for QoS provisioning, in both Uplink (UL) and Downlink (DL) directions of the Base Station (BS). Point-to-Multipoint (PMP) network topology is considered, where one BS serves multiple subscriber stations (SSs) and mobile stations (MSs) in a single cell. Throughout the simulation period, connections in the service class queues are assumed to be active in the network. Channel quality is assumed to be perfectly fed back by each SS to the BS. In this respect, resource allocation refers to the assignment of frequency-time slots in the Orthogonal frequency-division multiple access (OFDMA) frame. Even though the allocation of power is also possible, they are not covered in this thesis. It should be noted that there are other QoS related modules such as the admission control and buffer management, which remain outside the scope of this thesis.

1.5 Contributions

The focus of this thesis is on the development of fixed portfolio algorithms and portfolio peak algorithms for the WiMAX Networks. The proposed algorithms are diverse combinations of conventional algorithms each offering specific strengths and weaknesses that work combined to meet the demand for user to achieve high network performance.

Within this framework four principle contributions are proposed: Firstly, a new mode of scheduling algorithms has been developed which uses the requesting and granting of permission based on performance as a method of scheduling algorithms. Secondly, the proposed algorithms ensure that the scheduling transmissions of each user are maintained by assigning and switching to the highest scheduling algorithm of performance in each time frame which is proportional to the users' application demands. Thirdly, the transmission opportunity for each user will be assigned based on Quality of Service (QoS). Finally, the new proposed algorithms aim at reducing the delay resulting and maximise throughput in IEEE 802.16 standard.

In this thesis, 450 simulation experiments have been performed in order to study the performance of the proposed Portfolio algorithms and compare them with other conventional scheduling algorithms. The simulation experiments were performed under different network scenarios, as well as different settings and the various numbers of mobile nodes in Network Simulator (NS2) simulator. Thus, improving system performance through the exploitation of the available algorithms and optimising overall system throughput, without sacrificing QoS requirements.

1.6 Thesis Organization

This thesis is organized as follows: Chapter 1 provides a general introduction to the thesis with regards to the background of the subject and the problem statement, and it introduces the research objectives and highlights the scope and contributions of the thesis.
Chapter 2 provides a focused review of the literature on WiMAX Networks, and it covers the concept of scheduling algorithms. This Chapter also provides a general overview and classification of QoS classes. Further, the related work describes the ideas and concepts used in previous works briefly and highlights the strengths and limitations of these schemes.

Chapter 3 describes the methodology used in this thesis. The first section presents a brief description of our proposed algorithms. Then, the research framework of the proposed schemes is presented with an illustration. The stages of the research are depicted in the flowchart that shows the different patterns and their integration. The second part introduces the system models used in this study such as the channel capacity, queuing, traffic and network models. Finally, the Chapter gives an overview of the experimental parameters, environmental resources, and performance metric used.

Chapter 4 concludes with the results and observations of several experiments conducted to test and validate the proposed fixed portfolio algorithms in terms of efficiency, maximizing throughput and minimizing the delay.

Chapter 5 introduces a portfolio peak algorithms and provides its architectural design. The model presented in this chapter is an efficient algorithm for WiMAX real-time and non-real time transmission. The chapter concludes with the results and observations of several experiments performed with simulations.

In Chapter 6, the thesis is summarised, followed by a discussion of the key contributions of the work. Several directions for future investigation in future research are also suggested.

CHAPTER 2

BACKGROUND AND RELATED WORK

2.1 Introduction

This chapter presents the overview of the WiMAX Network, and background of the IEEE 802.16 standard. Scheduling Algorithms are presented in sections 2.3. Section 2.4 presents related works on scheduling algorithms. Finally, a summary of the chapter is presented in Section 2.5.

In the past two decades, wireless communications access technologies have seen an unprecedented demand for higher bandwidth by commercial centres and homes that required to build advanced generations of communication. The first generation (1G) was pitted by the need to establish a large geographical presence (Patel and Dennett, 2000). The second generation (2G) has moved from analog technology to digital communications and added features that allow modest data services along with voice communications for instance Global System for Mobile Communications (GSM) (Michel and Pautet, 1992). General Packet Radio Service (GPRS) is a representative 2.5G network that acts as transitional technology between 2G and 3G and supports low rate mobile data communications. The third generation (3G) mobile communication technology provides data services in the class of broadband communications (Patel and Dennett, 2000).

Eventually a new communication infrastructure that supports any form of media, be that data, voice, or video was developed: The fourth generation (4G) network. This mobile network can provide up to 100 Mbit/s, and is able to provide direct network connectivity to large areas including remote sites (Glisic and Makela, 2006). The spread spectrum radio technology used in 4G system is orthogonal frequency division multiple access (OFDMA), making it possible to transfer at very high bit rates. The peak bit rate is further improved by smart antenna arrays for Multiple-Input Multiple-Output (MIMO) communications.

In order to answer the need of QoS scheduling algorithms provisioning, this Chapter describes categorization of WiMAX QoS provisioning procedures that operate on two main layers: Physical (PHY) and Media Access Control (MAC). Then, this Chapter focuses on the progress of WiMAX QoS provisioning algorithms through intensive discussions of previous works on both MAC and PHY scheduling algorithms. Finally, it concludes with some comparison of QoS-based algorithms in both MAC and PHY layers used in WiMAX.

2.2 Overview of the WiMAX Network

WiMAX stands for worldwide interoperability for microwave access by the WiMAX Forum (IEEEStandard, 2018). It is based on IEEE 802.16 standard, officially known as Wireless Metropolitan Area Network (WMAN), it aims to provide wireless data coverage over a metropolitan area like a city. The forum describes WiMAX as “a standards-based technology enabling the delivery of last mile wireless broadband access as an alternative to cable and digital subscriber line (DSL)”, due to its high resource utilization, easy implementation and low cost. This technology becoming emerging for Broadband Wireless Access (BWA) due to its cost-effectiveness and compatibility with 4G-all-IP wireless networks.

Although the WiMAX standard supports both fixed and mobile broadband data services, the latter has a much larger market. The WiMAX system comprises of a radio tower, similar to a cellular base station (BS), and a WiMAX antenna and receiver at the customer end, which can be a modem, PC data card or even a mobile handset (IEEEStandard, 2018). Today WiMAX is one of the most outstanding standards capable of providing quadruple play technologies - data, voice, video and mobility - on a single network. The IEEE 802.16 family of standards developed to meet this demand. WiMAX not only attractive for areas without infrastructure but also offers higher data rate over a metropolitan area with a variety of Quality of Service (QoS) requirements (IEEEStandard, 2018).

2.2.1 Background of the IEEE 802.16 Standard

The Institute of Electrical and Electronics Engineers Working Group created the IEEE 802.16 in November 1998, to investigate and finalize the standard for Wireless Metropolitan Area Network (WMAN) (Alavi et al., 2005). The IEEE 802.16 standard established the base of the WiMAX technology, which was intended to be a Broadband Fixed Wireless Access (BFWA) system that will serve as a backhaul technology in the PMP network architecture. It was subjected to subsequent revisions by this standard to confirm that it will support the various mobility types.

The IEEE 802.16 standard defines MAC and PHY layers of the air interface for a Broadband Wireless Access BWA) in the frequency range of 10–66 GHz, using a Wireless Metropolitan Area Network-Single Carrier (WMAN-SC) with the target of delivering high-speed connections to businesses and homes that could not be reached with a wire line. The PHY layer radio interfaces have been defined as the early release of the IEEE 802.16 standard (Group, 2004).

The first version of the standard is called IEEE 802.16-2001 Standard (Marks, 2002), which was published on 8 April 2002 for fixed broadband wireless systems, also known as IEEE 802.16d standard, supports data rates up to 70 Mbps (2-10 Mbps per user) and covers up to 10 km^2, It is capable to operate under line-of-sight (LOS) with spectrum ranging from 2-11 GHz (Eklund et al., 2002, Kitti Wongthavarawat and Aura Ganz, 2003a).

The first amendment to the original standard was the IEEE 802.16c- 2002 standard (Marks, 2003) followed by the 802.16e which addeds support for mobility with 2-3

Mbps/user data rate for portable systems and 1-2 Mbps/user for mobile systems such that the coverage area extended to a radius of about 5 km (Fong et al., 2004, Hamiti, 2009). Figure 2.1 illustrates the layer coverage of the 802.16 standards (Eklund et al., 2002, Fong et al., 2004, Hamiti, 2009, IEEEStandard, 2018).

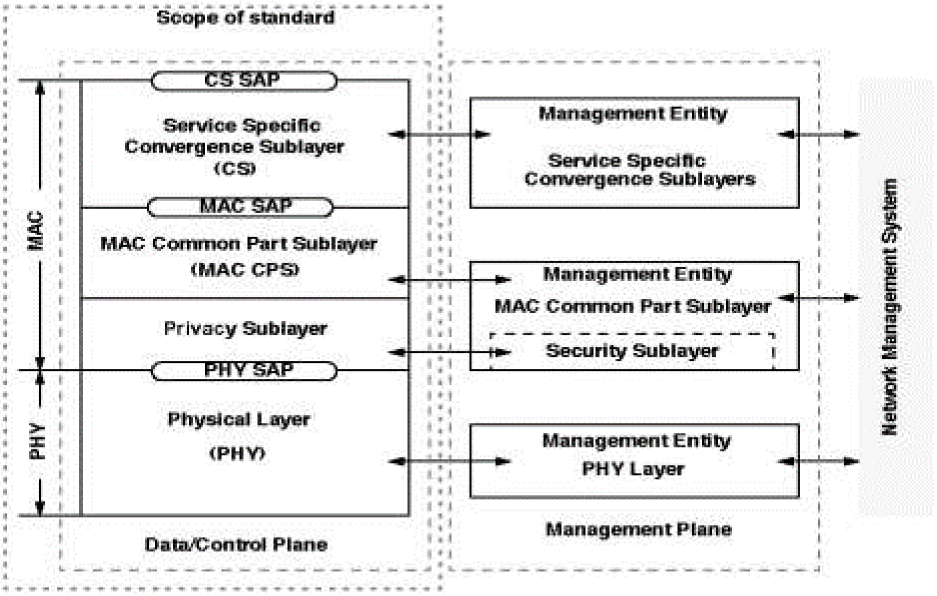


Figure 2.1: Protocol Layering of the IEEE 802.16 Standard

Source: (Hoymann, 2005, IEEEStandard, 2018)

Next, the IEEE 802.16a-2003 (Committee, 2004), targeted a lower air interface of 2-11 GHz with non-line-of-sight (NLOS) environment in order to support last-mile fixed broadband access. It improves the MAC layer and adds PHY layer specifications. The PHY layer specifications added are Orthogonal Frequency Division Multiplexing (OFDM), and Orthogonal Frequency Division Multiple Access (OFDMA). In the amendment, advanced power management techniques, multi-path propagation, adaptive antenna arrays, interference mitigation, and security were included.

The IEEE 802.16-2004 (Committee, 2004) integrated 802.16-2001, 802.16a and 802.16c and offered some clear improvements. It is called the Fixed WiMAX and is used for WiMAX certification. It supports both Time Division Duplex (TDD) and Frequency Division Duplex (FDD) operation mode. Its product profiles also utilize the OFDM 256- Fast Fourier Transform (FFT) system profile. The standard has shown significant enhancements, but it was only intended for fixed users without user mobility between the transmitting antennas. It also does not support the Management Information Base (MIB), a database used for managing all the entities in the network.

Due to the shortcomings of the Fixed WiMAX described above, the IEEE 802.16e-2005 (Group, 2004) emerged to support mobility. This is an extension to the Fixed WiMAX and is called Mobile WiMAX. It preserves all the features of the IEEE 802.16-2004 while adding mobility features to it. This new standard added features related to mobile operations, including the use of Scalable Orthogonal Frequency Division Multiple Access (SOFDMA), a modification to OFDMA, reserving a varying number of carriers such as 2000-FFT, 1000- FFT, 512-FFT and 128-FFT system profiles, Multiple Input Multiple Output (MIMO), and Adaptive Antenna Systems (AAS).

The IEEE 802.16f (Committee, 2004) standard is another variant of the fixed WiMAX to support MIB for the MAC and PHY layers. Which enabled mesh and multi-hop networks.

The IEEE standard has continued to evolve, including version 802.16g for management and procedures, 802.16h to improve co-existence with licensed exempt, and 802.16j-2009 to support relay station (RS) entity to extend the coverage and improve the performance and centralized/distributed control. The IEEE 802.16-2009 aimed to refine the MAC and PHY procedure for mobile operations (IEEEStandard, 2018).

Version 802.16m was introduced in March 2011 to amend the IEEE 802.16j-2009 and IEEE 802.16-2009 with Advanced Air Interface targeting data rates of 100 Mbit/s mobile and 1 Gbit/s fixed. It is also known as Mobile WiMAX Release 2 or Wireless MAN-Advanced which aims at fulfilling the International Telecommunication Union- Radiocommunication (ITU-R) -International Mobile Telecommunications-Advanced (IMT-Advanced) requirements for 4G systems.

The research presented in this research is based on the 802.16-2004 standard and 802.16e-2005 standard (IEEEStandard, 2018). These standards are geared to provide broadband Internet access to homes and commercial buildings. In the subsequent sections, some of the main features of the PHY and MAC layers as specified in the standards are presented.

The following table 2.1 shows the historical information regarding IEEE Std 802.16.

Table 2.1: Shows the Historical Information Regarding IEEE Std 802.16 [Reference] (IEEEStandard, 2018)

IEEE 802.16 Standards	Date approved by IEEE
IEEE Std 802.16-2001	6 December 2001
IEEE Std 802.16c™-2002 (amendment)	12 December 2002
IEEE Std 802.16a™-2003 (amendment)	29 January 2003
IEEE Std 802.16-2004	24 June 2004
IEEE Std 802.16f™-2005 (amendment)	22 September 2005
IEEE Std 802.16e™-2005 and IEEE Std 802.16-2004/ Cor1-2005 (amendment and corrigendum)	7 December 2005 (amendment) and 8 November 2005 (corrigendum)
IEEE Std 802.16g™-2007 (amendment)	December 2007
IEEE Std 802.16-2009	May 2009
IEEE Std 802.16j™-2009 (amendment)	May 2009
IEEE Std 802.16h™-2010 (amendment)	June 2010
IEEE Std 802.16m™-2011 (amendment)	March 2011
IEEE Std 802.16-2012	August 2012
IEEE Std 802.16p™-2012 (amendment)	October 2012
IEEE Std 802.16n™-2013 (amendment)	March 2013
IEEE Std 802.16q™-2015 (amendment)	February 2015
IEEE Std 802.16s™-2017 (amendment)	September 2017

2.2.2 QoS Scheduling Algorithms Categorizations in WiMAX

One of the challenges in WiMAX is efficient QoS scheduling design. Providing QoS scheduling for the WiMAX BS mainly involves the traffic scheduling at the MAC and subcarrier resource allocation at the PHY. Efficient algorithm scheduling is particularly important for their capability to maintain bandwidth and improve system usage throughput, besides QoS provisioning for diverse user requirements (IEEEStandard, 2018).

The IEEE.802.16 standard intentionally left the implementation of different protocol mechanisms open to allow vendors to provide an effective QoS for diverse user requirements. Such a mechanism shall comprise of traffic manipulation, traffic modelling, admission control and traffic scheduling in MAC layer as well as adaptive resource allocation in the PHY layer that deals with channel variations.

2.2.3 WiMAX Architecture

The WiMAX network consists of one Base Station (BS) and one or more Subscriber Stations (SSs). Usually, the BS is connected with the backbone network through a wired connection, however, in some cases it might be in the form of a wireless connection. Whereas, the SS receives services from the BS according to user requirements. WiMAX operates in two types of networking modes: Point to Multipoint (PMP) and Mesh Mode, as illustrated in Figure 2.2.

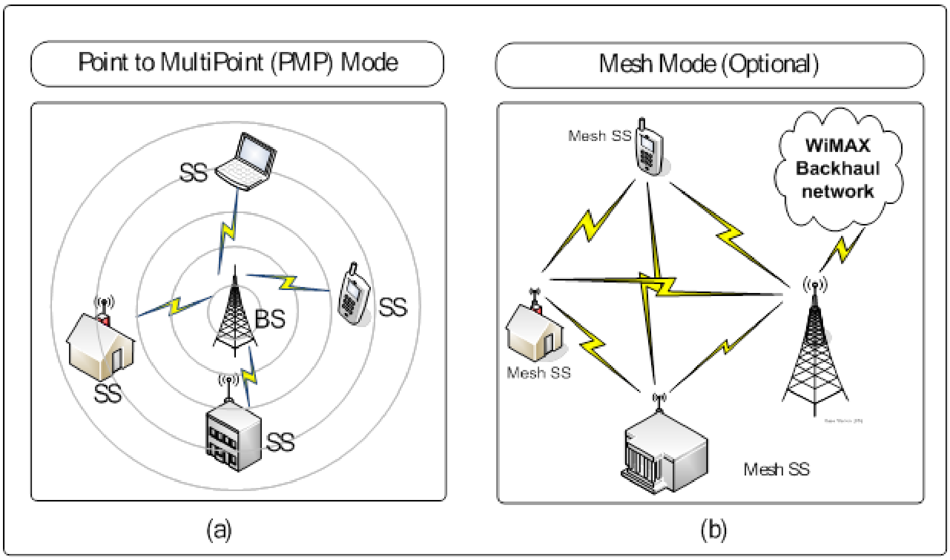


Figure 2.2: PMP and Mesh Operation Mode

In PMP mode, each SS in the network has a direct communication link with the BS and SSs are not allowed to communicate directly with each other. Whereas, in the case of Mesh mode, SSs might have direct communication with the BS as well as communicating with each other directly. The purpose behind this was to serve SSs that are in the depth coverage areas by using multi-hop links through intermediate SSs.

Meanwhile, transmissions take place through two independent channels: Downlink (DL) Channel which transmits from BS to SS and Uplink (UL) Channel which transmits from SS to BS. The UL Channel is shared among all SSs, while the DL Channel is used only by the BS.

The WiMAX standard specifies two scheduling modes. In Centralized Scheduling, BS acts as a central entity that performs all operations related to resource allocation and traffic scheduling. Whereas, in Distributed Scheduling, all SSs and BSs in the network participate in the process for scheduling and resource allocation, as all nodes compete for network resources by using an election algorithm. Usually, different traffic types and different nodes have different priorities within the network for proper scheduling of network resources (Msadaa et al., 2008).

2.2.4 WiMAX PHY Layer

The IEEE 802.16 physical layer (PHY) operates in the frequency range of 10-66 GHz and supports high capacity that requires in a LOS and operates in the 2-11 GHz frequency range, supporting NLOS, for both licensed and unlicensed bands. Based on (IEEEStandard, 2018), the Mobile WiMAX system profiles cover 5, 7, 8.75, and 10MHz channel bandwidths for licensed spectrum allocations in the 2.3, 2.5, 3.3, and 3.5 GHz frequency bands. This allows a certain flexibility depending on location; for instance, the frequency band (3.5 GHz) is already assigned to fixed services in many countries. The data transmissions are conveyed through OFDM symbols, which means involvement of up to 200 sub-carriers (IEEEStandard, 2018). The physical format of the OFDM system enables better performance of the NLOS environment than the single-carrier format systems, e.g., the Code Division Multiple Access (CDMA). The WiMAX Forum specified the 256 carriers of the OFDM format for the IEEE 802.16d standard. On the other side, the IEEE 802.16e standard adopted the OFDMA system and technique and the Scalable OFDMA (SOFDMA) is supported in the 802.16e by the WiMAX Forum as this provides easily scalable bandwidth and better immunity to interferences and channel fading. However, (IEEEStandard, 2018) specified the numbers of data carriers as 512, 1024, and 2048 owing to that these carriers provide better NLOS and handling of multi-path and channel fading.

The WiMAX PHY layer performs several tasks, such as the transfer of data, physical connections between end devices, the transmission power used, the type of signal used, and the type of modulation and coding scheme such as Binary Phase-Shift Keying (BPSK), Quadra-Phase-Shift Keying (QPSK), 16- Quadrature Amplitude Modulation (QAM) and 64-QAM applied. WiMAX is transmitted at high speed on the air interface through radio electromagnetic waves using a certain range of frequency bands. The 802.16 PHY layer covers a frequency range of 2–66 GHz.

2.2.5 WiMAX MAC Layer

This section elaborates on the main features and functioning of the MAC layer in the BS and the SSs with special emphasis on the BS.

The IEEE 802.16 MAC layer provides an interface between the PHY and the transport layers (upper layers). From the transmitter side, the MAC layer modifies and passes the MAC Service Data Units (MSDUs) from the transparent layer to the PHY layer, and the reverse mechanism takes place at the receiver's side. A Convergence 19 Sub-layer (CS) is included in both fixed and mobile WiMAX MAC layers (Figure 2.1). This CS layer is capable of linking upper layer protocols such as the Asynchronous Transfer Mode (ATM), Time Division Multiplexing (TDM), voice, and other advanced protocols with the MAC layer (Stallings, 2009, Andrews et al., 2007, Nuaymi, 2007).

2.2.5.1 WiMAX MAC Sublayers

The IEEE 802.16-2004 and IEEE 802.16e-2005 MAC design include three sublayers namely, the service CS layer, MAC Common Part Sublayer (MAC-CPS) and the Security Sublayer.

The following sections will go into more detail for MAC QoS scheduling algorithms as previously categorized in Figure 2.3.

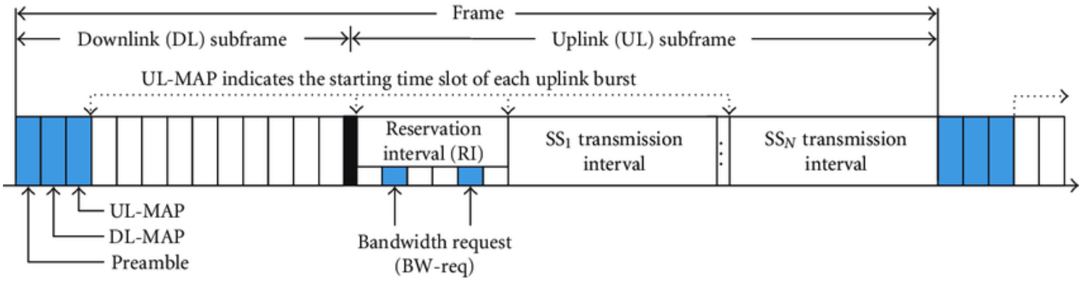


Figure 2.3: IEEE 802.16 MAC frame structure in the TDD/TDMA mode
Source: (Saffer et al., 2011)

2.2.5.1.1 The service Convergence Sublayer (CS)

The service-specific convergence sub-layer is a service-dependent layer that assures data transmission. It takes data from the upper layer entities (e.g., routing) and enables QoS and bandwidth allocation. It additionally affects payload header suppression and increases the link’s efficiency. Furthermore, the IEEE 802.16 specifies two types of SSCS for the mapping function:

(i) an ATM convergence sub-layer, which is a logical interface responsible for ATM services. In operation, it accepts ATM cells from the ATM layer, classifies them and then sends them as CS PDUs to MAC SAP. It differentiates virtual path, switched ATM connection and assigns channel Identifier (CID); and

(ii) the packet convergence sub-layer which is a packet-based protocol that performs packet mapping such as IP, IP v4, IP v6, IEEE 802.3 Ethernet LAN, VLAN and Point-to-Point Protocol (PPPP).

2.2.5.1.2 MAC Common Part Sublayer (MAC-CPS)

The next layer, which is the common part sublayer (CPS), defines the multiple access mechanism. It is responsible for the major MAC functionalities like system access, the establishment of the connection, maintenance, and bandwidth management. Since the WiMAX MAC is connection-oriented, this sub-layer provides service flows after registration of each subscriber’s station. Moreover, it provides QoS for service flows and manages connections by adding, deleting, or modifying the connection statically or dynamically. On a downlink channel, the only station that transmits is the BS and it does not require any coordination function. The SS receives only those messages directed to them on an uplink channel (Andrews et al., 2007).

2.2.5.1.3 The security sublayer

The security sublayer transmits secure procedures within a shared wireless network. This involves a supervision of the authentication, provision of secure key exchange and protecting the data with encryption policies between the BS and the MS. These sublayers are associated with each other via the Service Access Points (SAPs).

2.2.5.2 TDD - FDD and Time Division Multiple Access (TDMA)

The WiMAX networks support full and half duplex (IEEEStandard, 2018), as two types of transmission modes, namely, the Time-Division Duplex (TDD) and Frequency Division Duplex (FDD) modes, which are supported by the IEEE 802.16 standards. In the FDD mode, the UL and DL are located on separate frequencies with fixed-duration frame utilization (IEEEStandard, 2018). This allows for use of different modulation types (Andrews et al., 2007). Thus, both full-duplex (F-FDD) and half-duplex (H-FDD) subscriber stations may be used for transmission and reception simultaneously (Nuaymi, 2007, IEEEStandard, 2018). Additionally, the TDD frame structure has a fixed time duration and is divided into two parts; DL and UL sub-frames, located at different times in the same frequency (Figure 2.4).

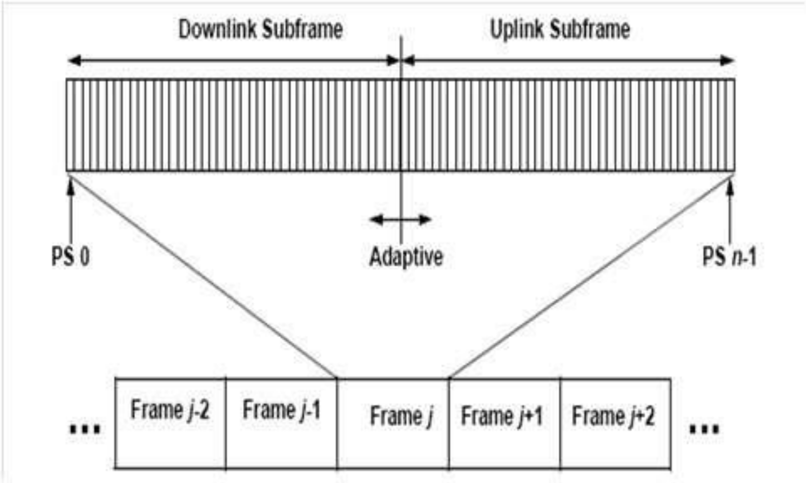


Figure 2.4: The TDD Frame Structure

Source: (IEEEStandard, 2018)

The TDD frame is actually partitioned into an integer number of physical slots (PSs) where the PS is the smallest unit of the partitioning bandwidth (Andrews et al., 2007, IEEEStandard, 2018).

The transmitting and receiving processes could be performed in both FDD or time division duplex TDD methods which are used separately by the DL and UL subframes. Both subframes that use FDD take place on separate frequency bands and may be synchronized in time. SSs can either transmit and receive simultaneously as full-duplex or transmit and receive at non-overlapping time intervals as half-duplex.

TDMA allows the SSs to share a wireless channel in the UL subframe. Nevertheless, a downlink map (DL-MAP) and uplink map (UL-MAP) information are broadcasting by the DL subframe, and the data are transmitted sent in bursts using TDMA as illustrated in Figure 2.5.

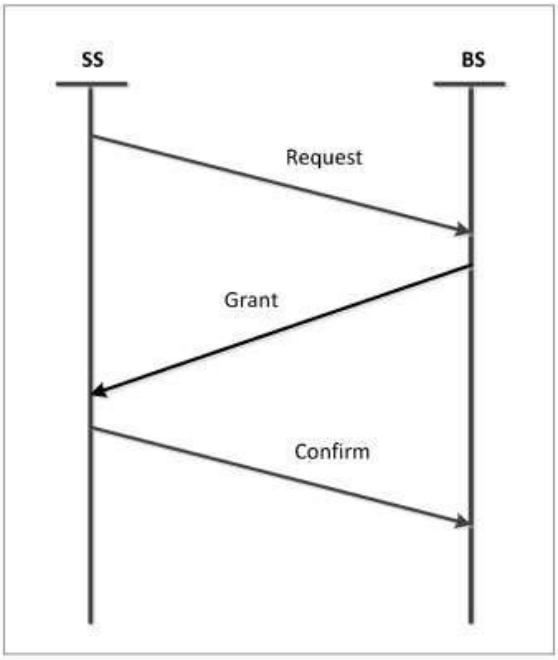


Figure 2.5: A Representation of the Three-Way Handshake in the WiMAX Networks

Source: (IEEEStandard, 2018)

In summary, the TDD mode is preferred over the FDD mode by the majority of implementations and in practice, most network companies base all initial WiMAX profiles on the TDD mode, except for two Fixed WiMAX profiles operating at 3.5 GHz, which are based on the FDD mode. The heavy reliance of network companies on the TDD mode can be attributed to the variety of advantages this mode offers:

1. Flexibility in choosing the UL to DL data rate ratios,
2. Ability to exploit channel reciprocity,
3. Ability to operate in the non-paired spectrum, and
4. Low complexity transceiver design (Andrews et al., 2007).

2.2.6 WiMAX OFDMA

The OFDMA was specially designed for the 4G wireless networks and is a combination of FDMA, TDMA, and CDMA (Andrews et al., 2007, Stallings, 2009, Nuaymi, 2007, Xiao, 2006). The OFDMA is recognised as a powerful access technology that increases system capacity. This is done by using a modulation technique of the OFDM by employing a combination of frequency diversity as well as multi-user diversity by allocating various divisions of OFDM subcarriers to many users giving adequate attention to interferences (IEEEStandard, 2018).

Review of the relevant literature demonstrates that there is agreement in the scientific community that OFDMA is currently the best access method for the multi-user environment (Bai et al., 2010, Sundaresan et al., 2008, Zhang and Lau, 2007).

More specifically, in both the OFDM and OFDMA PHY layers, the physical slots (PS) is the duration of four modulation symbols. The frame is not necessarily divided into two equal parts. In the TDD mode, the split between the UL and DL is a system parameter that is controlled at the higher layers (Kitti Wongthavarawat and Aura Ganz, 2003a, IEEEStandard, 2018).

As Figure 2.6 shows, the general format of a TDD frame specified for an OFDM PHY. The DL and UL sub-frames are transmitted in the same frequency but have different transmission times, on two separate zones (DL-zone and UL-zone). It has been reported that the contents are the same for the FDD and TDD modes (IEEEStandard, 2018).

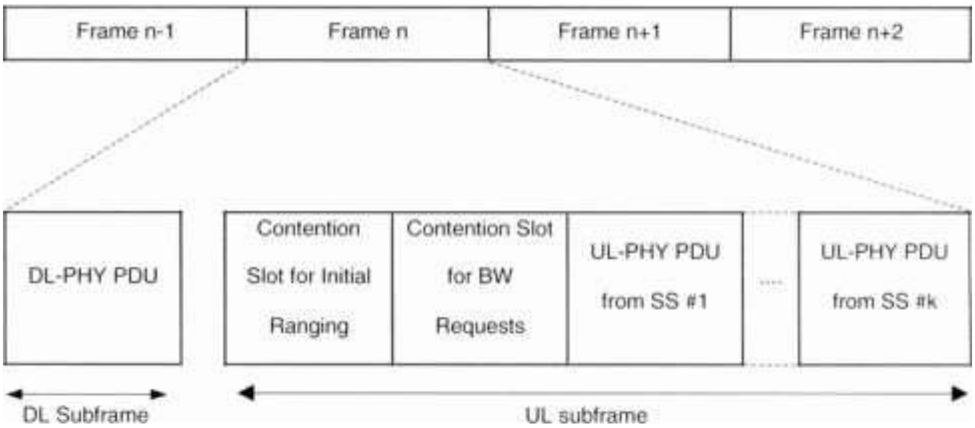


Figure 2.6. The TDD Frame Format

Source: (IEEEStandard, 2018)

2.2.7 Scheduling Service and Connections

Five different scheduling services are supported in the IEEE 802.16 standard (IEEEStandard, 2018, Andrews et al., 2007). These services determine how the network should allocate the opportunities for both UL and DL transmissions and how a mobile station (MS) can request bandwidth and performing handovers during movement (Mukherjee and Viswanathan, 2005). Ensuring QoS is one of the fundamental components of the WiMAX MAC-layer. The WiMAX network can guarantee and meet the QoS requirements during and after the handover to support different user applications. WiMAX handles the QoS control by using a connection-oriented MAC architecture. Table 2.2 lists the five scheduling services defined in the standard (IEEEStandard, 2018).

Table 2.2: Service Type Classes Characteristics

Scheduling Service	Applications	QoS Specification	Access to resources
Unsolicited Grant Service (UGS)	Constant bit rate real time service flows, and low jitter and delay tolerance such as T1/E1 transport.	Maximum sustained rate, Maximum latency tolerance, Jitter tolerance	Real-time data streams consisting of fixed-size data packets issued at periodic intervals, e.g., uncompressed voice and TDM circuits. The BS provides periodic grants for this type of service.
Extended Real-time Polling Service (ertPS)	Varies bit rate of real-time services. Applied on VoIP with silence suppression.	Maximum sustained rate, Maximum latency tolerance, Jitter tolerance Minimum Reserved Rate, Traffic Priority	Real-time service flows that generate variable data size packets on a periodic basis. The MS is allowed to change its BW requirements over time. Periodic grants are assigned like in the UGS and these grants can be used to transmit data as well as to request additional BW (unlike the UGS).
Real-time polling service (rtPS)	Real-time data streams comprising variable size data packets that are issued at periodic intervals such as streaming video (MPEG).	Maximum Sustained rate, Minimum Reserved Rate, Maximum Latency Tolerance, Jitter Tolerance, Traffic Priority	Real-time traffic. Periodic request opportunities are assigned, which allow an MS to specify the amount of BW required each time. The MS cannot use contention request opportunities; it only uses the assigned unicast poll.
Non-real-time polling Service (nrtPS)	Can be used for TCP-based applications with relaxed delay sensitivity such as FTP with guaranteed minimum throughput.	Minimum Reserved Rate, Maximum Sustained rate, Traffic Priority	Non real time. Delay-tolerant data streams comprising variable-size data packets for which a minimum data rate is required. It is similar to rtPS but the polling period is longer and contention request opportunities are

			allowed even though they can be restricted via the transmission/request policy.
Best effort (BE)	Lowest QoS level applications such as HTTP or e-mail.	Maximum Sustained rate, Traffic Priority	Non real time. Data streams for which no minimum service level is required and therefore may be handled on a space-available basis. All forms of polling are allowed.
Source: (Mehta and Gupta, 2012, ElgeredA et al., 2010 &, Yadav and Chauhan, 2016)			

The QoS parameters control the quality of service requirements, such as priority, reliability, speed, and amount of certain applications. Depending on the QoS parameters, such applications should be mapped to a scheduling service that supports their proper operation (Zeng and Zhu, 2009). Table 2.3 lists the QoS parameters supported for each scheduling service. For example, VoIP has a stable delay and jitter, so it should be mapped to the UGS or ertPS. In the ertPS service, the bandwidth requirements can change over time. Therefore, during silence periods the bandwidth requirement can be set to zero. To illustrate another example, an HTTP web application which does not have specific bandwidth or delay requirement can be sent over a BE scheduling service (Kim and Yeom, 2007).

Table 2.3: The QoS Parameters Supported For Each Scheduling Service

QoS parameter	**UGS**	**ertPS**	**rtPS**	**nrtPS**	**BE**
Maximum Sustained Traffic Rate (MSTR)	Y	Y	Y	Y	Y
Minimum reserved traffic rate (MRTR)	N	Y	Y	N	Y
Maximum latency	Y	Y	N	N	Y
Tolerated jitter	Y	N	N	N	Y
Traffic priority	N	Y	Y	Y	Y
Y: Supported. N: Non-supported.					

- MSTR: Defines the services’ peak information rate police and shape the traffic flow.
- MRTR: Specifies the minimum rate reserved for the flow calculated excluding MAC overhead.
- Maximum latency: Specifies the maximum interval between the reception of the packet at the transmit end and the arrival of the packet at the receiving end.
- Tolerated jitter: Specifies the maximum delay variation for the user’s connections.
- Traffic priority: Specifies the priority of the associated service flow over the other.

When a subscriber ranges into the WiMAX network, three bidirectional MAC CIDs are used for management of the subscriber station traffic and the scheduler’s station entity (BS, RS or MS) (IEEEStandard, 2018):

- Primary CID is used for delay-sensitive management messages, e.g., ranging or bandwidth allocation.
- Secondary CID is used for secondary management. It should be assigned locally by the access RS.
- Basic CID is used for less-sensitive MAC messages, i.e., the IP address request.

In addition, unidirectional data CIDs will be used to identify traffic flows between the subscriber station and the scheduler station. These data CIDs are mapped into different scheduling services depending on the users’ classification rules (Huang et al., 2009, Xiao, 2006, Tao et al., 2007).

In the DL transmission, the scheduler station decides which packets to schedule next, queues for different scheduling services to start filling-up, and then communicates the scheduling decision in the DL-MAP together with the corresponding burst within the same frame. In the UL transmission, the procedure is more complex, depending on the scheduling service. However, in the WiMAX, the subscriber stations’ connections may have either periodic transmission grants (i.e., UGS), periodic opportunities to request bandwidth (i.e., ertPS, rtPS, nrtPS), or contention opportunities to request bandwidth, i.e., ertPS, nrtPS, and BE (Zeng and Zhu, 2009, Kim and Yeom, 2007, Wang and Jia, 2010, Salah et al., 2010). The WiMAX or IEEE 802.16 standards allow the scheduling BS to allocate bandwidth in a request/grant scheme where request is always per connection. The grant can either be per connection as Grant Per Connection (GPC), or per subscriber, so Grant Per Subscriber Station (GPSS). The IEEE 802.16 standard recommends employment of the GPSS and leaves the allocation for each connection open to the MS scheduler (Andrews et al., 2007, Athanasopoulos et al., 2010).

2.2.8 Connection Admission Control Algorithms (CAC)

The connection admission control (CAC) is a regulator module placed in the BS. The main goal of CAC is to authorize the required bandwidth for each request after checking whether or not it will violate the requirements of other services in the system. The BS responds with an exclusive Service Flow Identifier (SFID) which checks the SFID allocation. The connection will not be established if the available bandwidth cannot meet bandwidth requirements or if the request would impede other services in the network. The CAC mechanism for wireless broadband services was obtained to contain the amount of data for active services such that the associated QoS parameters for each subscriber can be preserved at the desired level.

The CAC algorithms play an important role during the establishment of new connections to control the packet entry and reserve resources for guaranteeing QoS and enhance system throughput.

There are several algorithms that have been proposed to manipulate the traffic in terms of QoS support and different parameters. A survey on the issues and the approaches of call admission control schemes for 4G wireless systems was presented by (Niyato and Hossain, 2005a).

It is assumed that the CAC looks at the QoS requirements that reflect the available resources by the scheduler algorithm such as in the work by (Liu et al., 2004).

2.3 Scheduling Algorithms

The IEEE 802.16 standard specifies the scheduling service but does not specify the techniques that schedule these services. In the following, an explanation of the WiMAX Networks scheduling algorithms will be given.

The main component in the MAC layer that guarantees QoS for different multimedia service classes is the scheduling algorithms. These algorithms are implemented to control the traffic or connection that will be served first before others within the same queue. The priority must adhere to the procedures dictated by the scheduling algorithms which comply with QoS requirements. Indeed, the scheduler plays as a supplier to the connected SSs with the required resources.

Accordingly, an efficient algorithm would be able to improve the QoS and maintain the WiMAX network performance. Furthermore, the concern of these scheduling algorithms is to utilize the bandwidth fully, boost fairness among SSs and maintain the required QoS in the whole network. These scheduling algorithms are performed both in the BS and SSs sides. The main important scheduling algorithms take place in the BS to deal with both DL and UL streams. Therefore, two scheduling algorithms manipulate diverse traffic types located in the BS for both DL and UL sub-frame. However, one scheduling algorithm is placed in the SS to control the UL traffic that is allocated to its requested connections.

It is important to highlight that the UL-MAP and DL-MAP signalling messages are spread in the first portion of the DL sub-frame. Furthermore, the available queue information in BS makes the scheduling resolution for the DL traffic to be comparatively simple when it is not related to the information of PHY layer. The handling of UL traffic is relatively sophisticated where allocation depends on diverse characteristics of multiuser with various connections.

In order to assure efficient scheduling algorithms, specific aspects must be considered and evaluated in the architecture of the scheduling policy. Therefore, designing an efficient scheduling algorithm must consider the QoS parameters of different service classes, Scheduling algorithms need to be simple, efficient, fair, and scalable and have low computational complexity as described below.

Fairness: In addition to supporting QoS requirements for each user, the allocation of available bandwidth should be fairly distributed among the active users.

Channel Utilization: This represents the time slots that are utilized to transmit the required data for the requested packets. The scheduling design should be able to avoid wastage of resources that results from allocating resources for connections that do not have enough data to send.

Complexity: It is important that the scheduling algorithm is designed to be implemented in a simple manner to avoid increasing the complexity when dealing with different types of services constraints.

Scalability: This can be defined as the capability to cope with increased number of SSs in the network. A scheduling algorithm should be stable when the number of SSs increases.

The optimal selection of a scheduling algorithm relies on a combination of architecture mentioned above and its ease of implementation. Moreover, scheduling algorithms should handle multiple connections in order to guarantee the assigned throughputs and delay limit.

As already mentioned, the main role of a scheduling algorithm is to choose which service must be allocated next. Such an allocation procedure mainly depends on the QoS requirements of each service class. Therefore, designing scheduler algorithm at the BS would be more preferable in order to guarantee good performance. Furthermore, current scheduling structures that utilize QoS parameters need to be modified to cope with the particular characteristics, for example, the type of service class and the WiMAX frame structure.

In the following sub-sections, we present other scheduling algorithms and discuss the pros and cons for the state-of-the-art that have been proposed for QoS provisioning in WiMAX.

2.3.1 First-in, first-out (FIFO)

FIFO scheduling is considered the earliest and simplest algorithm, as it accepts packets from all input traffic classes. These are added to a single queue upon arrival and then serviced to the output links on a first come, first served basis. Due to its simplicity, it is suitable for UGS traffic. However, a FIFO scheduler cannot serve one class of traffic differently from any other which gives it a limited functionality (Semeria, 2001).

2.3.2 Round Robin (RR)

The Round Robin (RR) algorithm first designed by Hahne (1986) is another simple approach designed for time-sharing systems which considers priority on queues (Xiaojing, 2007). RR starts by classifying packets into low- and high-priority service classes. Once mapped, these packets are assigned to dedicated queues. Each filled queue is then allowed to send a packet, starting with the highest priority class, without any further priority assignment by the scheduler. RR serves all filled queues in cyclical order; once the queue is served from a given class in the current service round, it is not served again until all unserved queues have been served within a single round. If there are still filled queues, the algorithm moves to a next round until they are emptied. RR is therefore fair when equal packet lengths are used but unfair when variable packet lengths are used by different queues. Moreover, it cannot guarantee different QoS requirements for different service classes (Sarkar and Sachdeva, 2009).

2.3.3 Weighted Round Robin (WRR)

Weighted round robin (WRR) designed by Manolis Katevenis and colleagues (1991) (Manolis Katevenis et al., 1991) was developed to distinguish traffic in different queues to allow several streams to be served. This algorithm works similar to the RR algorithm, except in that it gives a weight to each queue. This weight is proportional to overall share of the available bandwidth used in the system. Therefore, the amount of packets that must be dequeued fluctuates based on the weight given to these queues. Thus, the mechanism of diversity of weights enforces the priority between active queues, and consequently the connected users. However, the disadvantage of WRR algorithm is still the same as RR algorithm, which is unfair allocation of data rates for those connections that are assigned to low priority queues.

2.3.4 Modified Weighted Round Robin (MWRR)

Modified Weighted Round Robin (MWRR) is a variation proposed by (Mardini and Alfool, 2011) to overcome this penalty for lower priority classes. It starts by computing the WRR weight for each queue based on priority and the number of all filled queues. This weighting permits each queue to transmit a certain number of packets in a single service round, which means that the total number of packets a WRR scheduler can deliver per service round is fixed. The MWRR scheduler multiplies each WRR weight counter by a constant integer value in order to increase the service round. This algorithm reduces the average delay and increases average throughput, particularly for lower priority classes, by lengthening the size of the service round over WRR. However, the multiplier used is static, which may lead to either increase in delay or decrease in throughput when it is inappropriately chosen.

2.3.5 Deficit Round Robin (DRR)

This scheduling algorithm mitigates the unfairness associated with RR and WRR algorithms described above (Shreedhar and Varghese, 1996). This scheduling algorithm is associated with a deficit counter (DC) which is originally established to distinguish the quantum of every queue in the system. This quantum is an assigned quantity of bytes that is used by each queue at any time it is allowed to be dequeued. The associated DC is periodically increased by one quantum on each round whenever the scheduler performs the dequeuing process for that queue, unless the queue is empty. These packets are dequeued if the total quantum added to the residual of the previous deficit counter is larger than packet size. Otherwise, this individual queue remains idle until the next visit of the scheduler. Alternatively, when these packets are entirely dequeued, any residual values in the DC will be set to zero., This particular procedure will lead to unfairness when that value is neglected and left without utilization. Cicconetti and colleagues (2006) (Cicconetti et al., 2006) adopted the latency awareness of these conventional algorithms in order to cope with characteristics of IEEE 802.16 schedulers. This decision is mainly motivated by the fact that both rtPS and nrtPS scheduling services entail a basic QoS requirement in term of minimum reserved traffic rate. For instance, for VoIP and other real-time services, DRR remains the most suitable selection among the conventional scheduling algorithms.

2.3.6 Adaptive Weighted Round Robin (AWRR)

Adaptive Weighted Round Robin (AWRR)AWRR is yet another variant, proposed by (Brahmia et al., 2014), again to avoid the problem of starvation of lower priority classes. It uses two schedulers: An input scheduler aims to prioritize video streams with superior quality (HD and SD) and consists of a High-Priority (HP) buffer for, e.g., UGS, ertPS, and rtPS, and a Low-priority Buffer (LB), which includes rtPS-web-TV, rtPS-mobile-TV, nrtPS, and BE.The objective of the output scheduler is to regulate data flows and manage all the service classes. Both schedulers use AWRR to adjust the weighting of the service classes. A threshold value is set for each class, which triggers dynamic weight adjustment whenever a threshold is exceeded. The HP queues have lower thresholds than the LP queues. The input scheduler controls the weights of the HP and LP queues based on the HP traffic load and buffers size. Control of the weights from the queue allows the algorithm to achieve minimum throughput of BE traffic under conditions of network stress and also gives preferential treatment to HP traffic. The dynamic weight is calculated using an algorithm in which the weights need to be positive to ensure minimum throughput for LP traffic, and an arbitrary constant is used to favor HP traffic classes such as UGS and rtPS. It employs a complex calculation to compute the weights when applied in WiMAX multihop networks.

2.3.7 Low Latency-Weighted Round Robin (LL-WRR)

LL-WRR is another variant, proposed by (Patel and Dalal, 2014), that enhances the latency and fairness of real-time services when the number of connections increases. The algorithm introduces a coefficient (γ), instead of a constant integer which is a function of the number of connections present in the network (Mardini and Alfool, 2011),. The coefficient γ is computed at the beginning of the WRR cycle and multiplied by the WRR of each connection. The value of γ varies as the number of connections increases. To keep the latency low, γ is decreased with an increase in the number of connections, because the latency is proportional to γ. The algorithm also improves fairness when the value of γ reduces faster than the growth rate of the connections; then, both the number of packets to be transmitted in a round and the weighted fairness index decrease. While this algorithm is able to achieve low latency and improve fairness, the computation of γ introduces additional complexity.

2.3.8 Weighted Fair Queuing (WFQ)

Weighted Fair Queuing (WFQ) is designed for a packet estimation of the Generalized Processor Sharing (GPS) algorithm. (Parekh and Gallager, 1993). GPS separates a packet into bits and schedules them individually. In this procedure, the WFQ algorithm achieves a higher performance in comparison to the WRR algorithm, particularly when handling variable size packets. The shortcoming of the WFQ algorithm is that it still dequeues the packets that have not even began service within the GPS algorithm, because it ignores the start time of a packet. Several modifications of WFQ has been proposed in the literature. The most effective modification is the Worst-case fair Weighted Fair Queuing (WF2Q) (Bennett and Zhang, 1996). This algorithm retains the delay constraints and accomplishes worst-case fairness. WF2Q utilizes the same characteristics of WFQ by implementing virtual time concept. Unlike WFQ, the WF2Q selects packets with the smallest ending time in the Head-of-Line (HoL) instead of the lowest virtual finishing time of all packets in the queue.

2.3.9 Self-Clocked Fair Queuing (SCFQ)

SCFQ is a fair WiMAX queuing algorithm proposed by (Golestani, 1994).. This algorithm adopts virtual time as the key of allocation opportunity. The packets in each queue are organized upon their finishing time in which higher priority is given to those packets that have smaller completion time to be transmitted in a FIFO manner. Such a system is good at providing fairness between different traffic streams. However, the complexity of this virtual time method can be challenging. For example, since the virtual clock is the reference for all traffic streams, it cannot be reset until all the streams are idle, which in fact seldom happens and this can lead to numerical overflow problems. A further disadvantage is that the selected packets must be sorted in the queue, which inevitably introduces a slight delay due to the swapping procedures.

2.3.10 Strict-Priority (SP)

In Strict-Priority algorithm, the selection order is based on the priority of weight order (Daniele Tarchi et al., 2006),. The packets are first categorized based on the QoS classes and then allocated into different priority queues. The algorithm services the highest priority queue until it is empty, after which, it moves to the next highest priority queue.

This SP algorithm may not be effective in a WiMAX network, because there is no compensation for inadequate bandwidth. Also, this technique is only appropriate for low bandwidth serial lines that currently use static configuration which does not automatically adapt to changing network requirements.

Finally, this process may result in bandwidth starvation for the low priority QoS classes when the packets may not even get forwarded and no guarantee is offered to even one flow.

Table 2.4. Provides a Summary Comparison of the Various Scheduling Algorithms in Terms of their Advantages and Disadvantages.

Table 2.4: Comparison of Scheduling Algorithms with their Advantages and Disadvantages

Scheduling Algorithm	Advantages	Disadvantages
FIFO	Suitable to UGS traffic class, and simple	No differentiation of service classes
RR	Simple, differentiates traffic classes, and fair when using fixed packet size	Unfair when using variable packet lengths and cannot guarantee QoS to various service classes
WRR	Simple, and support throughput requirements and assures QoS guarantee to various service classes, and fair when using equal packet size	Suitable for UGS class, starvation of low priority class, unfair when using variable packet size
MWRR	avoids starvation of low priority service class	Not standard- compliant
DRR	Simple, support variable packet size and low complexity	Not fair on a short time scale
AWRR	Avoids starving low classes	A complex calculation in computing the weight and applied in WiMAX multi-hop networks
LL-WRR	achieves low latency and improves fairness	Computation complexity
WFQ	Weight mechanism guarantee throughput, delay and fairness	Complex
SCFQ	Adopts virtual time that guarantee throughput and delay, providing fairness	Complex
SP	Simple, can meet the delay guarantee	Lower throughput, violates low traffic type

2.4 Related Works on Scheduling Algorithms

The topic of Scheduling techniques has been the subject of wide ranging research in the areas of telecommunications and computing. The initial solutions to the WiMAX schedulers have been developed on the grounds of current scheduling techniques. Studies such (Ghosh et al., 2008, Sayenko et al., 2008, Gkelias and Leung, 2009, Han et al., 2007). have emphasized the effectiveness of current scheduling techniques in the WiMAX networks by considering the variability of the WiMAX wireless characteristics through developing efficient scheduling algorithms. (Ma et al., 2010, Lakkakorpi et al., 2008, Cao and Li, 2001) have proposed that, unlike wired networks, there are particular characteristics of the WiMAX mobile networks that make the scheduling complicated, such as wireless link variability, fairness and QoS support. Various scheduling algorithms are available for WiMAX technology. However, it has been shown that one algorithm is not exclusively superior regarding performance and fulfilling all the QoS requirement. The scheduling algorithms employed in WiMAX are not hard laid protocols, so that any vendor can develop their scheduling techniques as per their need.

The following provides an overview and reviews the literature of comparative studies on scheduling algorithms in WiMAX networks under different simulator environments.

Kanda and Gill's (2015), compare various scheduling schemes using the OPNET simulator with different traffic and packet sizes. The traffic included voice, video, and other data. With different traffic packet sizes; parameters like delay, jitter, throughput was measured. Based on that, they suggested using schemes like MWRR, DWRR, PFS, and MPQ. Their results concluded that there is no such specific scheduling algorithm that best fulfils QoS requirements of the WiMAX IEEE 802.16.

Similar to Kanda and Gill , (Rehman and Shakir, 2014), made comparisons by taking four different scheduling algorithms: Priority Queue (PQ), Weighted Fair Queue (WFQ), Deficit Round Robin (DRR), and Modified Deficit Round Robin (MDRR). Their comparisons were made on the basis of buffer utilisation and delay in the

OPNET simulator. The performance was evaluated in real time scenarios using four types of HTTP under FTP and VoIP applications. They concluded that the MDRR scheduling algorithm outperformed other algorithms, but also pointed out that other algorithms can perform in similar fashion for high priority data traffic but not for low priority data traffic because of the fairness issue.

Another approach was suggested by the research of Murawwat et al's., (2012), The authors discussed performance parameters, applications, and evaluation in detail of various existing scheduling WiMAX algorithms for PMP mode. They took thirteen different schemes of scheduling algorithms for comparison and analysis. Unlike Kanda and Gill, their results concluded that there is no such specific scheduling algorithm among those measured that gives the best QoS requirement complaint to the WiMAX IEEE 802.16.

Using another approach, (Singla and Kamboj, 2012) did a comparative analysis of two different scheduling algorithms, WFQ and PQ. Their comparisons were principally based on the issues like an end to end delay, and traffic received, as well as basic parameters namely VOIP, Video Conferencing and FTP in the OPNET simulator. The results of their simulation showed the distinction and difference between various QoS while using different scheduling algorithms.

Alternatively, a two-level scheduling scheme named TLS was proposed by (Wei Nie et al., 2011). They argue that the advantage of a two-level scheduling scheme is that it has support for fairness and quality of service (QoS) for downlink traffic in WiMAX network compared to RR and WRR algorithms in the OPNET simulator. Their simulation was done with the intention of focusing on the traffic model. Four different traffic were used namely: VoIP, Video streaming, FTP, and HTTP. The QoS for each class was worked out, and differences were noted down for each case. They argue that their results showed that the simulations of QoS Priority and Fairness scheduling scheme have the highest throughput and the minimum delay for high QoS classes while using TLS algorithm compared to round robin and weighted round-robin algorithms.

In addition, the comparative analysis by (Kumar and Garg, 2011), presented a WiMAX simulation model in the OPNET modelling environment based on FIFO, PQ, and WFQ scheduling algorithms. Their simulations aimed to work out end to end delays for sent and received traffic. They also incorporated applications to execute Voice and Video. In contrast to other studies above, Kumar and Garg proposed that different scheduling algorithms provide the difference in terms performance.

The focus of research by (Kaur and Singh, 2011), was PQ and MDRR scheduling algorithms in WiMAX for PMP mode, in which they explain/ demonstrate QoS in WiMAX network. Their simulation was carried in the OPNET simulator, and comparisons and analysis were subsequently worked out. Furthermore, this research was carried out in near to real life scenarios for practicality and precise results. It included applications like data and voice calls with parameters to define QoS like average jitter, throughput, average load, and average delay. Their research established an important distinction between PQ and MDRR scheduling algorithms in that they proposed that the performance of each scheduler supported different QoS classes. The research of Kaur and Singh (2011) concluded that PQ should be regarded as better for throughput, but that it was prone to higher delay than the MDRR scheduling algorithm.

Similarly, (Cicconetti et al., 2007) investigated performance variations in Scheduling algorithms. Their simulation compared the usability and effectiveness of the rtPS, nrtPS and BE WiMAX QoS classes for the scheduling algorithms RR, SP, WFQ and WRR. They found differences in performance based on throughput and delay. I Cicconetti and colleagues (2007) also suggested that delay and throughputs depend on several factors like load partitioning and uplink bandwidth.

The scope of research in scheduling algorithms was further extended by (Akashdeep et al., 2016) who proposed a WiMAX-based network using the concept of fuzzy logic implementing Artificial Intelligence in its design. The researchers analysed queue length and suggested a network with latency guarantee in its design.

Complementary to (Akashdeep et al., 2016) and (Katidiotis et al., 2010) evaluated the performance of artificial neural network-based learning schemes for cognitive radio systems. The researchers suggested that proper management of radio spectrum through the use of neural network concept would lead to the convergence of Artificial Intelligence with the design of an adaptive and sophisticated wireless radio network.

In other studies by (Cheng et al., 2006, Chan et al., 2008, Lakkakorpi et al., 2008), researchers compared the QoS scheduling algorithms with the organizing algorithms classification and found that the RR algorithm would be a good QoS scheduling algorithm while it can be adapted to become an organizing algorithm by incorporating the organization conditions into connections scheduling.

Using the RR and Greedy algorithms as baseline algorithms to develop their approach, (Cheng et al., 2006, Cao et al., 2007) exemplified central and distributed scheduling mechanisms for the WiMAX mesh. As a result of their research, the mini slots allocation was made more flexible with increased utilization. Correspondingly, in Cao’s earlier work (Cao et al., 2005), the stochastic model for the distributed scheduler for the WiMAX mesh had been developed and evaluated (Cao et al., 2007).

There have been several pieces of research into the broad topic of centralized scheduling algorithms. Initially, (Fu et al., 2005) proposed a centralized scheduling mechanism that maximizes the spatial reuse and enhances the network throughput by taking the wireless environment interference into account. In further research in this area, (Liu et al., 2009), proposed slot allocation algorithms under a centralized scheduling scheme for the IEEE 802.16 based wireless mesh networks. The resulting algorithm was proposed for both the UL and DL links. Lui et al found that the algorithm achieved high spatial reuse and throughput and prevented frequent switching between reception and transmission within any two adjacent timeslots.
Shetiya and Sharma (2006) proposed a centralized scheduling algorithm with QoS guarantees for each flow to real and interactive data applications while utilizing the network resources efficiently.

The centralized scheduling algorithm proposed in 2006 by Cao and colleagues (Cao et al., 2006) was developed based on multi-path routing for the WiMAX mesh networks. These researchers considered the load balance, spatial reuse, and QoS demands of the users. Their proposed algorithm focused on the routing issue but did not fully consider the overhead and delay of the multi-hop structure and/or management signalling.

Other scheduling research findings by (Wei et al., 2005) proposed an efficient interference-aware, cross-layer design to increase throughput and utilization of the WiMAX mesh through appropriate design of multi-hop routing and scheduling.
In contrast, (Kim and Ganz, 2005), proposed a transmission scheduling algorithm for the IEEE 802.16 based WMN taking into consideration the fairness of MSs. Such an approach does not address the delay due to the multi-hop nature, and it does not apply to the MR network.

In 2007, (Kwak and Cioffi, 2007) suggested a resource allocation model for OFDMA multi-hop relay systems with fairness constraint for the OFDMA relay-based networks. They argued that this model would optimize the challenges to the source, relay, and subcarrier assignments which maximize the sum-rate from all sources to the destination.

Shetiya and Sharma (Shetiya and Sharma, 2006) proposed a centralized scheduling algorithm with QoS guarantees for each flow to real and interactive data applications while utilizing the network resources efficiently.

A centralized scheduling algorithm was proposed in 2006 by Cao and colleagues (Cao et al., 2006). It was developed based on multi-path routing for the WiMAX mesh networks. These researchers considered the load balance, spatial reuse, and QoS demands of the users. This algorithm focused on the routing issue and did not consider the overhead and delay of the multi-hop structure and/or management signalling.

In tandem, there is a body of research in the topic area of scheduling both in the fixed and mesh WiMAX networks including (Han et al., 2007, Peng et al., 2006, Du et al., 2007, Kim and Ganz, 2005, Lee et al., 2010). (Lee et al., 2010), proposed a scheme for realizing throughput fairness in a multi-hop WMN. Their scheme is highly scalable due to the fully distributed technique, which requires no global information. This algorithm has distributed weight estimation and channel scheduling procedures, which increase the performance of the system greatly. A central feature of most of studies of fixed and mesh WiMAX networks cited above studies gave higher priority to the subscriber's rather than to the scheduling station, especially, when slots are allocated to SSs/MSs. Of these, a few studies concentrated on the limitation of the spatial reuse, channel-aware fairness and other issues, which may restrict the QoS guarantees slightly (Şekercioğlu et al., 2009). However, such scheduling algorithms will result in the relay nodes with high delay consuming much of the time resources which will lead to degradation in the performance of the WiMAX MR networks, which may explain why this avenue is not highly researched.

On the other hand, significantly more research is focused on the purpose of optimizing, configuring, healing, and planning for the WiMAX and LTE 4G networks (ITU-R, 2009) and several schemes have been proposed (Dixit, 2004, WiMAX-Part, 2006, Anceaume et al., 2005, Ho et al., 2003). Self-organization principles were designed to support the automation of the tasks mentioned previously. The concept of a self-organizing network (SON) has been proposed as a solution to minimize operational effort, reduce operational costs, and provide quick response to customer needs. However, the challenging issues concomitant to a deployment of the WiMAX, such as interference mitigation, mobility management, and self-organizing network were highlighted (Niyato and Hossain, 2006, Pareek, 2006).

The SON plays an important role in the 4G networks and, in consequence, related research and standardization is being carried out at a fast pace to determine the requirements of SON (Alliance, December 2008) and suggest the best use of SONs (Alliance, May 2007). As this is a promising avenue, more realistic and focused research regarding the SON is needed. Reduction of cost and complexity is an important key driver for the 4G networks. It is of vital interest of related research and standardization efforts to minimize operational effort and cost by producing self-configuring and self-optimizing mechanisms (Alliance, May 2007). However, the automation of such tasks heads towards self-organizing features of next generation wireless networks, hence enhancing network performance and quality. The SON not only increases the network performance for the end users, but also improves network operability by reacting on dynamic network variation processes.

A substantial body of research has been built around Scheduling Algorithms. Abu Ali and colleagues (2009) classified them into four types:

Homogeneous algorithms are classical algorithms that attempt to provide solutions to challenges such as QoS guarantee, flow isolation, and fairness; they use a single type of scheduling algorithm for all WiMAX traffic. They can also be flexible to comply with both hybrid and hierarchical structures. Initially, these algorithms were intended for wired networks, but were later adopted by the majority of broadband services to manage the QoS requirements.

Hybrid algorithms combine more than one type of classical algorithm to provide QoS requirements for the different kinds of traffic in a WiMAX network. Some of these algorithms also provide a solution to the variable channel conditions in a WiMAX thus resulting in superior performance. One major aspect of these algorithms is the overall allocation of bandwidth among the scheduling services. Once bandwidth has been allocated to each class, a classical algorithm is executed for the SSs of that class to determine the bandwidth allocation within that class.

Opportunistic algorithms are algorithms that employ adaptive modulation and a coding scheme at the PHY layer in order to exploit the variability in channel conditions in WiMAX.

Hierarchical schedulers are scheduling techniques employed at multiple hierarchical levels in order to meet the needs of several service classes. The bandwidth is allocated in a certain way at the first hierarchy level to the associated service class and usually inherits conventional schemes to schedule different connections within each service class.

Consistent with the classification by Abu Ali, (Daniele Tarchi et al., 2006), propose a hybrid structure algorithm for UL direction, where UGS is scheduled by Packet Based RR, which is appropriate for steady allocation of bandwidth. In this work the Earlier Deadline First (EDF) is proposed for the rtPS service type, which is appropriate for real-time services that are sensitive to delay. The WFQ algorithm is proposed for non-real-time services such as nrtPS and BE which is capable of reserving suitable weights for such service classes. In further studies, (Esmailpour and Nasser, 2011), propose a dynamic QoS-based bandwidth allocation , to support heterogeneous traffic with different QoS requirements in WiMAX networks. The allocation bandwidth is dynamically adjusted for ongoing and new arrival connection based on traffic characteristics and service demand in order to maximize the system capacity.

(K. Vinay et al., 2006), proposed an enhancement implemented by combined EDF and WFQ to serve real-time application. (Kitti Wongthavarawat and Aura Ganz, 2003a) presented a constant allocation for real-time services using EDF, whereas WFQ and equal sharing are used to schedule connections of different service classes. Their results propose that hybrid algorithms, lead to higher complexity either by their structure implementation or huge resource consumptions. Furthermore, (Kitti Wongthavarawat and Aura Ganz, 2003b), were one of the first to work on a hierarchical algorithm in WiMAX systems for resource Uplink Packet Scheduling assignments.

(Jianfeng Chen et al., 2005) used two-tier hierarchical scheduling: In the first tier Deficit Fair Priority Queuing (DFPQ) assigns the total available bandwidth for DL and UL services. The main objective of DFPQ is ensuring fairness to the different

service class and to improve total system throughput. Additionally, DFPQ modify DL and UL bandwidth dynamically to improve system utilization.
(Haidar Safa et al., 2007) suggested the enhancement of DFPQ algorithm to improve the performance of real-time services by considering a deadline constraint. DFPQ cannot guarantee QoS requirements for real-time services such as VoIP and video streaming and through introducing a deadline constraint latency of real-time service is minimized. One downside of this approach is that it limits the opportunity for non-real-time services such as nrtPS and BE.

Several areas of research explore various models associated with Abu Ali's initial four classifications:

(Ikbal Chammakhi Msadaa et al., 2007) proposed new QoS scheduling architecture based on DFPQ. Their method gives more bandwidth to rtPS at the expenses of Best Effort (BE) traffic.

(Shang and Cheng, 2005) proposed a hierarchical model for UL packet scheduling for WiMAX system. This algorithm is based on the so-called soft-QoS and hard-QoS structure. The soft-QoS is for rtPS and nrtPS in which their QoS parameters fluctuate between the minimum bandwidth requirements and maximum bandwidth required, while the UGS service class is served under hard-QoS. This structure can allocate the available bandwidth among BE and other service classes effectively and at the same time ensures fairness among them.

(Sun et al., 2006) proposed two QoS schedulers located at BS and SS sides. Both schedulers give higher priority to UGS real-time services such as ertPS and rtPS during the connection setup. For rtPS services it guarantees required bandwidth by calculating the deadline based on arrival time. However, a fixed priority assigned to nrtPs and BE starves their connection when more rtPS connections existed in the network.

(Xiaojing, 2007) proposed Adaptive Proportional Fairness scheduling algorithm. It is designed to extend the PF scheduling algorithm to real-time services and satisfies various QoS requirements. The Adaptive Proportional Fairness algorithm successfully attempts to differentiate the delay performance of each queue based on the Grant per Type-of- Service (GPTS) principle.

(Wei Nie et al., 2011) proposed a QoS priority and fairness scheduling scheme for DL traffic which guarantees the delay requirements of UGS, ertPS and rtPS service classes. This is a two-level scheduling scheme that maximizes the BE traffic throughput. A strict priority between service classes is adopted on the first level. UGS is treated on the second level by assigning a fixed rate. Then the Adaptive Proportional Fairness algorithm allocates bandwidth for rtPS and ertPS service classes. For BE, Proportional Fairness algorithm is applied to treat the non-real-time application for nrtPS and BE service classes.

(Kitti Wongthavarawat and Aura Ganz, 2003a) defined three levels of priority. While prioritizing additional rtPS flow solves the problem of interrupting rtPS packets, it also leads rtPS flows to gain arbitrarily large free access at the expense of BE and nrtPS flows. To overcome this, shortcoming they considered a Customize Deficit Round Robin with an additional queue. In this way, only real-time packets which are just prior to the deadline are considered by the added queue (Laias and Awan, 2010). However, using an additional queue for rtPS connections causes additional access latency and bandwidth allocation issues and the flow becomes backlogged. This is undesirable especially if the flow is of type-high priority.

Several other scheduling algorithms with different design goals have been developed and proposed by (Lin et al., 2008, Deng et al., 2009, Hsieh et al., 2012). However, up-to-date trivial algorithms have been used to create effective scheduler structure with significant performance in the two-tier hierarchical scheduler, which requires further optimization.

(Rath et al., 2006) proposed the Opportunistic Deficit Round Robin scheduler, which is an analytical method for getting an optimal polling interval for UL data traffic via the polling interval mechanism, the BS polls service traffic periodically to make sure that the traffic delays are achieved. The system considering several situations, for instance, the SSs must ensure that the queue should not be empty as well as the receive Signal to Noise Ratio must exceed the threshold value. However, the allocation mechanism of the Opportunistic Deficit Round Robin algorithm leads to an additional overhead at the BS because it requires the manipulation of quantum size and a DC for each SS, repeatedly.

(Ball et al., 2005) proposed a scheduling algorithm for the rtPS. This algorithm manipulates a scheduling list that contains all the SSs that can be served at the next frame. However, the algorithm specifies that the SSs that have low transmission quality are suspended temporarily from the transmission list. This mechanism is repeated periodically for all SSs. If the transmission quality is still low, the scheduler grants another suspended period of time.

(Gan et al., 2009) designed a cross-layer scheduling algorithm to cope with the features of UL traffic in the WiMAX system. This algorithm is referred to as dynamic Modulation and Coding Scheme and Interference Aware Scheduling. The main feature for its structure is taking into account the queue status, the status of the channel and the QoS parameters of service type queue. (Gan et al., 2009) identified the main role of this algorithm as to improve the total throughput, besides sustaining the QoS requirement of diverse classes. Gan et al concluded that it dealt well with optimization.

(Niyato and Hossain, 2005b) proposed a queue-aware algorithm for UL direction in the SS side. The algorithm defined set thresholds for bandwidth allocation for the connected services. This is to recognize the required bandwidth to be allocated in the UL subframe. Specifically, the sum of bandwidth assigned to the polling service is considered as a function of the amount of PDUs in a queue. However, in this work it is estimated that the period amongst successive thresholds in the set is equal.

(Lin et al., 2008) proposed a latency and modulation aware bandwidth allocation algorithm called Highest Urgency First. Highest Urgency First converts the incoming data rates into time slots in order to determine the influence of several Modulation and Coding Schemes. However, Modulation and Coding Scheme diversity does not fully exploit this. Furthermore, Highest Urgency First procedure is forcing the request that is approaching its deadline to be discarded.

(Laias and Awan, 2010) proposed a Customized Deficit Round Robin The algorithm takes care of real-time service by adding a new queue to schedule real-time connections just prior to the deadline. The downside is that, this extra queue increases the delay for non-real-time connections such as nrtPS and BE. The interruption caused by the transmission of non-real-time packets in the extra queue will degrade the overall system throughput and violate the real-time connection deadline for the packets in the rtPS queues. This is due to the interception of the extra queue for the real-time signal, which leads to increased overhead for the system, which is not desirable, in particular, when the real-time traffic is high.

(Tarhini and Chahed, 2012), proposed a resource allocation strategy to allocate more subcarriers to users that experience bad channels in order to maximize its data rate. Consequently, this increases the dropping of resources for users that experience good channels such as those that are located near a BS.

(Hua Wang and Lars Dittmann, 2010) proposed a downlink resource management framework on a hierarchical resource allocation model; this involves the differentiation of service type and aggregate resource allocation based on users' respective priorities.

One way of tackling the scheduling problem in multicarrier systems is to focus on fairness in resource allocation, which has been discussed by a number of researchers (Ergen et al., 2003, Nguyen and Han, 2006, Girici et al., 2010).

(Choi et al., 2008) proposed a scheme to maximize system throughput while satisfying QoS requirements of real-time and BE services. While this strategy improves system throughput, it can cause starvation of other users with bad channels due to insufficient exploitation of channel variations. This results in the degradation of system performance and violates fairness.

(Ali-Yahiya et al., 2010) investigated a different strategy of resource allocation for OFDMA frames. They studied two strategies for simple allocation algorithms: Adaptive Slot Allocation and Reservation-based Slot Allocation (RSA). Adaptive Slot Allocation procedure is unfair for non-real-time application because the real-time application is satisfied first, and this violates the overall throughput for this capacity.

As discussed earlier, the IEEE 802.16 standard stipulates the signalling mechanisms of QoS and the scheduling services classes to cope with the various application requirements. Table 2.5 shows the existing IEEE 802.16 QoS support algorithms that are implemented at MAC Layer, which includes state of the-art algorithms that are executed at the BS. More details about their characteristic and their contribution and drawbacks of supporting multimedia traffic are discussed.

Table 2.5: Summary of WiMAX Scheduling Algorithms

Papers	**Objective**	**Key Idea**	**Limitations**
Daniele Tarchi et al., 2006	A hybrid structure algorithm for UL direction	Scheduling UGS by PBRR, EDF for the rtPS and WFQ for nrtPS and BE	Complex due to depending on different algorithms to handle their QoS requirements
K. Vinay et al., 2006	Allocating UL bandwidth traffic to give better performance for rtPS	Combined EDF and WFQ to serve rtPS	Complex due to combining two algorithms.

Kitti Wongthavarawat and Aura Ganz, 2003a, Kitti Wongthavarawat and Aura Ganz, 2003b	Enhancing the system throughput and fairness to queues	Constant allocation for rtPS using EDF, whereas WFQ and Equal Sharing are used to schedule nrtPS.	Hybrid algorithms lead to higher complexity
Jianfeng Chen et al., 2005	Ensuring fairness to different service classes and improving throughput	Two-Tier hierarchical scheduling for DL and UL services	Unable to guarantee QoS requirements for rtPS services
Haidar Safa et al., 2007	Improving the performance of real-time services	Modify UL and DL bandwidth Dynamically by considering a deadline constraint	Less suitable for non-real-time services such as nrtPS and BE
Ikbal Chammakhi Msadaa et al., 2007	Minimize the latency of real-time connection	Guarantee QoS requirements for real-time services	Gives more bandwidth to rtPS at the expenses of BE traffic
Shang and Cheng, 2005	Allocating the available Bandwidth among BE.	This algorithm is based on the soft- QoS and hard-QoS structure	Complex and unfair for overall system traffic
Sun et al., 2006	Guaranteeing required bandwidth	Two QoS schedulers located at BS and SS to Gives higher priority to UGS, ertPS and rtPS during the connection setup	Fixed priority assigned to nrtPs and BE starved their connection when more rtPS connections existed in the network
Xiaojing, 2007	Extending the PF scheduling algorithm to real-time services and satisfies various QoS requirements	Differentiating the delay performance of each queue based on GPTS principle	Applied to treat the non-real-time application for nrtPS and BE

Wei Nie et al., 2011	Guaranteing the delay requirements of UGS, ertPS and rtPS in downlink traffic	QoS priority and fairness scheduling	Does not support nrtPS and BE
(Rath et al., 2006)	Making sure that the traffic delays are achieved	Getting an optimal polling interval for uplink data flow via the polling interval mechanism	The allocation mechanism of the algorithm leads to an additional overhead at the BS

The research literature on scheduling algorithms analysed in the table above identifies key objectives and outcomes in WiMAX. It also points to an important omission in current research which, as can be seen in the table. has been limited in scope and does not quantitatively address the issue of the Portfolio Algorithm which is the principal focus of this thesis.

Importantly, Kanda and Gill (2015) who compared various scheduling schemes concluded that a *single algorithm* will not give the *optimal performance* to meet the required QoS for different types of application traffic of the WiMAX IEEE 802.16. The majority of current research has focussed on enhancement or the creation of a new single algorithm which was not capable of superior performance. Thus, this thesis addresses an important gap in current research literature which will contribute to the body of research in this area.

Many scheduling algorithms are available for WiMAX technology and have been extensively evaluated but it is recognized that a *single algorithm* will not give the *optimal performance* to meet the required QoS for different types of application traffic. (Kanda and Gill 2015).

2.5 Summary

This chapter has provided background information on relevant topics to facilitate the discussion on the design of traffic scheduling algorithms for BWA networks in subsequent chapters. First, an overview of BWA networks was presented, focusing on the QoS architecture of the IEEE 802.16 standard as well as the MAC and PHY layers. Then, several essential concepts for designing wireless scheduling algorithms were explained, which include wireless channel modelling, Adaptive modulation and coding technique and multiuser diversity and the challenges and ideas for traffic scheduling with the wireless channels. The next chapter describes the methodology and procedures adopted in the current study.

CHAPTER 3

RESEARCH METHODOLOGY

3.1 Introduction

As highlighted in Chapter 1, the main aim of this thesis is to propose Portfolio algorithms that support QoS by making the network auto-configurable and flexible in order to maximize overall network throughput, as well as being customized to produce less time delay over the real-time and non-real-time services based on IEEE 802.16 Wireless Broadband Networks. This will be done by considering the channel response to swapping the scheduling connections with the available bandwidth using efficient and reliable algorithms.

This thesis proposes two new methods of Portfolio algorithms:

(i) *Fixed Portfolio Algorithms* which is a fixed series of algorithms, and

(ii) *Portfolio Peak Algorithms (PPA)* which is described as an algorithm for data traffic control in high-speed WiMAX networks by switching algorithms based on peak performance of each existing algorithm.

Comparisons of Portfolio performance against the performance of previous scheduling algorithms are done after 450 of a multi-experiment and implementing different scenarios based on the main network parameters. It also describes the research framework, which shows the System Description and Parameters of all the three case studies. Furthermore, it will be established how the proposed Portfolio in both methods can achieve the best performance.

3.2 Research Framework

The initial stages involved formulating the research question and defining the parameters of inquiry. Next, the literature was consulted to discover the scope of the study. Based on this, different portfolios were proposed and subsequently tested and evaluated. Figure 3.1 presents an outline of this process.

The research proposes Portfolio algorithms, which will be evaluated against conventional algorithms in section 4.3 and the performance results of comparative algorithms shown in section 4.4 that adopt the basic allocation mechanisms which have been evaluated in the NS2 simulator such as: Droptail, Deficit Round Robin (DRR), Fair Queueing (FQ), Random Early Drop (RED), Stochastic Fair Queueing (SFQ).

3.2.1 Problem Formulation

Throughout the initial stage of research design, existing methods of scheduling and Connection Admission Control (CAC) were reviewed that attempt to reduce the delay and improve the throughput in mobile WiMAX networks. Additionally, we discuss in detail schemes that exploit the allocation optimization in order to minimize wastage of bandwidth and improve the overall network throughput.

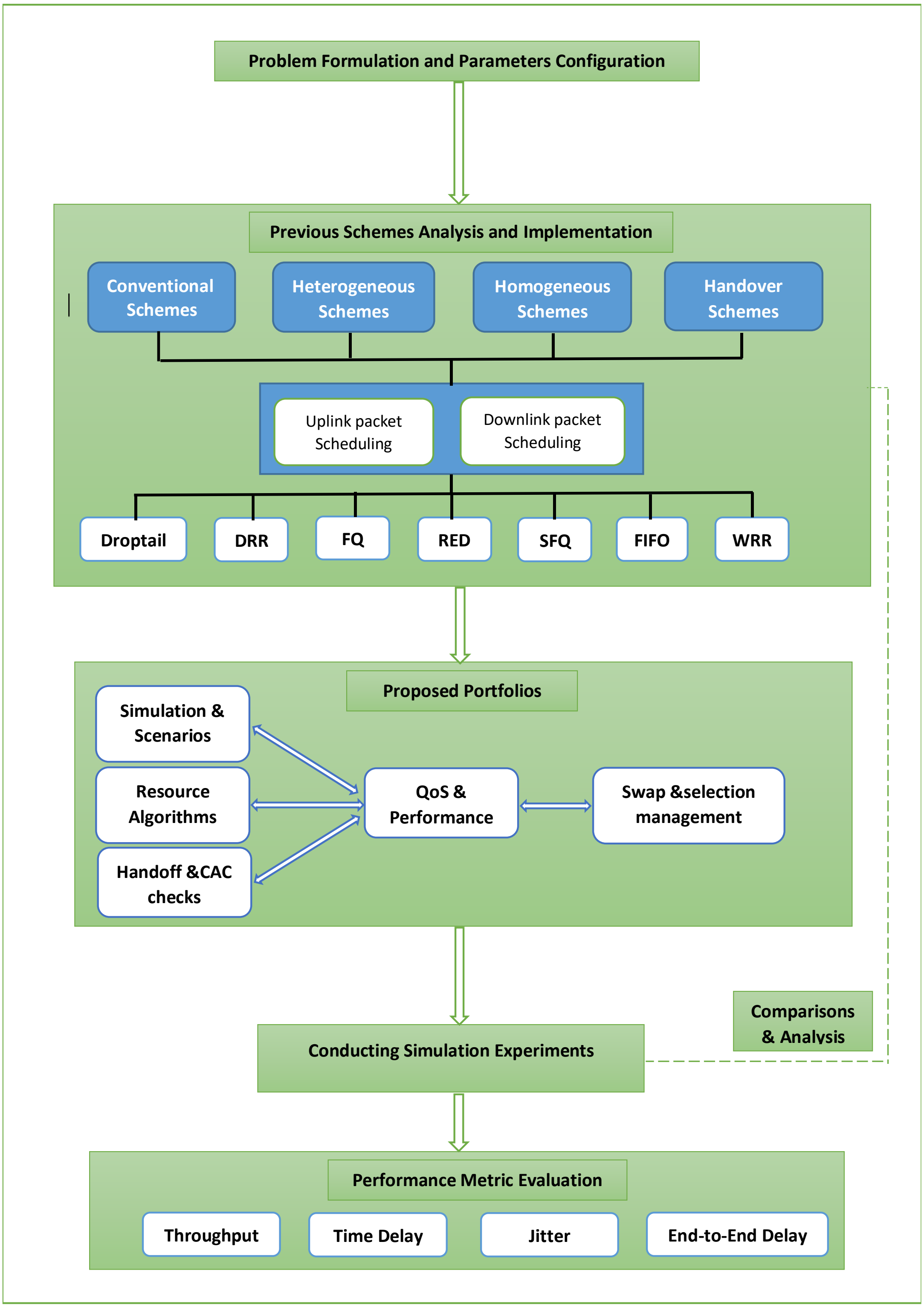


Figure 3.1: The Research Framework

3.2.2 Design Challenges

An important problem in a mobile WiMAX network is finding and maintaining routes since host mobility can cause topology changes. Several routing algorithms for WiMAX have been proposed in the literature and they differ in the way new routes are found and existing ones modified.

The design of algorithms for WiMAX poses new and interesting research challenges, some of which are particular to mobile WiMAX networks. Algorithms for a WiMAX must self-configure in order to adjust to the environment and traffic where they run and goal changes must be posed from the user and application. This is why the current research looked in particular into proposing a portfolio of algorithms, which will control and update transmission with the required algorithm by the various portfolios for supporting QoS such as; the guarantee of a maximum throughput for real-time traffic.

Data communication in a WiMAX network differs from that of wired networks in several aspects. The wireless communication medium does not have predictable behaviour, as in a wired channel. On the contrary, the wireless communication medium has variable and unpredictable characteristics. The signal strength and propagation delay may vary with respect to time and the environment where the mobile nodes are. Unlike a wired network, the wireless medium is a broadcast medium, that is; all nodes in the transmission range of a transmitting device can receive a message.

An important challenge in the design of algorithms for a WiMAX network is the fact that its topology is dynamic. Since the nodes are mobile, the network topology may change rapidly and unexpectedly, thereby affecting the availability of routing paths. Given all these differences, the design of algorithms for WiMAX networks are more complex than their wired counterpart.

3.2.3 The Proposed Algorithms

Reasoning for the algorithms

The proposed Portfolio algorithms in this thesis put emphasis on the provision to QoS and fairness for mobile users through seamlessly wide resource allocation for diverse applications and conditions in the network. They were designed to enhance packet flow by allocating scheduling algorithms on demand fairly between different users of that bandwidth in terms of robustness and efficiency.

Details of the algorithms:

The following will explain the two proposed algorithms in detail.

(i) *Fixed Portfolio Algorithms* which are a fixed series of algorithms, and

(ii) *Portfolio Peak Algorithms (PPA)* which are described as an algorithms for data traffic control in high-speed WiMAX networks by switching algorithms based on peak performance of each existing algorithm.

The algorithms have the ability to interact with each other. They can efficiently select bandwidth and utilization, whilst at the same time, guaranteeing QoS requirements and user fairness.

Once bandwidth has been allocated to diverse sets of users and their corresponding UL and DL flows have been scheduled, the system will dynamically reset the initial arrival rates. The system then starts a new set of calculations for bandwidth allocation and scheduling steps based on the optimal values and QoS of each scheduling algorithm and by considering the optimization of frequency and time dimensions.

Following that, a dynamic weight is introduced based on the traffic load of each queue and the static weight of each schedule to increase the service rate in each round. The increase in service rate reduces the effect of the burst traffic and therefore the algorithm not only reduces the average delay and packet loss but also improves average throughput. The advantage of Portfolio algorithms is that they make it very efficient in terms of full utilization and supporting seamless and effective bandwidth resource allocation of a scheduling algorithm.

The performance evaluation of scheduling algorithms plays a crucial role in the design of a WiMAX system. Several performance analysis techniques exist for the performance evaluation of WiMAX networks. These techniques consist of an experiment, computer simulation, and mathematical analysis (Kassab et al., 2009). By definition, an experiment means conducting experiments with actual test beds and is the most suitable means for monitoring the status of a system for a specific network configuration. Although the technique provides the most accurate performance assessment, building test beds are costly. Reconfiguring and sharing them is complex and they are relatively inflexible (Breslau et al., 2000). Furthermore, it can be difficult to replicate some network phenomena, such as wireless interference, signal attenuation, noise, and fading. A mathematical analysis uses a simple mathematical model. This technique provides a theoretical background for a given technology but only provides limited insight into the technology. It has a lower cost and requires less effort than the other two techniques. However, mathematical models often become intractable (Wehrle et al., 2010). Computer simulation provides a real-world process where the system is ‘imitated’ over time (Law, 2008) (Jerry, 2005). This technique produces replicable results (Kassab et al., 2009). Thus, simulation has become a powerful technique used by researchers in conducting performance evaluations of emerging technologies (Kassab et al., 2009) (Kim et al., 2008).

As mentioned in chapter 2, there are many ways of testing algorithms. Due to ease of use, cost and replicability, the current research employed computer simulation to test the Portfolio algorithms.

Figure 3.2 illustrates the proposed Portfolio mechanism which is also summarized by the following steps:

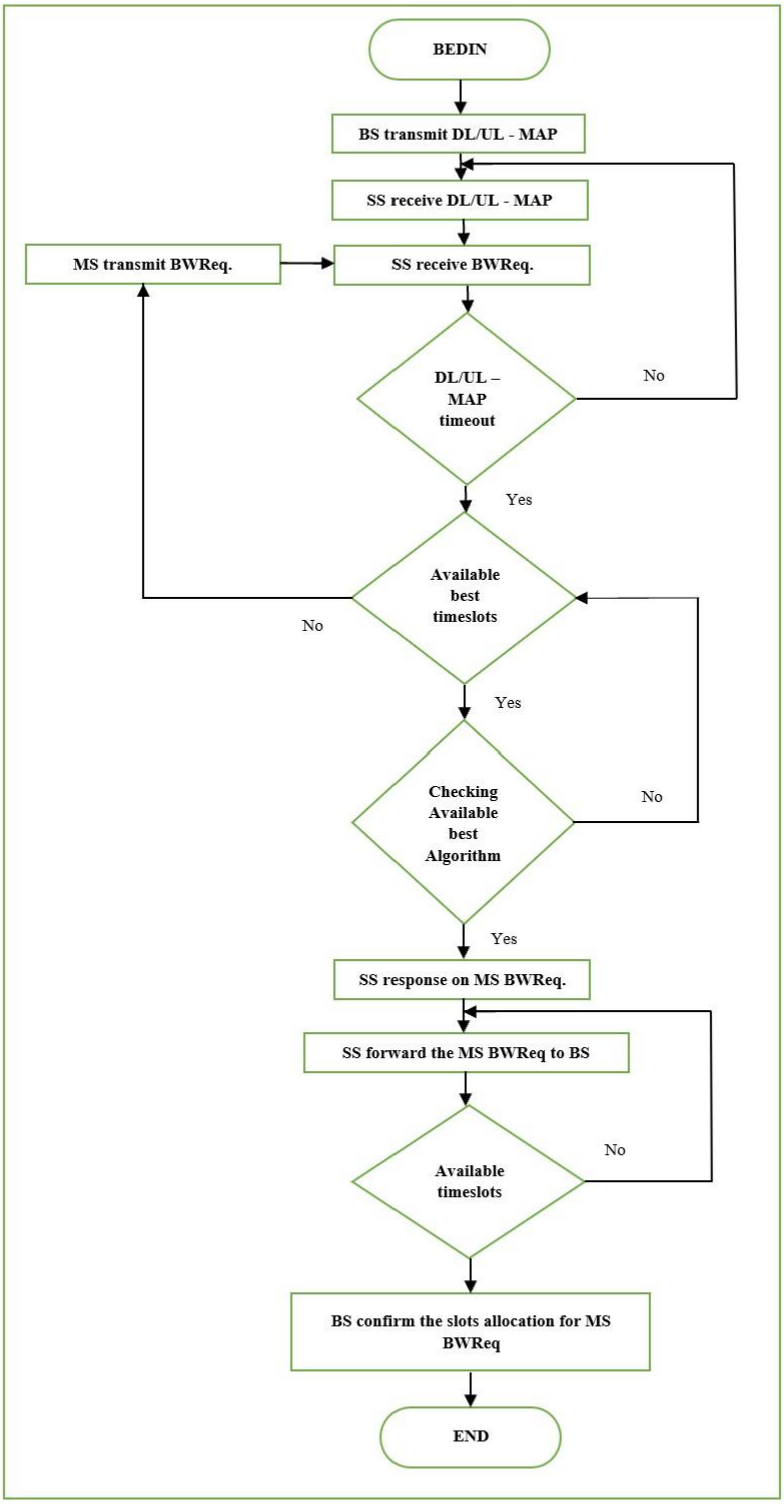


Figure 3.2: The Proposed Portfolio Mechanism

Step 1: $MS\ ai$ sends the bandwidth request message to its superordinate SS.

Step 2: Upon reception of the bandwidth request message from the subordinate MS, and based on the latest MAP message received from the BS, the SS grants the requested bandwidth for all MSs by sending the grant message $BG{,}ai$.

Step 3: Right after sending the grant message, the SS forwards the bandwidth request message $Bd{,}ai$ or $Bu{,}ai$ to the BS for it to allocate the desired bandwidth.

Step 4: Upon reception of the bandwidth grant messages from the superordinate SS, the MS starts functioning, i.e., transmits its data to SS.

Step 5: After the BS performs the bandwidth allocation signalling, based on Step 2, the BS sends the confirmation message to SS.

Step 6: After Step 4 and Step 5 are performed successfully, the SS forwards the received bandwidth request message from MS up to the BS; otherwise, the process is directed to Step 3.

The first proposed algorithm is the fixed portfolio. This is a combination of five conventional algorithms. The first algorithm works from the beginning to a time of 10 to 20 second, the second algorithm from 20 to 30 seconds, the third algorithm from 30 to 40 second, the fourth algorithm from 40 to 50 second, the fifth algorithm from 50 to 60 second. The series of the first portfolio algorithm is (Droptail, DRR, FQ, RED and SFQ), The series of the second portfolio algorithm is (FQ, RED, DRR, SFQ, and Droptail,), The series of the third portfolio algorithm is (SFQ, FQ, RED, DRR, and Droptail,), The series of the fourth portfolio algorithm is (FQ, Droptail, RED, SFQ, and DRR).

The second proposed algorithm is the Portfolio Peak. Portfolio Peak Algorithms are described as algorithms for data traffic control in high-speed WiMAX networks by switching algorithms based on the peak performance of each existing algorithm. In this way, the portfolio maintains the multiplexing flexibility of algorithms, switching, while ensuring the highest throughput rate at the same time. Evidence from scenarios clearly supports the hypothesis.

These algorithms are both designed to maximise throughput and minimise delay. The design of the peak portfolio enables it to outperform the Fixed Portfolio because of its capability to select the best performance.

As indicated in section (4.5 Results and Discussions) the scheduling selection process of each algorithm differs. In the case of the Fixed Portfolio algorithm, the selection is based on the series of algorithms assumed. In contrast, in section (5.4 Results and Discussions) the scheduling selection for the Peak Portfolio allows for the choice of the best available algorithm and has the flexibility to switch selection between the best available algorithms at the time, for example, performance showing high and sometimes showing low based on specific circumstances.

3.3 System Models

This section describes the models used to design and implement the proposed Portfolio algorithms in regards to channel capacity, queuing, traffic and network.

3.3.1 Channel Capacity Model

Channel capacity of users is time-varying and location-dependent due to fading and shadowing; the so-called multi-user diversity effect. Since users' channels fade independently at any given time, some subset of users will likely have strong channel conditions. The scheduling takes advantage of the instantaneous channel conditions to select and swap to the best scheduling algorithm of each user. The channel capacity of the Portfolio algorithms in frequency-time dimensions formulated in this thesis is based on the well-known performance and behaviours of existing WiMAX algorithms.

3.3.2 Queuing Model

In many applications, the variability in arrival and service processes are essential to the performance of the system. Queuing models help us to understand and quantify the effect of variability and the process of packets for connections. The inter-arrival times are independent and arrive according to a Poisson stream in which the Poisson assumption provides lower bounds on the delays under more bursts and correlated traffic models.

Priority queuing disciplines were used to differentiate and classify each arrival into several categories of service. Each category is then given a priority level. We define the time of arrival as the moment when the user arrives. We assume that this is an independent, continuous and random variable which is here described by the random variable A and the arrival rate by the variable λ.

The assumption that each inter-arrival time is governed by the same random variable implies that the distribution of arrivals is independent of time. We can show that the average or mean inter-arrival time is given by:

$$E(A) = \frac{1}{\lambda}$$

In this modelling, it is assumed that the connection has already been established among SS and BS through a Call Admission Control policy and that different SSs have generated multi-rate traffic. This traffic is categorized according to the rules of IEEE 802.16 standard based on data rate and bandwidth necessities to service type classes.

A queue is represented by an arrival process, waiting list requests and the discipline by which these requests are processed. In WiMAX, the BS needs to schedule packets for a limited number of SSs. A BS scheduler consists of three components: a packet classifier, a buffer management and a packet scheduler. The packet classifier receives the arrival packets and distributes them to the corresponding user's buffer management according to the user identification number and QoS requirement.

The buffer manager comprises several different types of data buffers (classified buffers), with subsidiary QoS statistics (e.g., delay deadline and Head-Of-Line (HOL) packet arrival time) being recorded for each buffer. The packet scheduler schedules each user according to scheduling priorities, user channel state information and QoS statistics recorded in each buffer management.

3.3.3 Traffic Model

Much research on traffic modelling has been conducted to investigate the characteristics of various traffic sources for different communication networks, which is summarized in chapter two. In this work, the results of the previous researches were discussed in the literature review in chapter two.

The bandwidth management and Portfolio algorithms were modelled with non real time traffic (e.g. FTP in nrtPS and HTTP as BE class) and real time traffic (VoIP for rtps and ertps) where VoIP traffic is modelled as two-state Markov ON/OFF source. Specifically, the flow of real-time and non-real-time traffic which are directed to their corresponding service class queues respectively. The transmission mode is determined by the instantaneous signal to noise ratio. To utilize the PHY layer resources more efficiently, fragmentation at the MAC layer is enabled. A separate queue with a finite queue length of MAC protocol data units (PDUs) is maintained for each connection at the base station. We assume that the MAC PDUs are of fixed size, each of which contains information bits.

Traffic models of WiMAX applications and their traffic features fall into two broad categories. The real-time traffic type such as VoIP for Real-Time Polling Service (rtps), video conference for ertps and non-real-time traffic types such as File Transfer Protocol (FTP) served in Non-Real-Time Polling Service (nrtPS) class and HTTP evaluated as Best Effort (BE) class as shown previously in Table 2.2.

Each subscriber can send one or more applications to the BS. A video conference session consists of a VoIP and a video session where VoIP traffic is modeled as a two-state Markov ON/OFF source.

A traffic generator generates packets coming from the application server destined for the SS. These packets are categorized into five different traffic flows: VOIP without silence suppression, VOIP with silence suppression, MPEG, FTP, and HTTP. All generated packets are forwarded to the Packet classifier for classification into the various types of scheduling services.

3.3.4 Network Model

The network model topology consists of a single BS and a number of mobile nodes. The service priorities from the upper layers are classified in the BS into corresponding queues associated with their QoS requirements. This is done through the Connection Identifier for five specific service types.

At the PHY layer, different transmission modes are controlled by the adaptive modulation and coding (AMC) which adapts the transmission mode based on the channel information feedback from the transmitter through a feedback channel. These feedback signals are generated at the BS to select the appropriate transmission mode according to the instantaneous signal to noise ratio (SNR) (Hua Wang and Lars Dittmann, 2010).

3.4 Experimental Environments

3.4.1 Simulation Experiments

In this thesis, 450 simulation experiments have been performed in order to study the performance of the proposed Portfolio algorithms and compared them with other conventional scheduling algorithms. The simulation experiments were performed under different network scenarios as well as different settings and the various numbers of mobile nodes in NS2 simulator. However, for the scheduling algorithm in the Portfolio, a fixed series has been assumed. The adaptive Portfolio Peak has been efficiently evaluated on parameters and different settings. The proposed algorithms were tested and explored in simulations and experiments. The simulations focus on how to design Portfolios in NS2 using a general programming language (C++) including (Otcl/Tcl) scripting programming language.

To analyse the performance of the proposed schemes, several resources and parameter settings have been identified. The following section describes the experiment environment that is used throughout this thesis such as software and hardware resources, WiMAX topologies and the experimental setup.

3.4.2 Parameter settings

The presented algorithms are implemented using (Otcl/tcl) scripting programming language and (C++) programming language in NS2 Simulator version 2.35. The operating system is Linux Ubuntu 14.10 in 64-bit which run on an Intel machine i5-5490 CPU with 3.30 GHz processor and 8.00 GB of RAM. The results are depicted and data extracting using excel files and GraphPad Prism software. The compilers Otcl interpreter and Visual C++ are used to create the new proposed scheduling algorithms.

3.5 Performance Metrics Evaluation

The main goal of the proposed packet scheduling and Portfolios is to enhance system performance by reducing the delay and increase network throughput.

These enhancements can be exhibited by maintaining QoS requirements which are mainly based on user demands, as well as those of a service provider. The performance metrics are measured and benchmarked against related scheduling algorithms. Since we are focusing on the QoS improvements at both MAC layer and PHY layer, the proposed performance measures are Throughput, Delay, and Jitter.

Additionally, the proposed algorithms were implemented and compared with the Droptail, Deficit Round Robin (DRR), Fair Queueing (FQ), Random Early Drop (RED), Stochastic Fair Queueing (SFQ) algorithms using the NS2 simulator. The scope of this chapter focused on the scheduling for the IEEE 802.16 network in terms of performance metrics and the best QoS performance.

Throughput

Throughput is defined as the rate of successfully transmitted packets that the communication channel can transfer through the network from a Base Station BS to a Subscriber Station SS within a period of time (Anouari and Haqiq, 2012) (Sadri and Khanmohamadi, 2013). It is measured in terms of packets per second or per time slot and can be used to estimate the efficiency of a network (Anouari and Haqiq, 2012). It can be expressed as an individual station throughput or system throughput. System throughput is the throughput of all subscribers that share a common communication medium. The mathematical formula used to calculate this metric is given by:

$$Throughput = \frac{(Total\_Byte\_Sent \times 8)}{(TimeLPR - TimeFPR)}$$

Where TimeLPR is the time when the last packet is received and TimeFPR is the time when the first packet is received as the TimeLPR and TimeFPR of the total duration time of simulation. As defined in WiMAX form, the value of throughput should be high otherwise it affects every service class in the network (Mehta and Gupta, 2012).

The throughput is the key metric that will be used to evaluate the performance of our proposed algorithms.

Delay

Delay is a performance measure of high-speed services such as voice, data, and video. The delay of a network specifies how long it takes for a successfully transmitted packet to travel across the network from source to destination (Anouari and Haqiq, 2012). It is typically measured in multiples or fractions of seconds. The

WIMAX forum recommends that the delay should be less than 150 *ms* (AL-Hawawreh and Zreikat, 2017). Delay may differ slightly from packet to packet depending on the location of the specific pair of communicating nodes and can be classified into processing, queuing, transmission, and propagation delay.

The mathematical formula used to calculate Delay is:

$$\begin{aligned}\text{Delay} = {} & \text{Propagation time} + \text{transmission time} + \\ & \text{queuing time} + \text{processing delay}\end{aligned}$$

Jitter

Jitter is the variation in delay between different packets. For packets arriving at a destination as a series of frames, the space-time between arrival frames is referred to as Jitter. Jitter is measured to evaluate which traffic is capable of tolerating a certain amount of delay, such as real-time traffic.

Measuring Jitter is a critical element to determining the performance of a network in terms of stability and consistency. No Jitter equates to a network with a constant latency (Anouari and Haqiq, 2012) . The value of Jitter is calculated from the end to end delay using a bit error ratio (BER), as well as combined random and deterministic Jitter.

The mathematical formula used to calculate total jitter is:

$$Jitter_{(Pkt)} = TD_{Current(pck)} - TD_{Previous(pkt)}$$

where $TD_{\text{Current(pck)}}$ transmission delay of the current packet and $TD_{\text{Previous(pkt)}}$ transmission delay of the previous packet.

Packets

The traffic packets are the main recognizable elements in computing Throughput, Delay and Jitter. Packets are partitioned into three events: packet arrival events, packet departure events, and packet reception events. A packet arrival event interacts with the BS, while packet departure events interact with the SS. The first event to occur is always a packet arrival, since the simulation starts in the inactive state with all queues empty. In any case, the arrival event is triggered if the packet arrival time is smaller than the packet departure time. A packet arrival event represents packets generated from the application server destined for downlink flows. Each packet is mapped to its respective traffic class.

Table 3.1 lists the parameters used in configuring the PHY and MAC layers for simulation analysis and shows the default configuration of the NS2 WiMAX simulation. The variables were selected to achieve a clearer understanding of the performance of the algorithm under a variety of conditions. By varying speed, frame and the number of users the resulting differentiated outcomes/results demonstrate how each algorithm performs.

The table shows a number of simulation parameters for three different case studies covering a number of variables. There is consistency across the variables with the exception of Frame Duration, Mobile Nodes and Duplexing Mode which give differentiated results. The case studies allow the different performances of each algorithm to be checked in different conditions and provide a comparative measure of each algorithm's performance.

Table 3.1: Simulation Parameters and Values

Simulation parameters	Case study 1	Case study 2	Case study 3
Channel bandwidth	10 MHz	10 MHz	10 MHz
Carrier frequency	2.6 GHz – 3.5 GHz	2.6 GHz – 3.5 GHz	2.6 GHz – 3.5 GHz
OFDMA frame size	5ms	5ms	5ms
Number of slots in each frame	48	48	48

Subcarriers in each slot	18	18	18
Subcarrier frequency spacing	10.94 kHz	10.94 kHz	10.94 kHz
Data Subcarrier	420	420	420
Pilot Subcarriers	56	56	56
FFT size	512	512	512
DL/UL frame ratio	28/25	28/25	28/25
Frame Duration	**10 ms**	**20 ms**	**25 ms**
Transmission Power for BS	20 dBm	20 dBm	20 dBm
Transmission Power for SS	15 dBm	15 dBm	15 dBm
Transmission Power for MS	10 dBm	10 dBm	10 dBm
Channel Bandwidth for DL	10 dBm	10 dBm	10 dBm
Channel Bandwidth for UL	10 dBm	10 dBm	10 dBm
Modulation	QPSK,16QAM,64-QAM	QPSK,16QAM,64-QAM	QPSK,16QAM,64-QAM
Mobile Nodes	**5 – 140 Users**	**5 – 140 Users**	**5 – 140 Users**
Mobile user velocity	60 km/h	60 km/h	60 km/h
Duplexing Mode	**5 Mb 4 ms**	**20 Mb 10 ms**	**30 Mb 15 ms**

3.6 Portfolio Decision Event

The most complex part in the execution of the Portfolio algorithms in real-time is the correct scheduling of the different algorithms because switching between algorithms operates based on chronologically consecutive events on a system's state. The best results are achieved by mixing the composition of algorithms in a timely fashion.
An inaccurate swap of the Portfolio events would impede the ability of the simulator to facilitate the arrival of its packets and thus impact QoS and fairness. Performance can be affected in WiMAX networks by using the different applications, the number of mobile nodes and channel conditions etc.

If the BS is found to be idle, it is set to be busy, and a packet departure time is scheduled. If the BS is found to be busy, the arriving packet will be sent to its queue. Packet loss will occur if that queue is full. Otherwise, the respective queue counter is incremented. The packet departure event is at the heart of the data encryption standard (DES), which schedules packets from the queues corresponding to various flows. It is activated by the event scheduler when the packet departure time is smaller than the packet arrival time. The time advancing mechanism is updated to the packet departure time, and the BS is flagged. All packets arriving at the BS will be served and leave the system vicinity. The occurrence of a packet departure is achieved by selecting that queue and the packet to be serviced next. The departure of a packet from the BS is done by selecting one of the available queues that are non-empty and transmitting a packet from it. Any queue from which a packet is transmitted has its queue counter decremented. The number of packets transmitted is incremented, and the next departure time is scheduled in that queue.

If the queues are empty, the departure time is set to a very large value for each queue, so that the departure events for each queue will be delayed until there is an arrival event in that queue. The packet delay is computed at the same time as the packet is transmitted. The packet received event happens when all the departed packets have successfully arrived at the SS. These packets will continue to be accumulated until the end of the simulation.

This is the primary component for any discrete event simulation. It manages the list of events to be processed, based on their respective times of arrival with best QoS and highest throughput performance. In the proposed Portfolio, packet arrival has been considered to happen first, before any other events. To enforce the sequencing of these events by different scheduling, the packet arrival event has been flagged first with an initial time = 0.

Subsequent packet arrival events are subject to the chronological development of the events. The time advancing mechanism does not advance in discrete steps but advances with the occurrence of each event.

These procedures are repeated by switching to the proper scheduling algorithm until the termination of the simulation is reached in highest throughput performance. The performance metrics are used to compute the statistical results. The swap technique is the main tool for the performance of the proposed Portfolio algorithms in WiMAX networks. The focus of the simulator is to implement the DL packet scheduling and CAC as well as related algorithms at the MAC layer.

3.6.1 Definitions and Conventions

This section provides definitions and conventions used for the frequent terms used in the study clarity.

The term “network” refers to the broadband wireless network.

The term “traffic” refers to a unidirectional flow of packets specified with QoS parameters.

The term Subscriber Station (SS), which in other research is used interchangeably with Mobile Stations, is used to represent Subscriber Station.

3.6.2 Classifier

In WiMAX, the packet classifier classifies incoming packets from the upper layer that are received by the MAC layer at the BS. These packets, destined for the SSs, are associated with a SFID and mapped to a MAC connection using a Connection ID (CID). These packets are then put into their per-class queues and continue to wait at the BS for transmission over the link toward a given SS.

3.6.3 MAC Queue

This sub-component represents the queues for the various scheduling services that cannot be sent immediately. Each scheduling service, such as UGS, ertPS, rtPS, nrtPS, and BE, has a queue associated with it and has several packets waiting to be transmitted via the link. These packets have a maximum time limit for waiting in the queue, depending on the scheduling service.

To simulate the MAC Queue, a random probability is used to describe the maximum waiting time. To simulate the statistical properties of this maximum waiting time, random numbers are assigned to it. Each queue has a fixed capacity of packets, which is represented using an array data structure. If the number of packets in the queue is less than its capacity, an incoming packet is added to the queue. If the number of packets in the queue is greater than its capacity, the incoming packet is dropped. The queues associated with different scheduling services are controlled by the DL scheduler on a frame basis.

3.6.4 Downlink Scheduler

This subcomponent schedules the transmission of the packets waiting in the queues which are to be sent in the next DL frame. It regulates the transmission of packets, based on the QoS requirements of each queue and the queue status of the different DL connections. The Packet Scheduler selects which queue to service first and how many packets to be transmitted, by having access to a shared transmission resource. The selected packets are transmitted on a frame basis.

3.6.5 Call Admission Control

This sub-component is responsible for accepting requests from the MS application. There are two requests: new connections and hand-off connections. This sub-component reserves bandwidth for the handoff connections and the leftover bandwidth is used to admit new connections. To accept handoff connections, CAC checks whether the available bandwidth will be enough to admit a handoff connection without degrading the QoS of the admitted connections. If not, it rejects the handoff connection.

3.6.6 Initialization

The initialization sub-component triggers the operation of the entire simulation by initializing all parameters: the control parameters, the performance metrics, etc. These parameters are either initialized to fixed or variable values. They are also responsible for initializing all the various traffic sources to start in an inactive state. The duration of inactivity is a random statistical distribution for each traffic and is determined based on the statistical characteristics of the traffic source. If the inactive times are determined, the time for the first arrival of each traffic source can be established. The duration of the activity can also be found based on the various statistical distributions. It is also assumed that the system state starts at idle, which is equivalent to a starting time of zero.

The traffic sources initialized are forwarded to their respective queues, and each queue is mapped to a traffic type. After the initialization step, the first packet arrival times and the active period for each traffic source are known. The event scheduler then checks all the packet arrival times and selects the smallest packet arrival time among all the traffic sources. The smallest packet arrival time is chosen as the overall first packet arrival time. Thus, the event scheduler is responsible for selecting the next event to be called.

3.7 Summary

This chapter described the researcher's proposed portfolio algorithms. Then, the framework of the proposed schemes was presented together with an illustration of the starting point at pre-analysis of the existing schemes through to the evaluations and results.

The stages of the thesis were depicted in a framework that showed the different models and their integration and simulation experiments. The system models used in this study are outlined in the second part of this chapter. To reiterate, these are the Channel capacity model, Queuing model, Traffic model and Network model.

The final part of this chapter provided details of the performance metrics and experimental environments. The modelling of the portfolio decision event and scheduler event are presented.

CHAPTER 4

RESULTS OF A PORTFOLIO FIXED ALGORITHMS

4.1 Introduction

This chapter is concerned with performance analysis of the existing WiMAX scheduling algorithms against the Portfolio algorithms proposed in this thesis. It discusses the results obtained from experimental setups and the simulations.

4.2 Analytical Model

In this research, the BS is represented in the form of an analytical model. In the model, a single BS and several SSs and MSs are considered, as shown in Figure 4.1. Some of the MSs are in the cell that requests new connections, while others are around the cell that requests handoff connections.

In this situation scheduling algorithms are needed as well as single algorithms with different performance behaviours. The MSs request bandwidth from the BS whenever there is a new or a handoff connection. All the WiMAX service classes are used at the BS, such as UGS, rtPS, ertPS, nrtPS, and BE. Each of the service classes requires QoS guarantees and request either a new admission or a handoff connection.

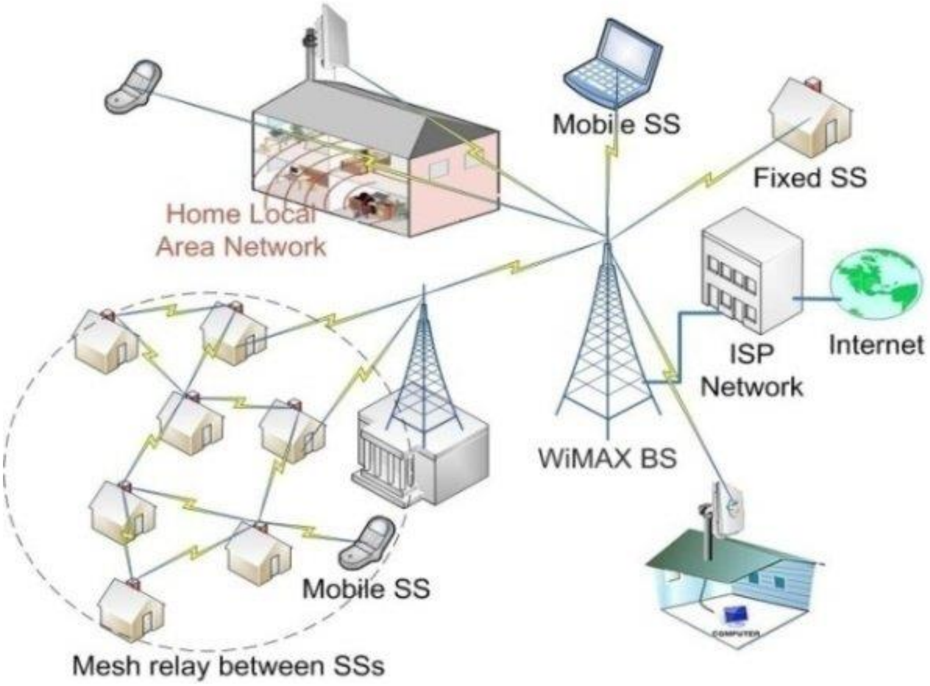


Figure 4.1: WiMAX PMP Topology Including Mesh Relay Network

The Markov process is used to model the proposed QoS-Aware CAC With BR and BD analytically for each class i of traffic for i = 1, 2, 3...k. In each class i, the model assumes that new and handoff connections arrival rate follow a Poisson process, while their mean service rates follow exponential distributions.

Based on the above, the BS transits its state when there is an admission or a rejection or a degradation of the service class. In addition, it is assumed that the BS admits or rejects or degrades a single connection at any given time. The future state of the BS depends only on the current state of the BS, and does not depend on its past states. Therefore, it is concluded that the states of the BS form a Markov chain, and hence the BS can be analytically modeled using a Markov model. Moreover, since the Markov chain obtained is ergodic, it is inferred that it is irreducible and aperiodic.

The CAC algorithms play an important role during the establishment of new connections to control the packet entry, reserve resources for guaranteeing QoS and enhance system throughput as previously mentioned in chapter 2.

The research also considers a CAC scheme in the BS that reserves bandwidth for handoff connections and admits a new connection whenever there is sufficient. bandwidth left or degrades the service of the ones already admitted at their minimum. Otherwise, it rejects either of the connections.

A Portfolio process is used analytically for each traffic to model the proper algorithm based on experimental scenarios as shown in section 4.5.
The algorithms are evaluated in terms of the new connection blocking rate and the handoff connection dropping rate, and the throughput by means of simulations.
When the network arrival rate is low, all the algorithms have similar throughput because all of them use equal bandwidth requirements. But as the network arrival rate increases, the proposed algorithm performs better than the other algorithm as well as the proposed algorithm increases the throughput. The increase in the throughput is attributed to the new admission criteria introduced.

The research provides three case studies and each case with a large quantity of scenarios to examine the performance of the scheduling algorithms.

4.3 System Description and Parameters

The proposed Portfolio algorithms in this chapter will be a focus on the fixed series of algorithms, with assumed scenarios, have been implemented in NS2 simulator using the WiMAX module (Fall and Varadhan, 2005). The NS2 simulator has been used to simulate and evaluate the algorithms’ behaviours. The performance metrics used to measure of the proposed algorithm, are: Throughput, Delay and Jitter. All tested scenarios have been simulated with SSs included between the BS and MSs.

4.3.1 The Simulation Setup and Parameters

The simulation setup corresponded to the connection of the BS to another SSs as illustrated in Figure 4.2. The MSs are associated and that they communicate with the SSs which is linked with a base station (BS).

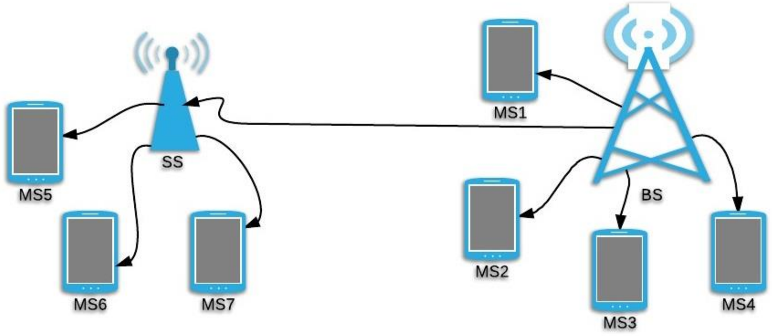


Figure 4.2: Simulation Scenario with 140 MSs

The developed 802.16-based WiMAX module named as the Mac802.16 class is in accordance with the specifications of the IEEE 802.16-2004 standard and based on the ns-2 version 2.35. All modules are designed by using object-oriented programming language C++ and modelled as several classes. The relationship between the WiMAX module and legacy ns-2 modules is based on the original network component stack of the ns-2.

First, the traffic generating agent TGA is considered simply as an application level traffic generator that generates VoIP, MPEG, FTP, HTTP traffic, and so on. This traffic is classified into five different types of service; the UGS, rtPS, ertPS, nrtPS, and BE, each with its own priority. All packets will be transferred to different types of priority queues according to their service types by using CS layer SFID-CID mapping mechanism. The data packets in these queues are treated as MSDUs and will be selected to pass into the WiMAX module in a round robin manner.

While the WiMAX module in the SS receives the MSDUs from the Queue object, the MAC management component will initiate the ranging process to enter the WiMAX system or to transmit the MSDUs according to the scheduled time obtained from UL-MAP. Once the process has been successfully finished in the MAC layer, the Network Interface will add a propagation delay and broadcast in the air interface.

In each simulation run, varying numbers of mobile stations were used. In order to simulate the IEEE 802.16, PHY and MAC modules were implemented in NS2 with the 802.16 functionality.

The algorithms have been tested in a total of 405 scenarios. All scenarios used the 5 conventional algorithms and 4 proposed fixed Portfolios. Within the simulations, the number of MSs varied, ranging from 5 to 140 MSs.

Each algorithm was tested in one scenario with 5 MSs and one with 10 MSs.

A number of simulations with varying numbers of mobile stations (MSs) in each simulation run were carried out by the researcher (5, 10, 20, 30, to 140 MSs) representing 15 scenarios that have been simulated as follows:

- Each algorithm was tested in one scenario with 5 MSs and one with 10 MSs until the last one with 140MSs.
- 15 scenarios equating to measured intervals of 5/10/20 – up to 140 MSs distributed over all three case studies in total of 405 scenarios.
- All scenarios used the 5 conventional algorithms and 4 proposed fixed Portfolios.

When the number of users at the BS is small, comparing the performance of algorithms is less accurate since it leads to a smaller effect size.

Conversely, when the number of MSs is large, the comparison in the performance metrics will be clear due to the fact that the values for these metrics will be generally high and differ widely between algorithms because high bandwidth requests should be served by BS/SSs (Chan et al., 2008, Huang et al., 2009, Ghosh et al., 2009). In each scenario, the MSs were located randomly under base station coverage. The Constant Bit Rate is mainly used in networking applications as content can be transferred through limited channel capacity. Additionally, the Constant Bit Rate was used as the application between the source and destination nodes. The Constant Bit Rate in the NS2 simulator has five different precedence levels to represent the five QoS classes (Fall and Varadhan, 2005) as summarised in Table 4.1.

Table 4.1: The QoS Services and their Precedence

Service Type	Precedence
UGS	0 (highest)
ertPS	2
rtPS	3
nrtPS	4
BE	7 (lowest)
Source (Fall and Varadhan, 2011)	

The rate of Throughput, Delay and Jitter of the proposed Portfolio algorithms were compared with Droptail, DRR, FQ, RED, and SFQ algorithms.

In this section, the most important features and components used in the simulations of case studies are discussed. In all case studies, the scenarios can be implemented in general scheduling algorithms, which are basically applicable to any OFDMA network. More information about the WiMAX model in NS2 can be found in Appendix.

4.3.2 Network Topologies

The scenarios used the topology of the packet scheduling algorithm described in Section 3.4 that consists of a BS, SSs and MSs. The traffic source is an application that provides four different traffics was used namely: VoIP, Video streaming, FTP, and HTTP, each of which is associated with one distinct traffic type.

4.4 Performance Evaluation of Portfolios Experiments

Evaluation of the performance of the proposed Portfolio algorithms experiments can be done in three case studies modes in the network scenarios. Each case study consists of (15) scenarios and each scenario consists of different numbers of mobile nodes MSs. (Each case study has different configurations as mentioned in table 3.1. All the experiment in each case study has the same topology however the number of mobile node are increase from 5 to be 140 for the last experiment, to get a variety of performance results.

The mobile nodes move about within an area whose boundary is defined as shown in Table 4.1 of the Simulation Parameters. A TCP connection is set up between the mobile nodes. Packets are exchanged between the nodes as they come within hearing range of one another. As they move away, packets start getting dropped.

A mobile node consists of network components like Link Layer (LL), Interface Queue (IfQ), MAC layer, the wireless channel nodes transmit and receive signals. At the beginning of a wireless simulation, the type for each of these network components needs to be defined. Additionally, other parameters such as the type of antenna, the radio-propagation model and the type of routing protocol used by mobile nodes. Further details about these network components can be found in Appendix.

An analytical model and simulation experiments have been conducted in this research work to investigate the performance of the conventional algorithms against Portfolio performance. The algorithms are evaluated in terms of the average throughput, average delay, and average Jitter. These metrics are defined in Section 3.5.

The scheduling service is provided for both downlink and uplink traffic. The MAC supports frequency-time resource allocation on a per-frame basis. The resource allocation is delivered in MAP messages at the beginning of each frame. The resource allocation can be changed from frame to frame in response to traffic and channel conditions.

4.5 Results and Discussions

This section discusses the results obtained from the simulations in further detail. The main objective is to evaluate the effectiveness of algorithms against well-known algorithms such as Droptail, Deficit Round Robin (DRR), Fair Queueing (FQ), Random Early Drop (RED) and Stochastic Fair Queueing (SFQ).

In the following sub-sections, the performance investigation of the conventional algorithms and the proposed Portfolio algorithms are presented.

4.5.1 Throughput Results

Based on the performance metrics that have been employed in this research work, the proposed Portfolio algorithms are effective for the WiMAX network in terms of throughput in all case studies. For such networks, various common deployment topologies are viable alternatives. The performance of the proposed Portfolio algorithms is analyzed using two methods. The first is a fixed series of algorithms and the second is a Portfolio Peak as mentioned earlier in (3.1).

In order to simulate the WiMAX operation, this research implemented MAC and PHY modules in NS2 with the 802.16 functionality. The Droptail, Deficit Round Robin (DRR), Fair Queueing (FQ), Random Early Drop (RED), Stochastic Fair Queueing (SFQ) have been implemented with all scenarios that specified the standard with which the proposed Portfolio algorithms should be compared.

As more MS(s), sequentially, between BS and SS(s), means higher delay and higher throughputs, simulated scenarios with varying numbers of mobile stations and varying numbers of SSs (up to three SSs) were set up by the researcher in order to evaluate the proposed Portfolio algorithms.
Next, the simulation results on all performance are presented, as well as intensive simulation work carried out to confirm that Portfolio algorithms have performed accurately in all the Case Studies.

In general, when the number of users associated with BS/SS is small (e.g., one or two MSs), the evaluation of the performance metrics (e.g., throughput, delay, or Jitter) for two different algorithms will be inaccurate as the values for these metrics will be very low in one case and close in one algorithm to those in the other case, and vice versa.

Case study 1 **Case study 2** **Case study 3**

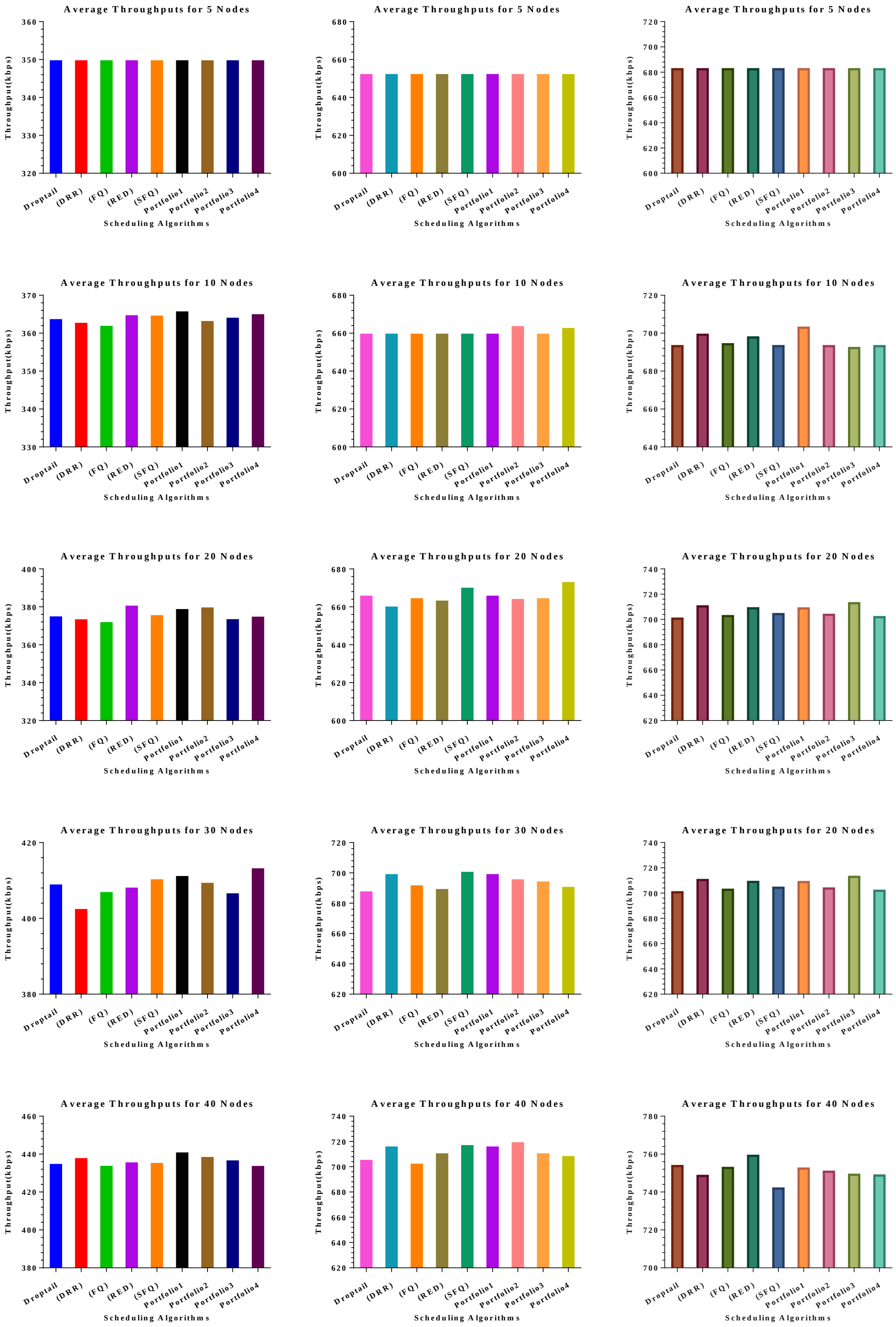


Figure 4.3: Average Throughput Performance for (5-40) Nodes

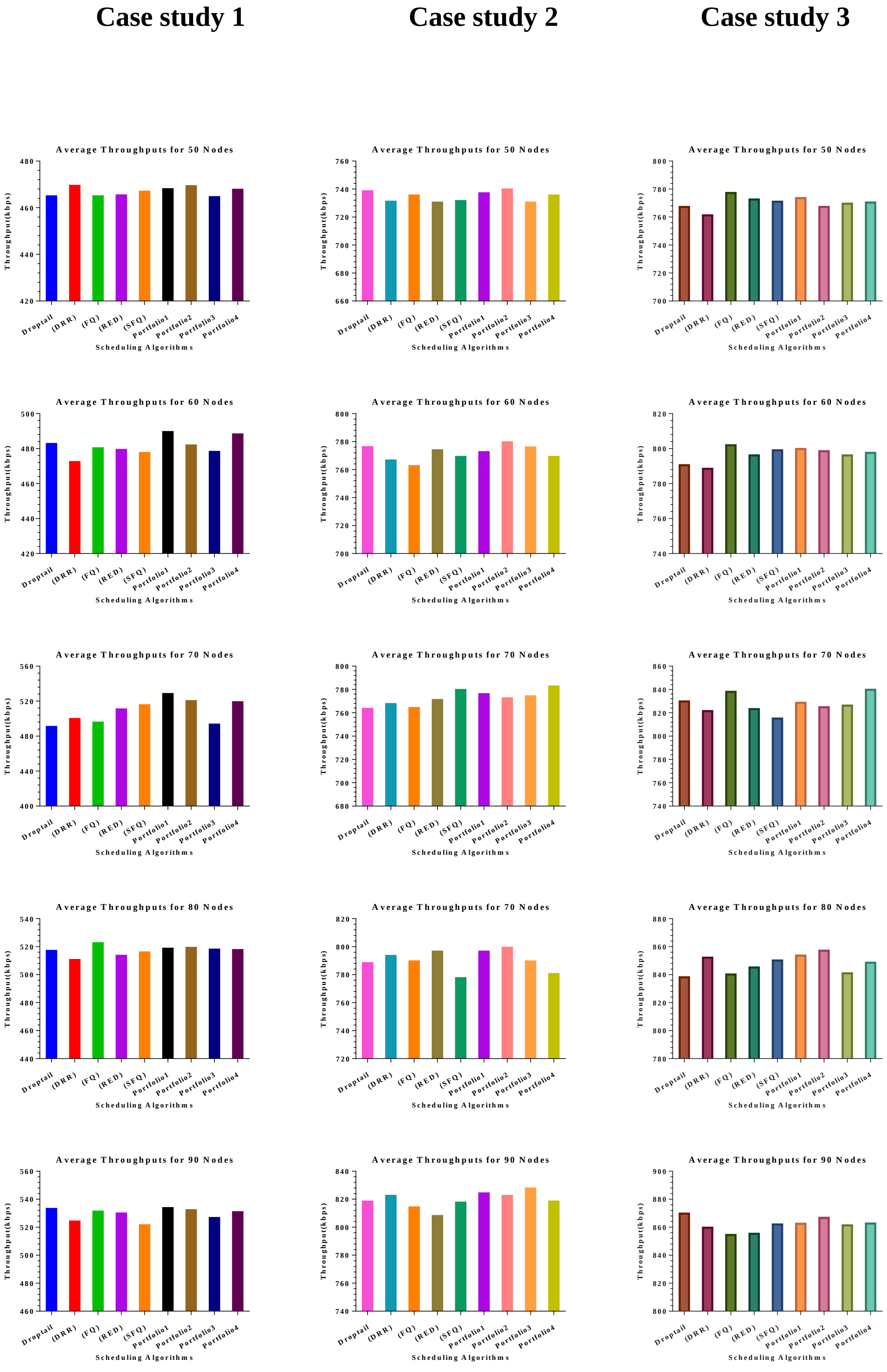


Figure 4.4: Average Throughput Performance for (50-90) Nodes

Case study 1 **Case study 2** **Case study 3**

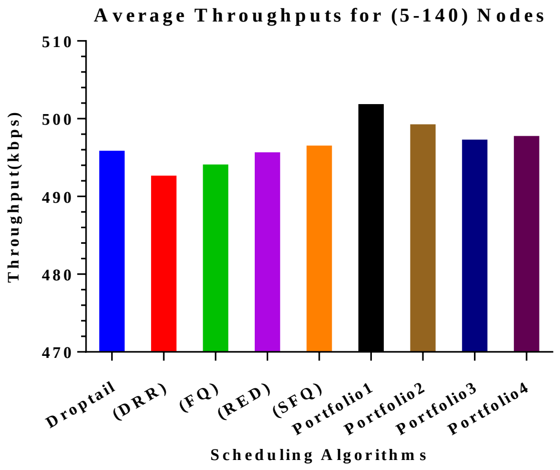
Figure 4.5: Average Throughput Performance for (100-140) Nodes

The proposed algorithms perform more efficiently against traditional algorithms within most of the scenarios no matter whether the number of MSs becomes larger or smaller. That Portfolio1 achieves the highest capacity among all these schemes.

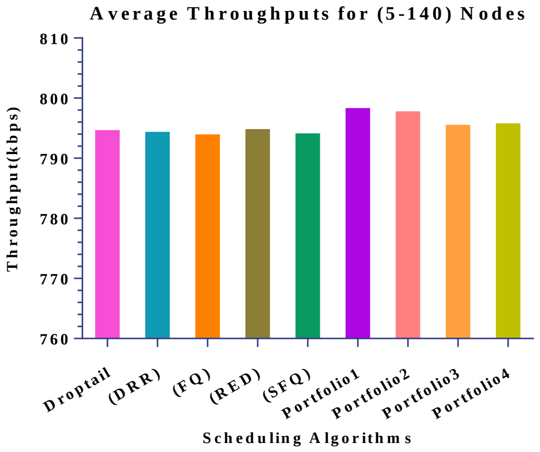
The proposed Portfolio algorithms produce higher throughput than the conventional algorithms as the number of MSs increases or decreases as illustrated in Figure 4.6 of the total average of 5 to 140 MSs.

The simulation results of the system throughput under the experimental scenarios demonstrate that the throughput increases when the number of MSs increases.

Case study 1



Case study 2



Case study 3

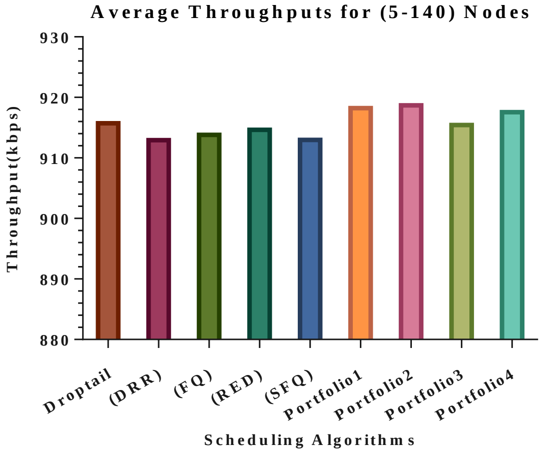


Figure 4.6: Average Throughput (5-140) Nodes for all Case Studies

4.5.2 Delay Results

The average delay for all traffic classes increased with the number of MSs in the network in every case study as shown in Figures (4.7 to 4.9). This is to be expected in each scenario environment due to queuing delays at the Mobile node(s). Furthermore, the increment in delay is also affected by different factors such as channel conditions, sizes or numbers of the bandwidth requests, as well as the number of MSs and SSs. When the conventional algorithms have been used or the proposed Portfolio algorithms have been utilised, its effects on average delay for all traffic classes appears to be minimal. However, the average delay for the different numbers of MSs with the conventional algorithms was higher than with the proposed Portfolio algorithms of fixed series under most scenarios.

The simulation results show that the average delay with the conventional algorithms under the scenarios is higher than that of the proposed Portfolio algorithms. Figure 4.10 displays the simulation result for the average delay with scenarios of different numbers of MSs. The simulation results confirm that the average delay with the proposed Portfolio is lower than that obtained by conventional algorithms. In addition, the delay is increased substantially when the number of MSs increases.

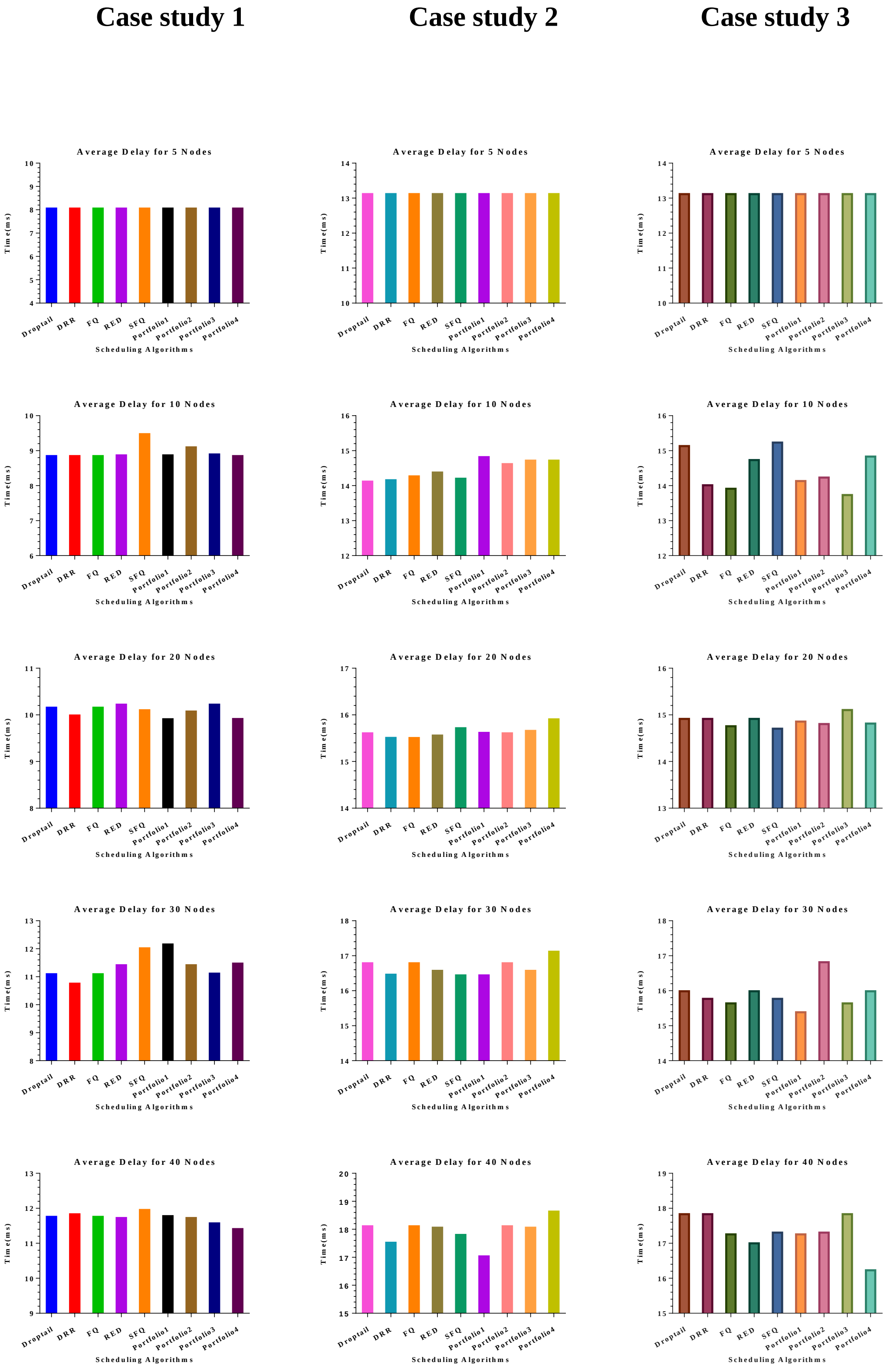


Figure 4.7: Average Delay Performance for (5-40) Nodes

Case study 1 **Case study 2** **Case study 3**

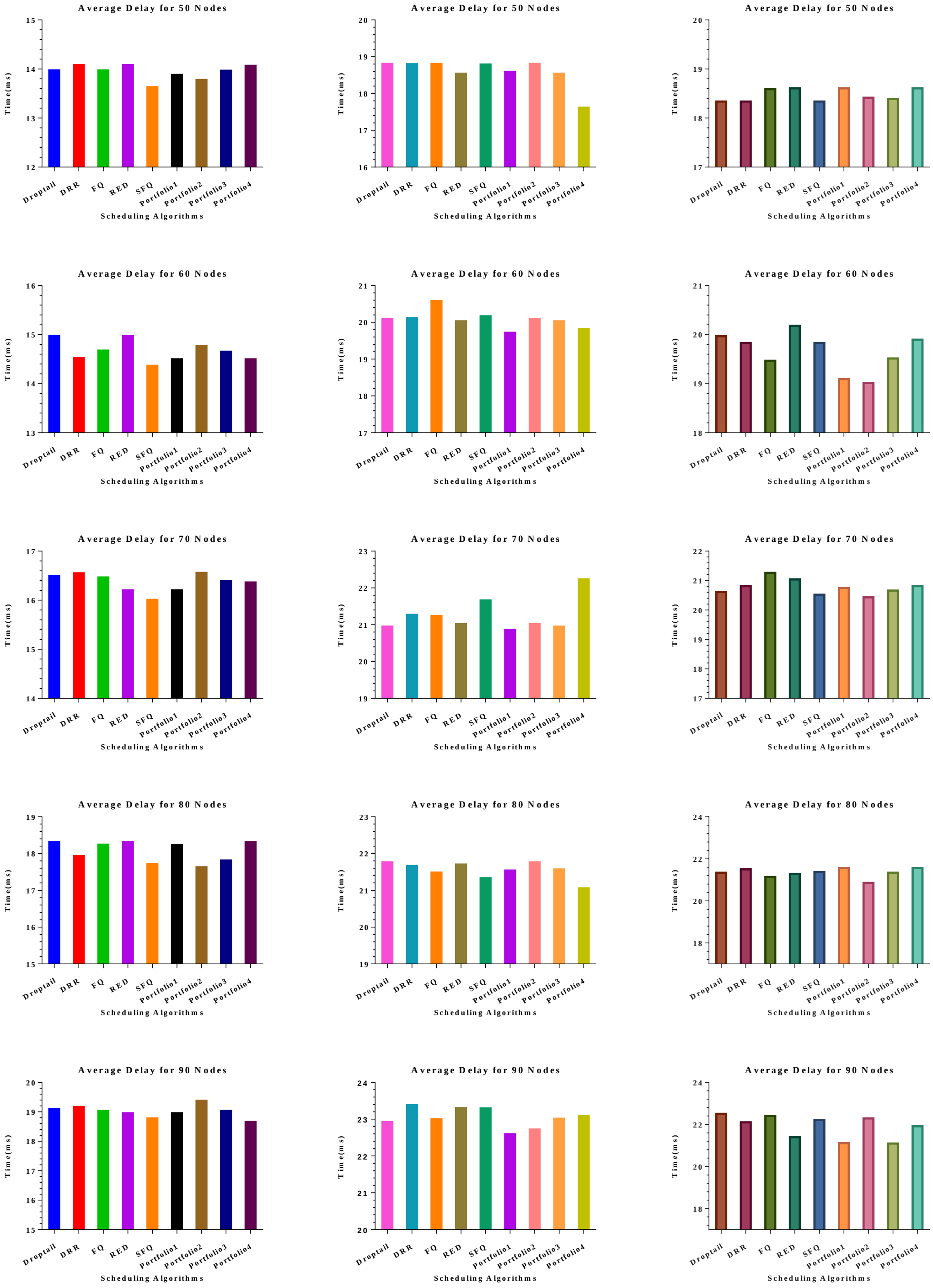


Figure 4.8: Average Delay Performance for (50-90) Nodes

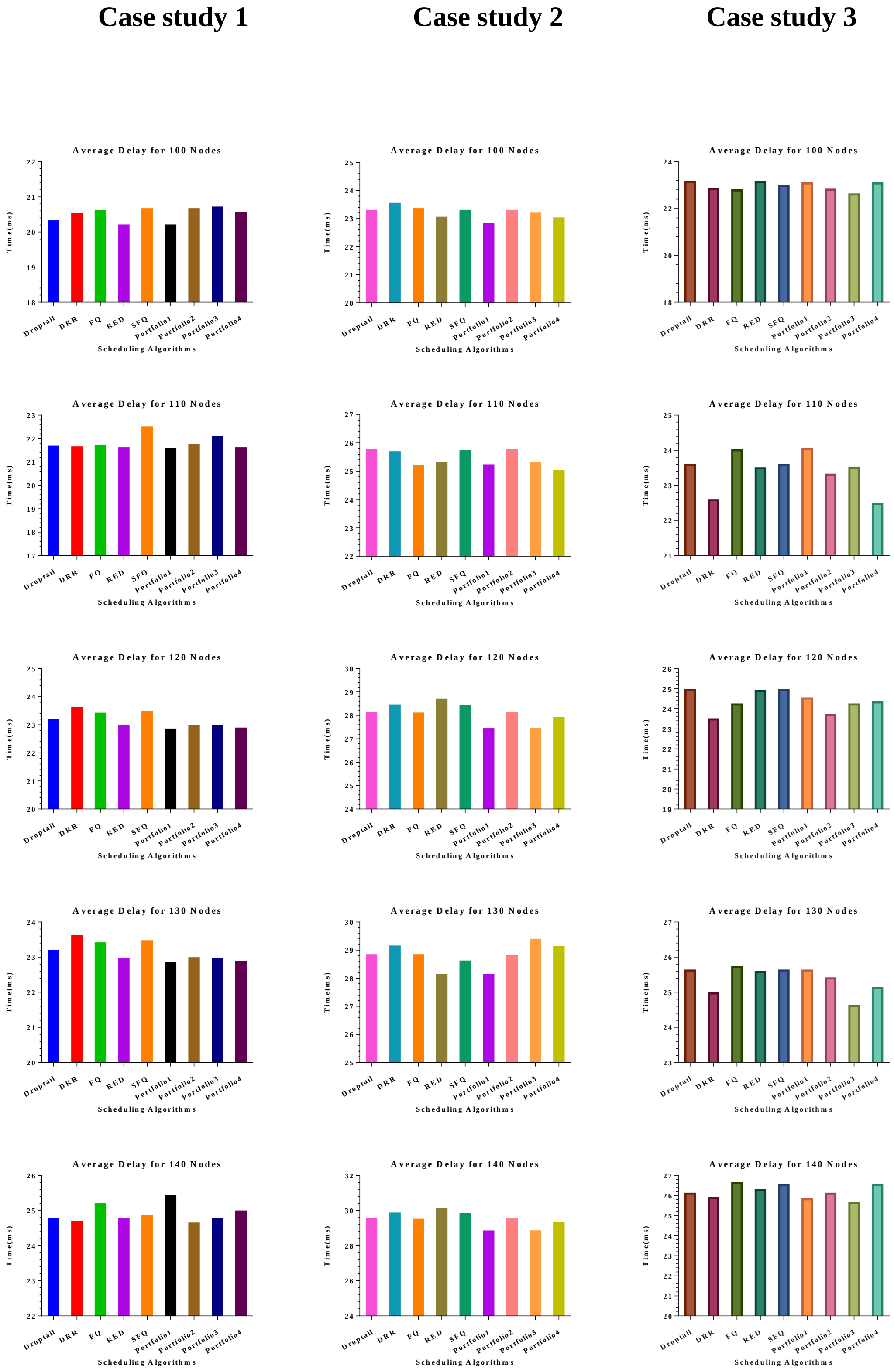
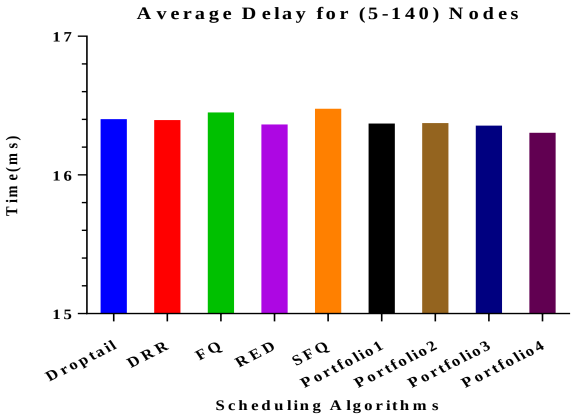
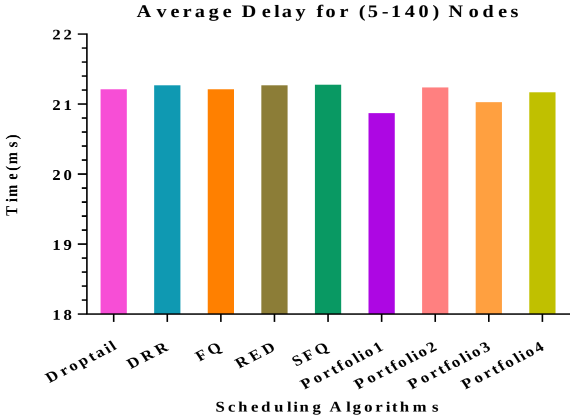


Figure 4.9: Average Delay Performance for (100-140) Nodes

Case study 1



Case study 2



Case study 3

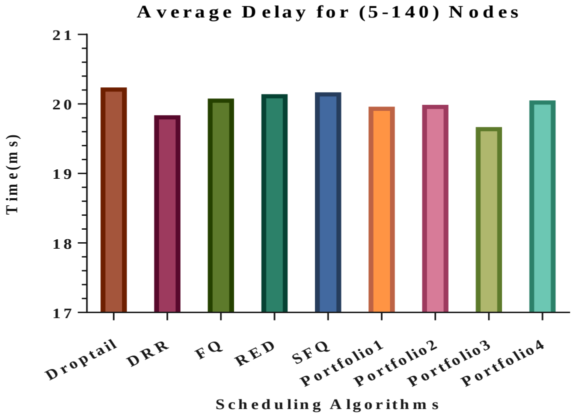


Figure 4.10: Average Delay (5-140) Nodes for all Case Studies

4.5.3 Jitter Results

The proposed Portfolio algorithms achieve less Jitter relative to the conventional algorithms operating with the mechanism currently specified in the standard and based on the number of MSs used. In addition, the gap between the proposed Portfolio algorithms and the conventional algorithms becomes larger with the number of MSs as shown in figures (4.11 to 4.13).

Figures 4.14 illustrates the Jitter performance of the scenarios of varied numbers of MSs. The figure shows how the proposed algorithms achieve the optimum independent of the number of MSs. The results also suggest that the proposed algorithms are more stable than the conventional algorithms due to their lower rate of Jitter. It can also be observed that as the number of MSs increases, the Jitter increases slightly as a consequence of the increment in the network delay resulting from the high number of users leading to a delay in MAPs reception from the BS.

Figure 4.11: Average Jitter Performance for (5-40) Nodes

Case study 1 **Case study 2** **Case study 3**

Figure 4.12: Average Jitter Performance for (50-90) Nodes

Case study 1 **Case study 2** **Case study 3**

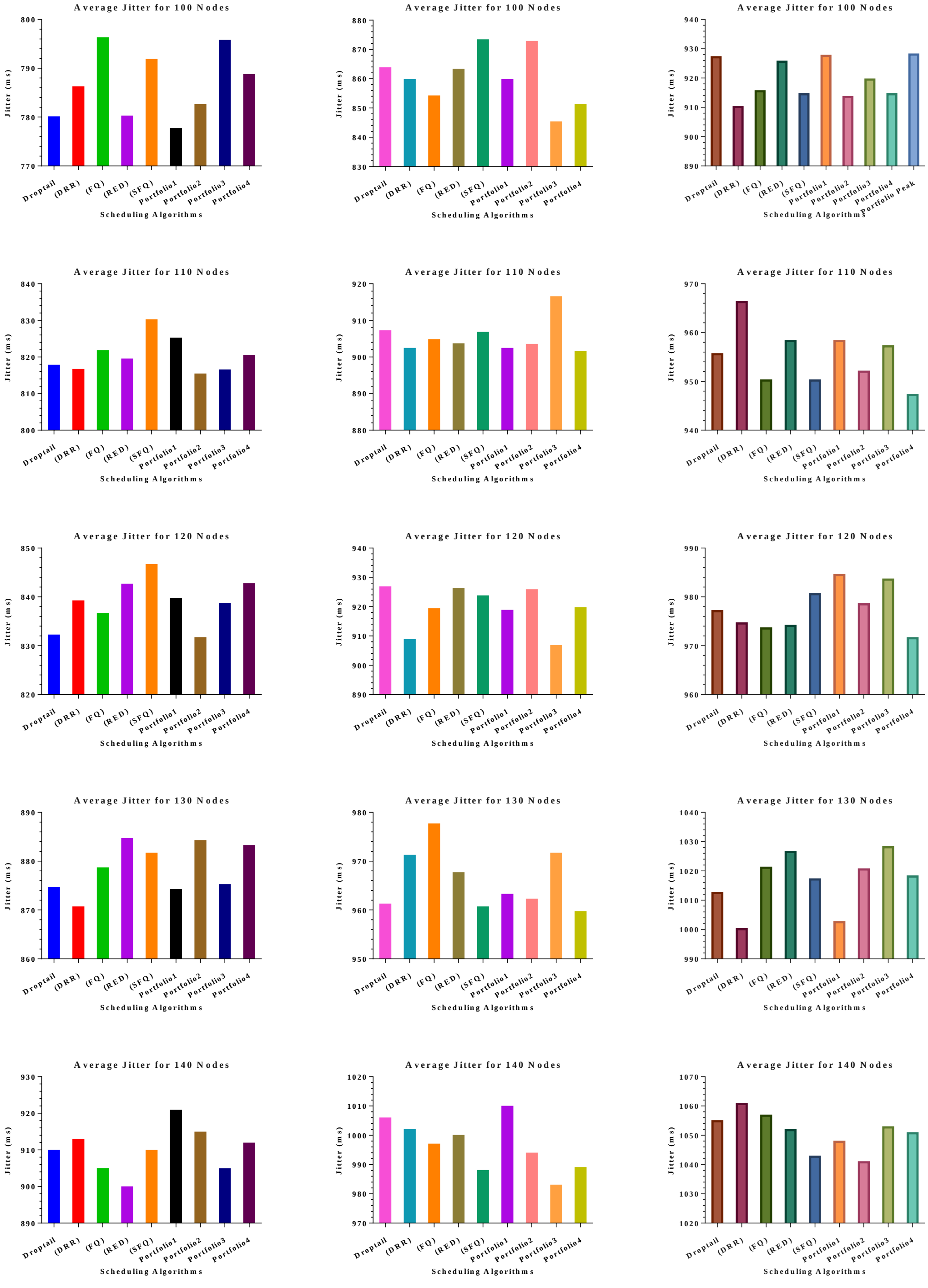
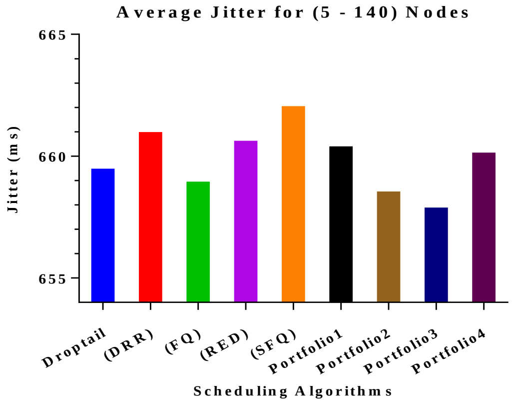
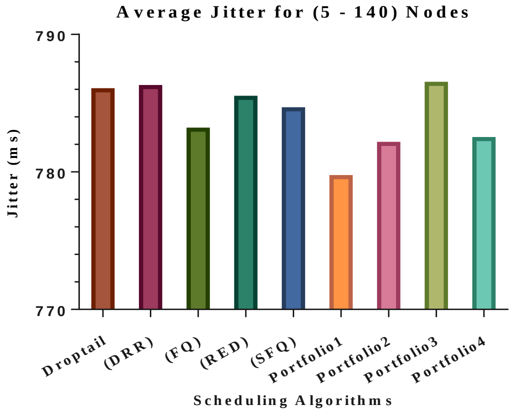


Figure 4.13: Average Jitter Performance for (100-140) Nodes

Case study 1



Case study 2



Case study 3

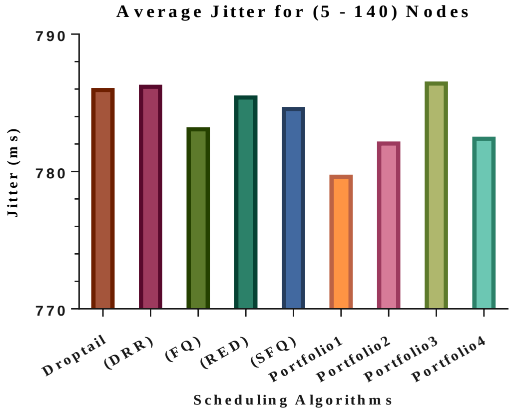


Figure 4.14: Average Jitter (5-140) Nodes for all Case Studies

4.6 Summary

This chapter was concerned with the results of the performance evaluation between the proposed Portfolio ~~scheduling~~ algorithms and the conventional algorithms for the WiMAX Networks.

The results indicate that the proposed fixed Portfolio of algorithms out-performs the existing algorithms in terms of network throughput in most cases, as well as in the total of the average number of 5 to 140 Nodes.

The proposed algorithms have demonstrated substantial enhancement in terms of system throughput, reduction of average delay, and reduction of Jitter of the IEEE 802.16 standard.

CHAPTER 5

RESULTS OF A PORTFOLIO PEAK ALGORITHMS

5.1 Introduction

In recent years, broadband wireless systems have been established as one of the fastest developing areas in the field of telecommunications. The IEEE 802.16 standard (Group, 2004), also known as WiMAX, addresses broadband access technology for wireless metropolitan area network (WMAN). Current trends and demands point to the delivery of multimedia services such as voice, video, high-definition TV (HDTV) or interactive games with guaranteed QoS, which can be provided by WiMAX at the MAX layer (Borin and da Fonseca, 2008). In order to support high quality multimedia services, high capacity data transmission is necessary. Most wireless systems operate in high frequency bands above 2 GHz (Sikora and Groza, 2005). As a result, the transmitted signal is highly attenuated in comparison with lower bands.

Multimedia applications must be able to simultaneously support different types of traffic with distinctive QoS requirements such as: bandwidth, throughput, delay, and jitter. End user devices are becoming increasingly powerful and users desire apps with increasingly complex functions, which is why QoS remains a crucial issue for the IEEE 802.16 network performance. One way in which the QoS demands of different service classes can be approached, is through efficient scheduling of the MAC IEEE 802.16 layer.

Scheduling algorithms are essential to providing the required level of network QoS (Wang et al., 2012, Nguyen et al., 2014, Chen et al., 2014). Several algorithms have been proposed to support QoS in wired and wireless networks (Manolis Katevenis et al., 1991, Shimonishi et al., 1997, Yoon et al., 1999, Sayenko et al., 2008, Wei Nie et al., 2011). Some are classical algorithms that were not specifically designed for IEEE 802.16 but have been applied practically. Among these, WRR, which differentiates services among various traffic classes, is the most commonly used algorithm because of its simplicity and low computational cost (Sayenko et al., 2008, Wei Nie et al., 2011).

This chapter will focus on throughput performance evaluation of the scheduling algorithms and expansion of the simulation of Portfolio algorithms.

Results show that the portfolio peak is configured to guarantee QoS in WiMAX throughput and outperforms in all scenarios for these three cases.

This chapter is structured as follows: Section 5.2 presents the Proposed Portfolio Peak Algorithm. Section 5.3 presents an approach to Portfolio peak, Section 5.4 the results and discussion, while Section 5.5 is a summary of this chapter.

5.2 Proposed Portfolio Peak Algorithms

Portfolio algorithms have clear advantages over conventional algorithms. How and why will be explored in this chapter.

The portfolio is comprised of a collection of algorithms. These can be selected, through careful scheduling of each individual algorithm, to best serve the requirements of each service class. Different algorithms can also be combined, through allocating their time slots simultaneously, to achieve the best results.

The selection of algorithm is determined according to the highest performance of the service classes in each case. From the portfolio, can select the single best algorithm or allocated time slots to combine different algorithms.

The ability to time algorithms means that algorithms can be scheduled while a solution for a problem is sought.

In deciding when to select the algorithm, the methods enable a decision to be made before the solving of the actual problem starts or while a solution to the problem is sought. How this selection is made depends on the circumstances. For example, this decision could focus on maximising throughput and minimising the average delay in terms of QoS. Another essential step concerns finding information to help the selection, such as feature selection and extraction, and the performance of the use of the selected algorithms in the past.

The performance of algorithms can also vary dramatically when dealing with different problems. In this case, consideration must be given to the performance of the algorithm over three different instances of a different number of Mobile Nodes. In order to bridge this gap, the possibility of combining algorithms in the context of the peak performance of algorithms in each traffic of UL and DL transmissions is comprehensively studied in detail.

Figure 5.1 illustrates the proposed Portfolio peak algorithms in its relation to conventional algorithms, the modes of fixed Portfolio algorithms with the combination of the number of SSs and MS, as well as the type of application.

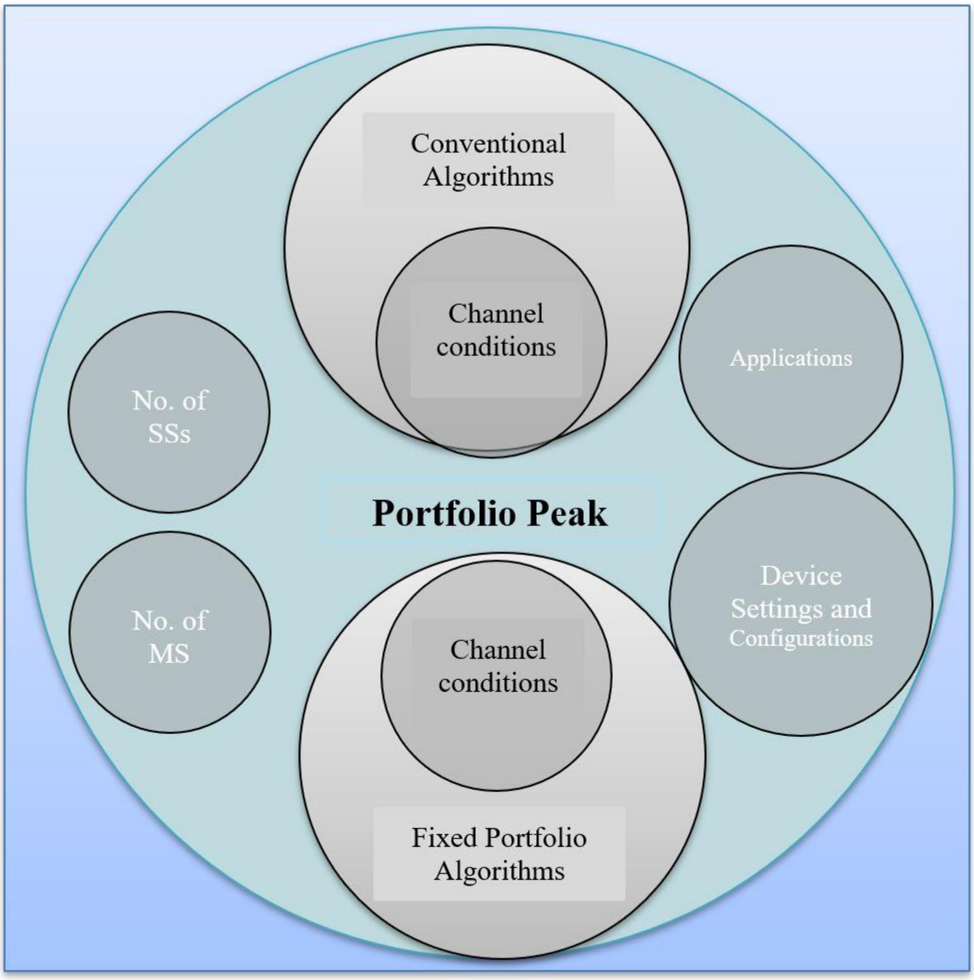


Figure 5.1: The concentration and a Schematic Diagram for the Proposed Portfolio Peak Algorithms

5.3 Principle and approach of Portfolio peak Algorithms

Providing quality of service (QoS) in wireless communication networks has become an important consideration for supporting a variety of applications. IEEE 802.16 based WiMAX is the most favourable technology for broadband wireless access with best QoS features for triple play (voice, video and data) service users. Unlike wired networks, QoS support is difficult in wireless networks due to the variable and unpredictable nature of wireless channels. In the transmission of voice and video, the main issue involves the allocation of available resources among the users to meet

QoS criteria such as delay, jitter and throughput requirements to maximize goodput and minimise average delay while keeping feasible algorithm flexibility and ensuring system scalability. WiMAX assures guaranteed QoS by including several mechanisms at the MAC layer such as admission control and scheduling. Packet scheduling is a process of resolving competition for bandwidth which determines the allocation of bandwidth among users and their transmission order. Various approaches for classification of scheduling algorithms in WiMAX have appeared in literature as homogeneous, hybrid and opportunistic scheduling algorithms.

A challenging research issue in high-speed WiMAX networking is how to control the transmission rate of data flows in order to guarantee the maximum throughput. In the scheduling distributed mode, the Base Station (BS) can determine the bandwidth allocation and scheduling for its subordinate stations, SSs/MSs, with or without information from the SS. In this case, a BS operates in distributed scheduling according to the peak performance and merging of different algorithms into a coordinated workflow that serves several traffics of different applications to ensure the highest QoS.

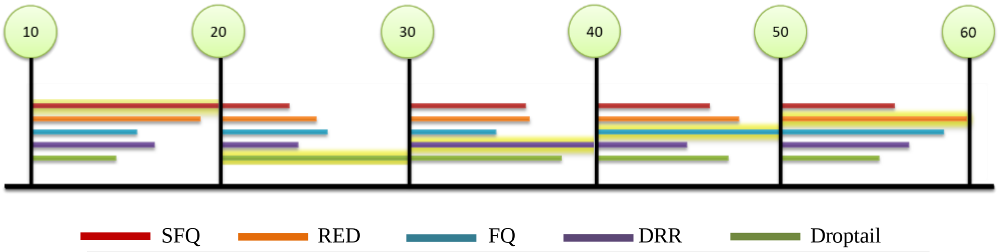
Over the years, there have been many approaches to solving the algorithm selection problem. Especially in artificial intelligence. Researchers have recognized that using algorithm selection techniques can provide significant performance improvements with relatively little effort.

Basically, a collection of existing algorithms is often referred to as components. Which is a combination of multi/different algorithms and/or diverse copies of the same algorithm running on different properties into a portfolio: is to improve on the performance of the component algorithms and meet all the QoS requirements.

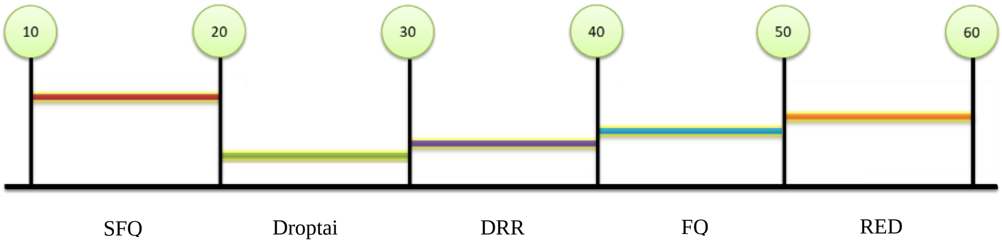
When portfolio peak algorithms are used in the WiMAX network, they will outperform single algorithms in all scenarios and under any conditions. Evidence from scenarios clearly supports the hypothesis.

Figure 5.2 gives examples of a network scenario over time with Five algorithms and explains the selection based on the best performance of the algorithms, the first graph shows the performance over time of Five algorithms. The second graph highlights the best performance of these algorithms. And the third graph is the final form of the Portfolio Peak Algorithm.

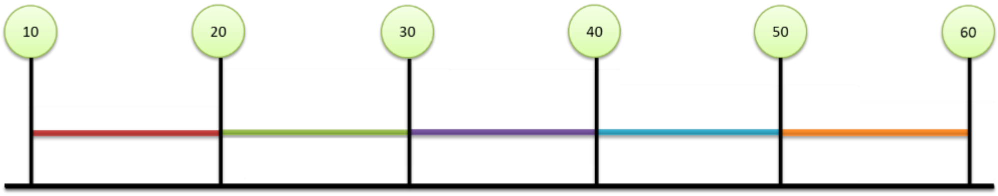
The selection of the algorithm is determined according to the highest performance of the service classes in each case. From the portfolio, we can select the single best algorithm or allocated time slots to combine different algorithms.



The performance of the five algorithms



Selected the best performance of the single algorithm



The portfolio Peak Algorithm

Figure 5.2: Implementation and Illustration for the Proposed Portfolio Peak Algorithms model

Each case study has different configurations as mentioned in table 3.1. All the experiments in each case study have the same topology. However, the number of mobile nodes is increased from 5 to 140 for the last experiment, to obtain a variety of performance results.

A Portfolio Peak Algorithms (PPA) is described as an algorithm for data traffic control in high-speed WiMAX networks by switching algorithms based on peak performance of each existing algorithm. In this way, the portfolio maintains the multiplexing flexibility of scheduling algorithms, switching while ensuring the highest throughput rate at the same time. The algorithms have been tested through simulations by evaluating their performance in order to support the QoS classes. The simulation has been carried out via the Network Simulator NS-2.35.

A Portfolio Peak Algorithms (PPA) is a modified method that provides the highest throughput performance in real-time and non-real-time applications according to the peak performance of the provided portfolio and Conventional Algorithms.
Moreover, PPA is suggested as the best choice for maintaining a higher throughput and minimising the mean average delay by providing a better result that shows a significant improvement in terms of throughput.

This chapter also examines the performance of a number of scheduling algorithms besides proposed Portfolio algorithms. The selection of the best algorithm is highly competitive. Through analysing these results, the chapter seeks to provide insight into the strengths and weaknesses of the algorithms.

5.4 Results and Discussions

Several heuristic approaches have been proposed to manage the scheduling problem in WiMAX Network. Their performances greatly depend upon the network topology and the bandwidth request of each node. Notably, they do not achieve optimal performance in all cases. This chapter has proposed the portfolio peak algorithms

(PPA) for minimizing the number of time slots required for a given set of bandwidth requests in WiMAX, as well as maximising the throughput performance to ensure the best QoS.

This chapter describes the results obtained from simulations using the NS2 simulator (Fall and Varadhan, 2005), which were used for comprehensive performance comparison for five well-known schedulers and the four Proposed *Fixed Portfolio Algorithms* against the *Portfolio Peak Algorithms.* The performance of the proposed Portfolio Peak Algorithms with the scheduling algorithms is compared in terms of the mean throughput, mean delay, and mean Jitter. These metrics are defined in Section 3.7.

This research study aims to analyse the Throughput, Delay and Jitter properties of an IEEE 802.16 network for evaluating the performance of various real-time and non-real- time applications.

The scenarios used the topology of the packet scheduling algorithm described in Section 3.4 that consists of a BS, SSs and MSs. The traffic source, an application that provides four different traffics was used, namely: VoIP, Video streaming, FTP, and HTTP, each of which is associated with one distinct traffic type.

The results and discussion on Portfolio Peak Scenarios are presented here for three case studies. In each case study, the throughput values for varying byte sizes, frame rates, and a number of nodes were measured.

The WiMAX throughput illustrated in figures 5.3 to 5.17 show the comparison between Proposed Portfolio Peak and conventional scheduling algorithms for three case studies, under different conditions for each case study. The highest performance recorded among all case studies and scenarios belonged to the Portfolio peak where there was a clear and significant difference.

The throughput is increased in the Portfolio Peak Algorithms (PPA) in all three case studies with no corresponding effect of increasing and/or reducing the number of nodes.

Figures (5.4 to 5.17) demonstrate that the throughput of Portfolio Peak is at its highest when measured against conventional algorithms for all scenarios in the three case studies. Neither is it affected by a/the number of nodes, SSs, or the particular type of application.

The proposed PPA outperforms all conventional algorithms in terms of network throughput.

However, when considering the average delay in figures 5.4(d,e,f) with 10 Nodes – 5.5(d,e,f) with 20 Nodes, there is no noticeable change in terms of getting best performance, and still within the mean.

Therefore, Figures such as 5.6(e) with 30 Nodes – 5.7(d,f) with 40 Nodes – 5.8(e) with 50 Nodes – 5.9(d,e) with 60 Nodes – 5.10(f) with 70 Nodes – 5.14(d) with 110 Nodes – 5.15(e,f) with 120 Nodes show that PPA achieved the lowest average of delay together with the best performance. Another interesting observation on the results of figures 5.11(d,e,f) with 80 Nodes, 5.12(d,e,f) with 90 Nodes, 5.13(d,e,f) with 100 Nodes, 5.16(d,e,f) with 130 Nodes - 5.17(d,e,f) with 140 Nodes is that PPA recorded the lowest mean of delay and, in all cases, consistently achieved best performance.

However, the simulation results of average Jitter illustrated in figures 5.4(g,h) with 10 Nodes - 5.5(i) with 20 Nodes - 5.6(g) with 30 Nodes - 5.7(i) with 40 Nodes - 5.8(h) with 50 Nodes - 5.13(g,h) with 100 Nodes - 5.16(g,i) with 130 Nodes, show that PPA had the lowest average of Jitter together with the best of performance. Another important observation on the results of figures 5.11(g,h,i) with 80 Nodes - 5.13(g,h,i) with 90 Nodes - 5.18(g,h,i) with 140 Nodes is that PPA achieved the lowest average of Jitter in all cases, along with best performance.

In general, when the number of users associated with BS/SS is small (e.g., one or two MSs), the evaluation of the performance metrics (e.g., throughput, delay, or Jitter) for two different algorithms will be unreliable as the values for these metrics will be very low on the one hand and close in one algorithm to those in the other, and vice versa.

The simulation study reveals that the proposed portfolio peak provides WiMAX networking with maximal throughput, the shortest transmission time and the proposed PPA achieves optimal performance results in all cases for guaranteed QoS.

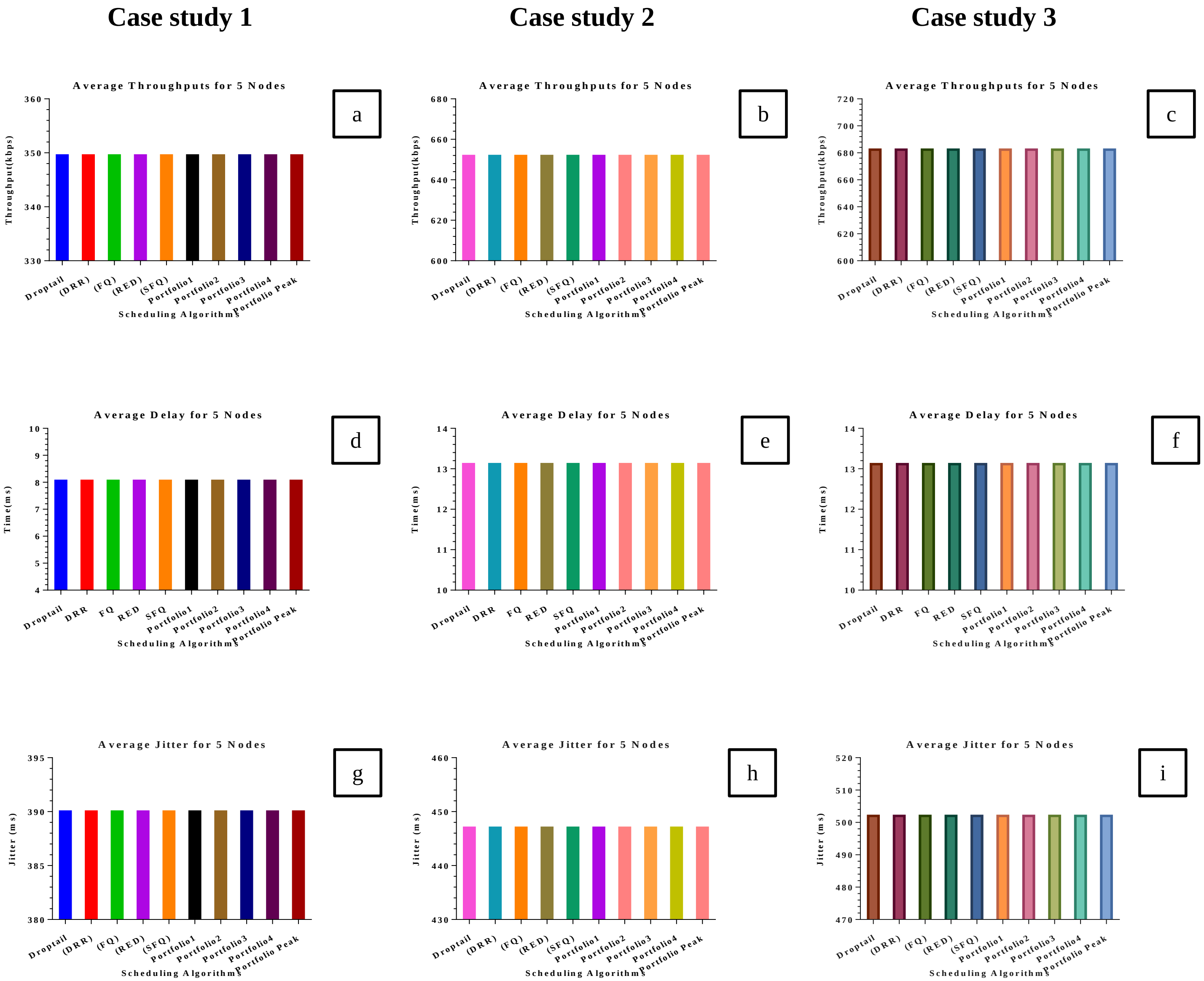


Figure 5.3: Average Performance for 5 Nodes

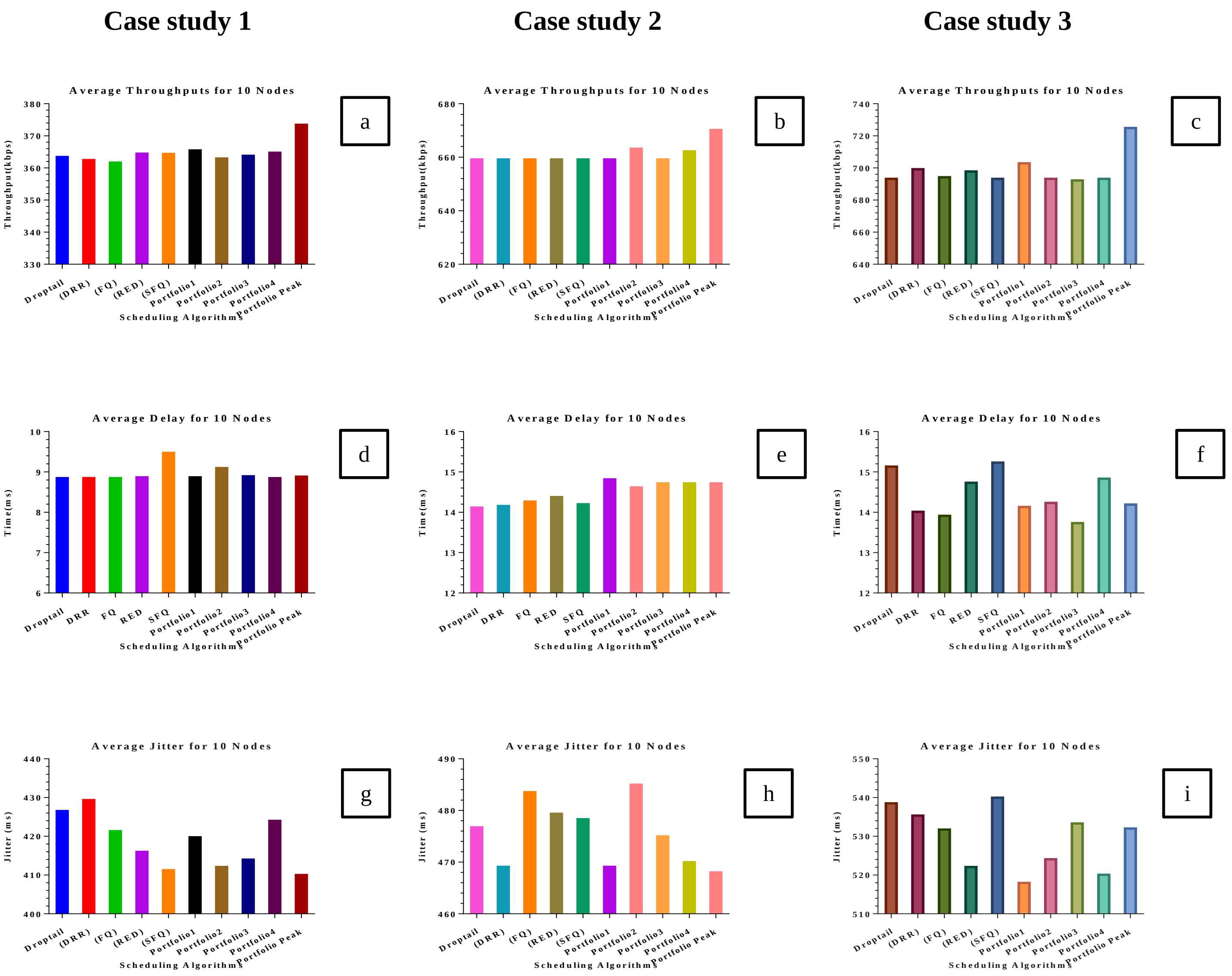


Figure 5.4: Average Performance for 10 Nodes

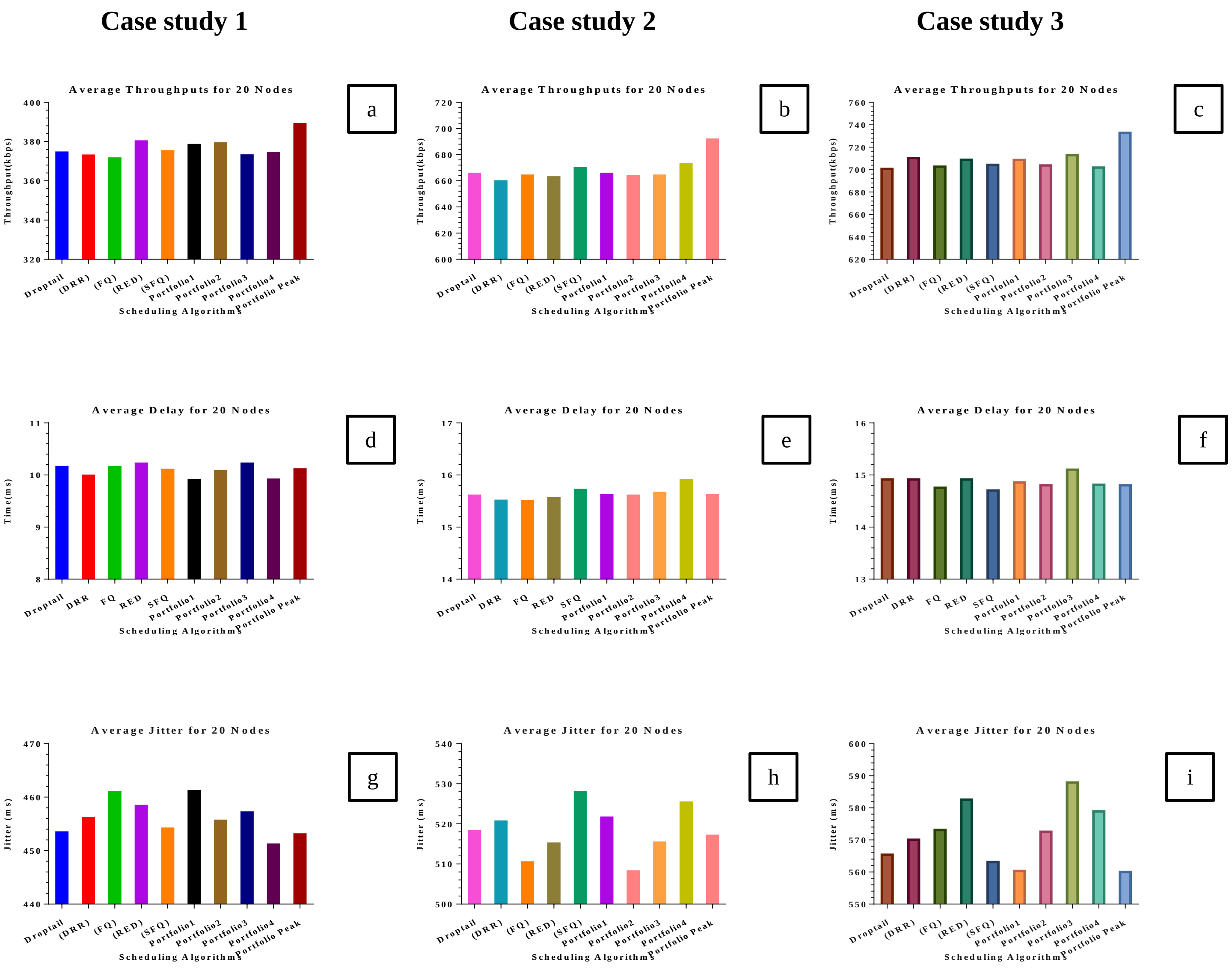


Figure 5.5: Average Performance for 20 Nodes

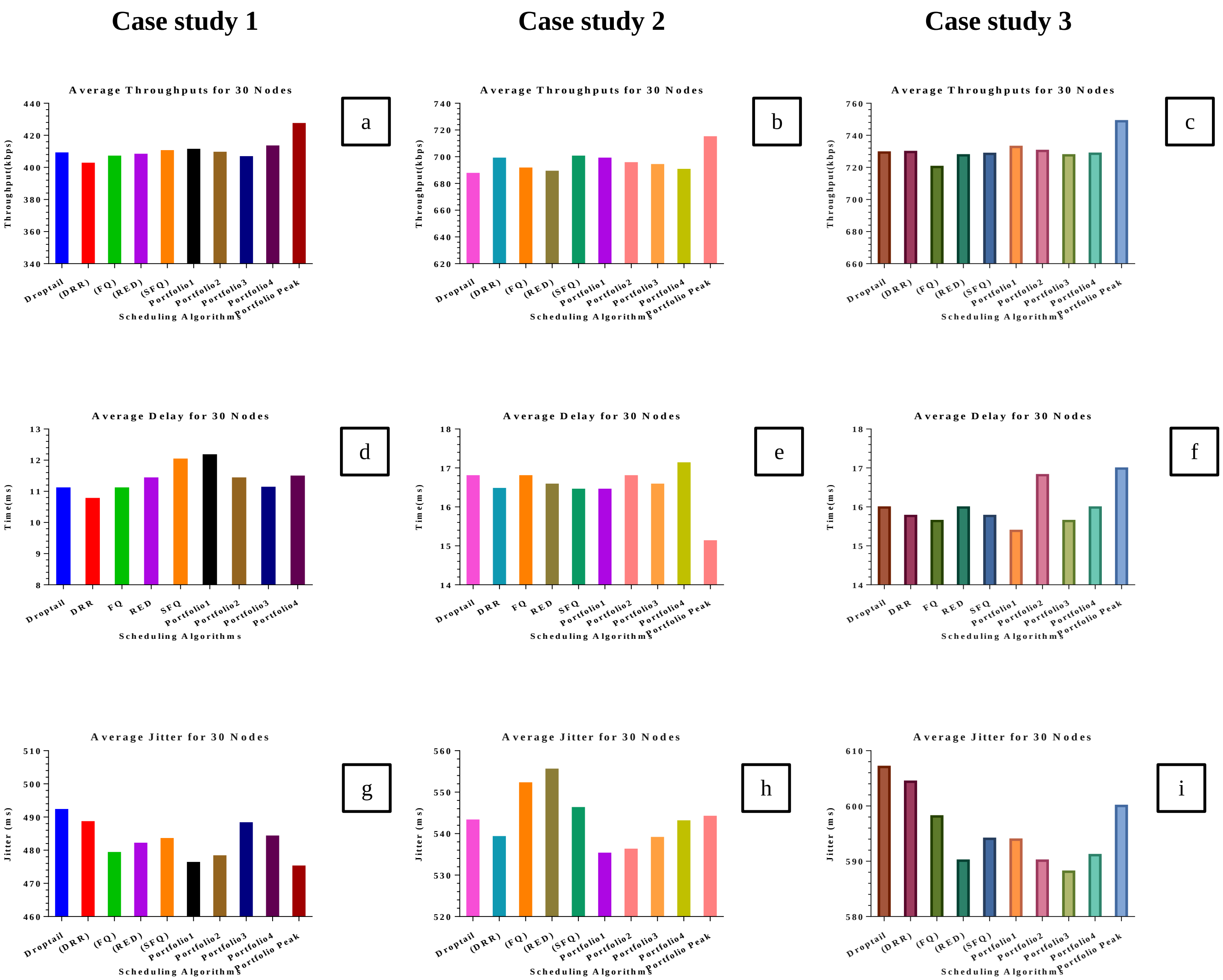


Figure 5.6: Average Performance for 30 Nodes

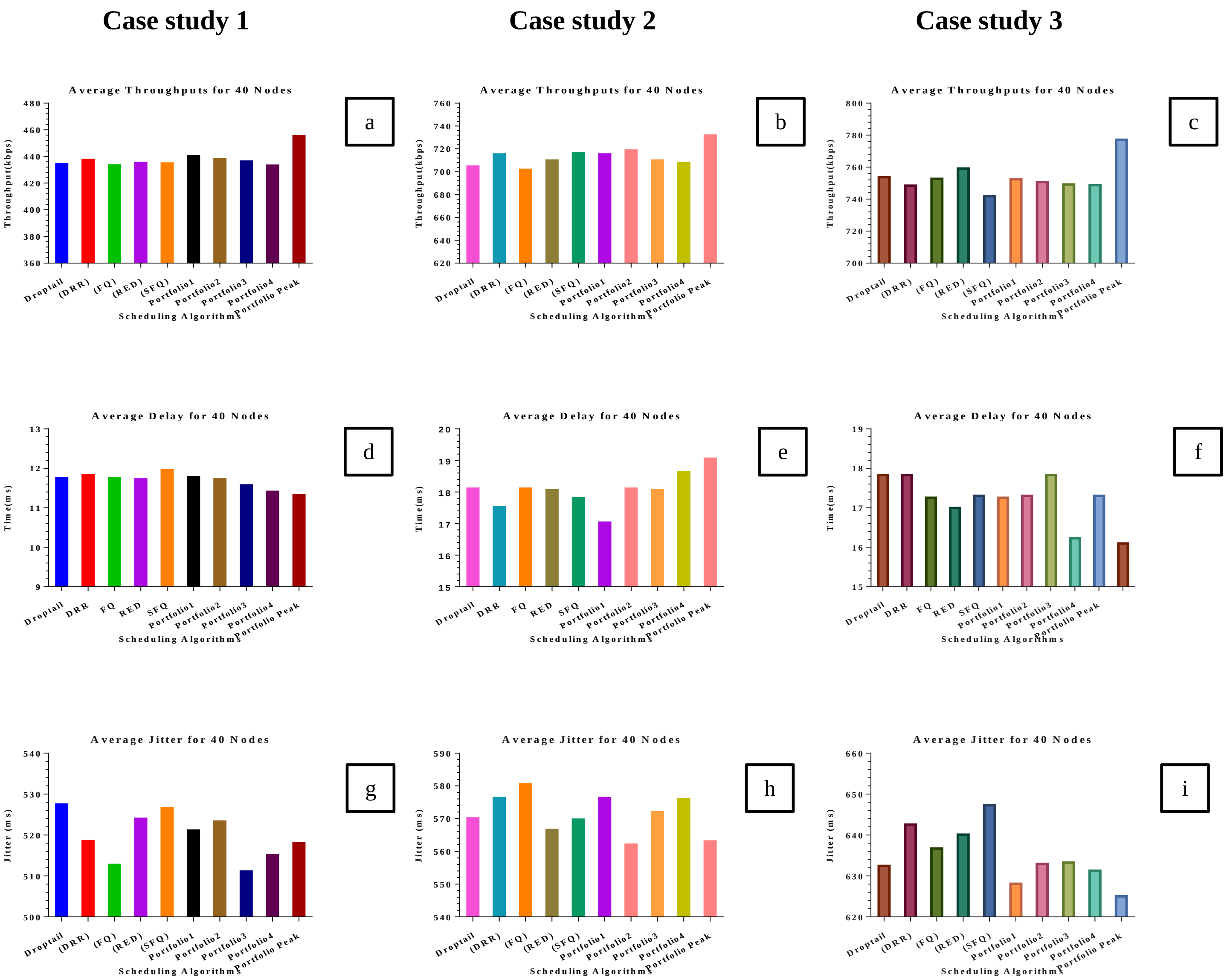


Figure 5.7: Average Performance for 40 Nodes

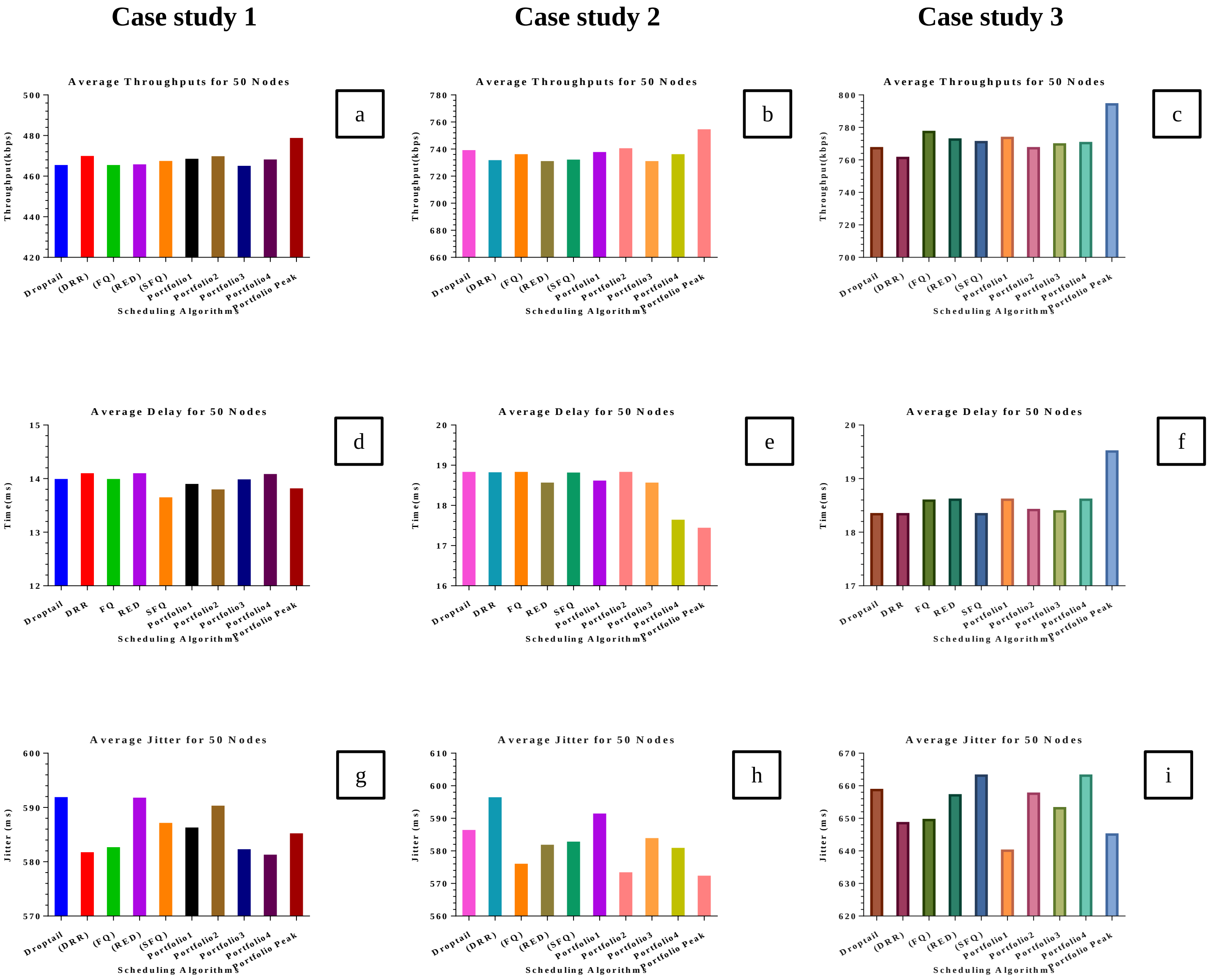


Figure 5.8: Average Performance for 50 Nodes

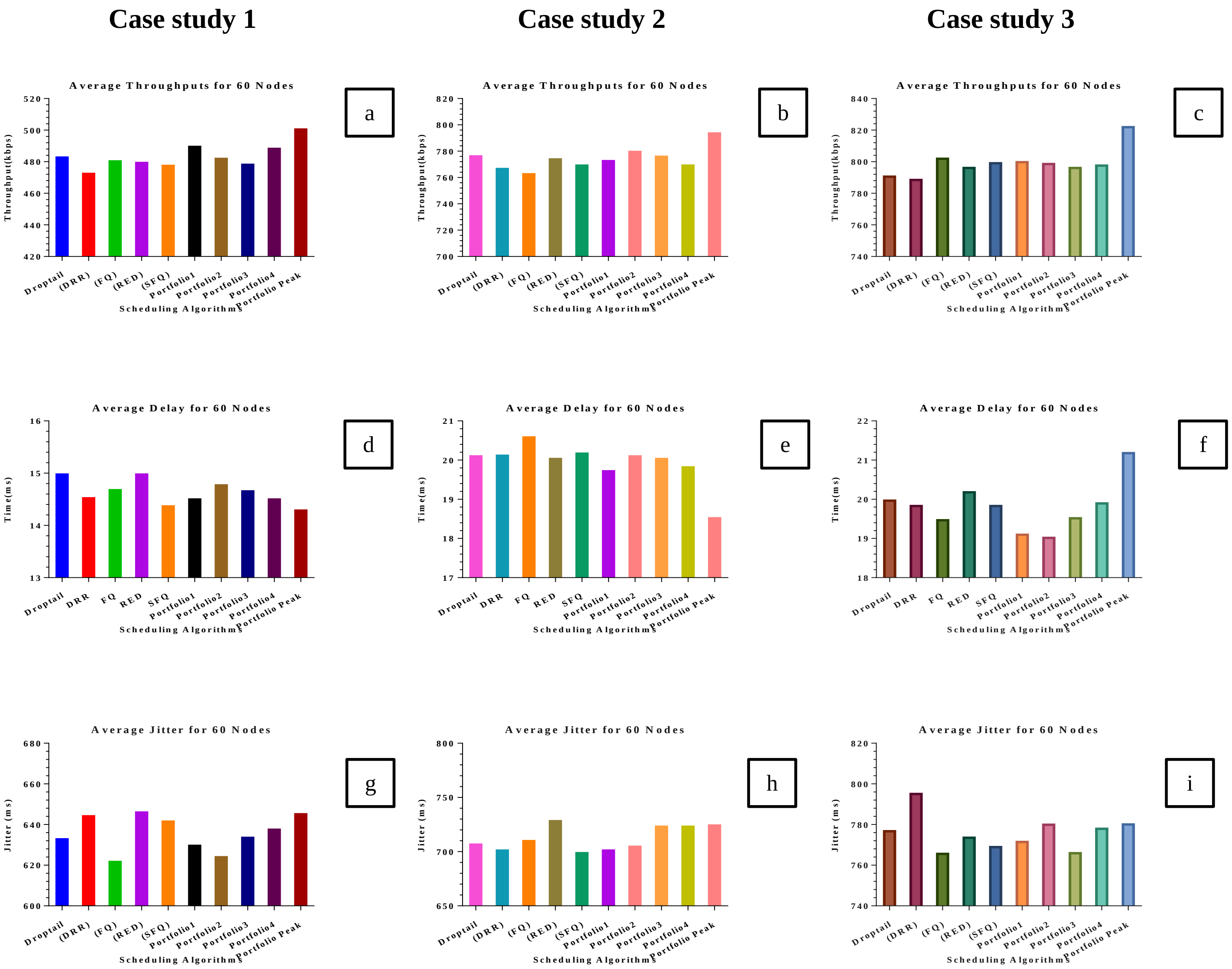


Figure 5.9: Average Performance for 60 Nodes

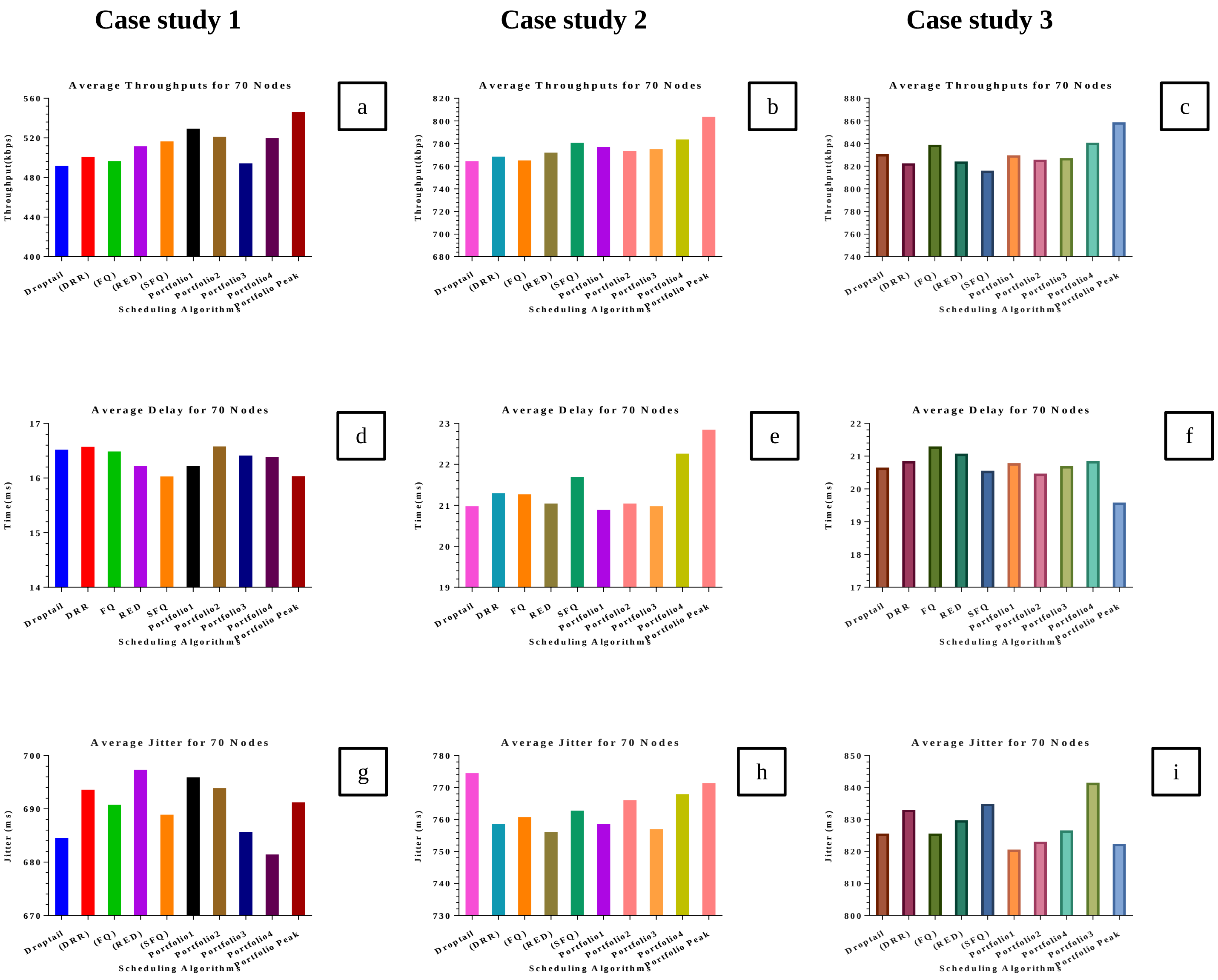


Figure 5.10: Average Performance for 70 Nodes

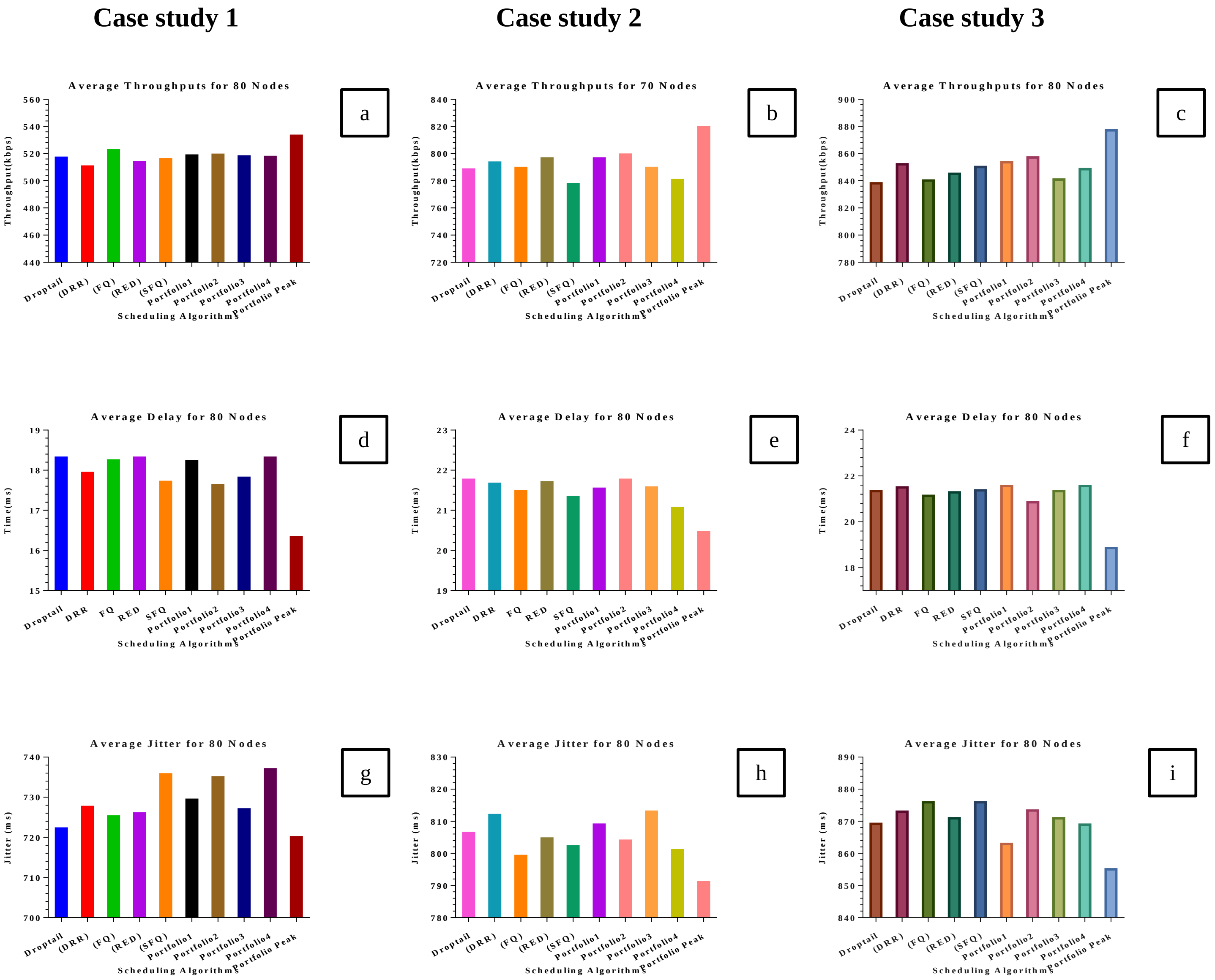


Figure 5.11: Average Performance for 80 Nodes

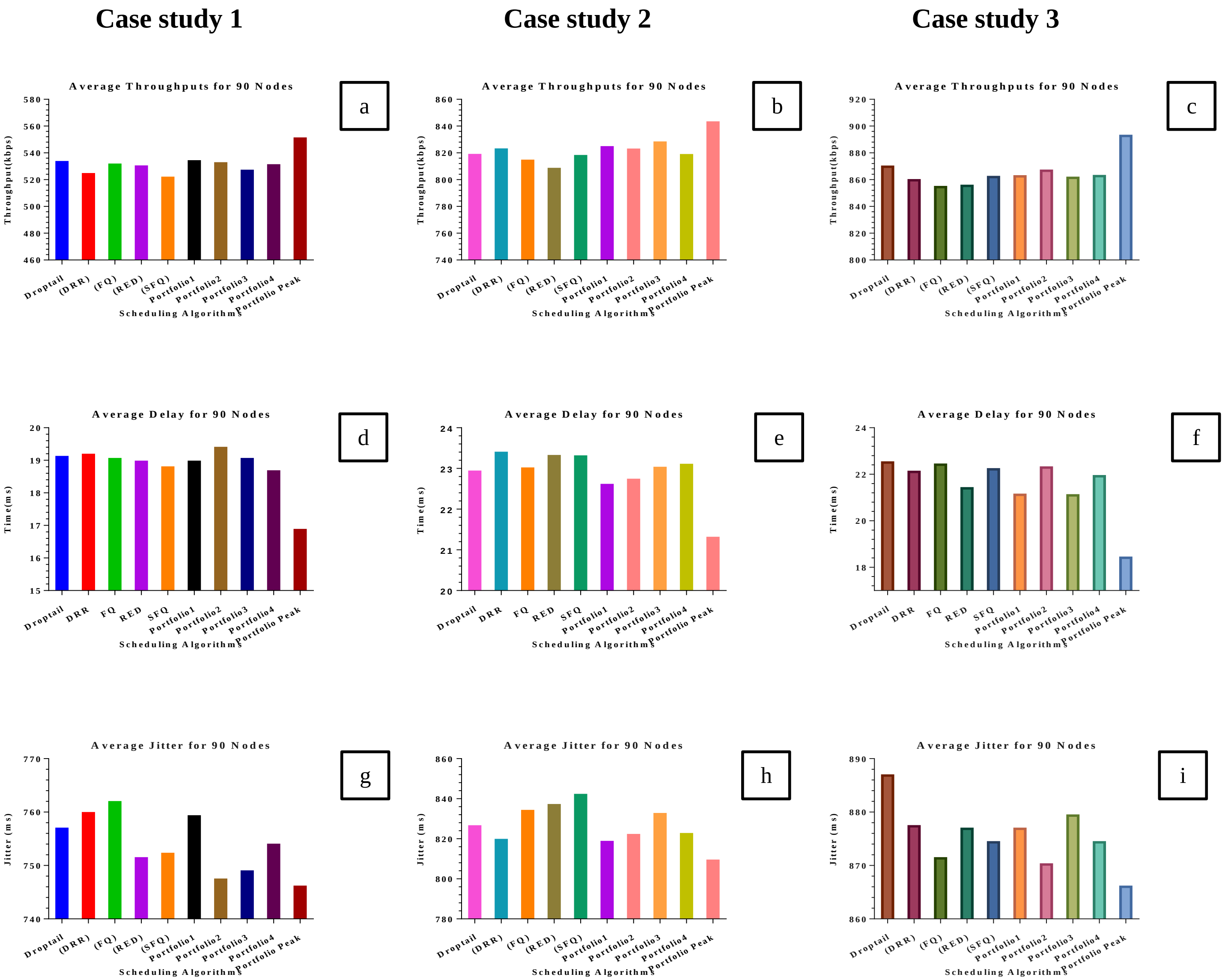


Figure 5.12: Average Performance for 90 Nodes

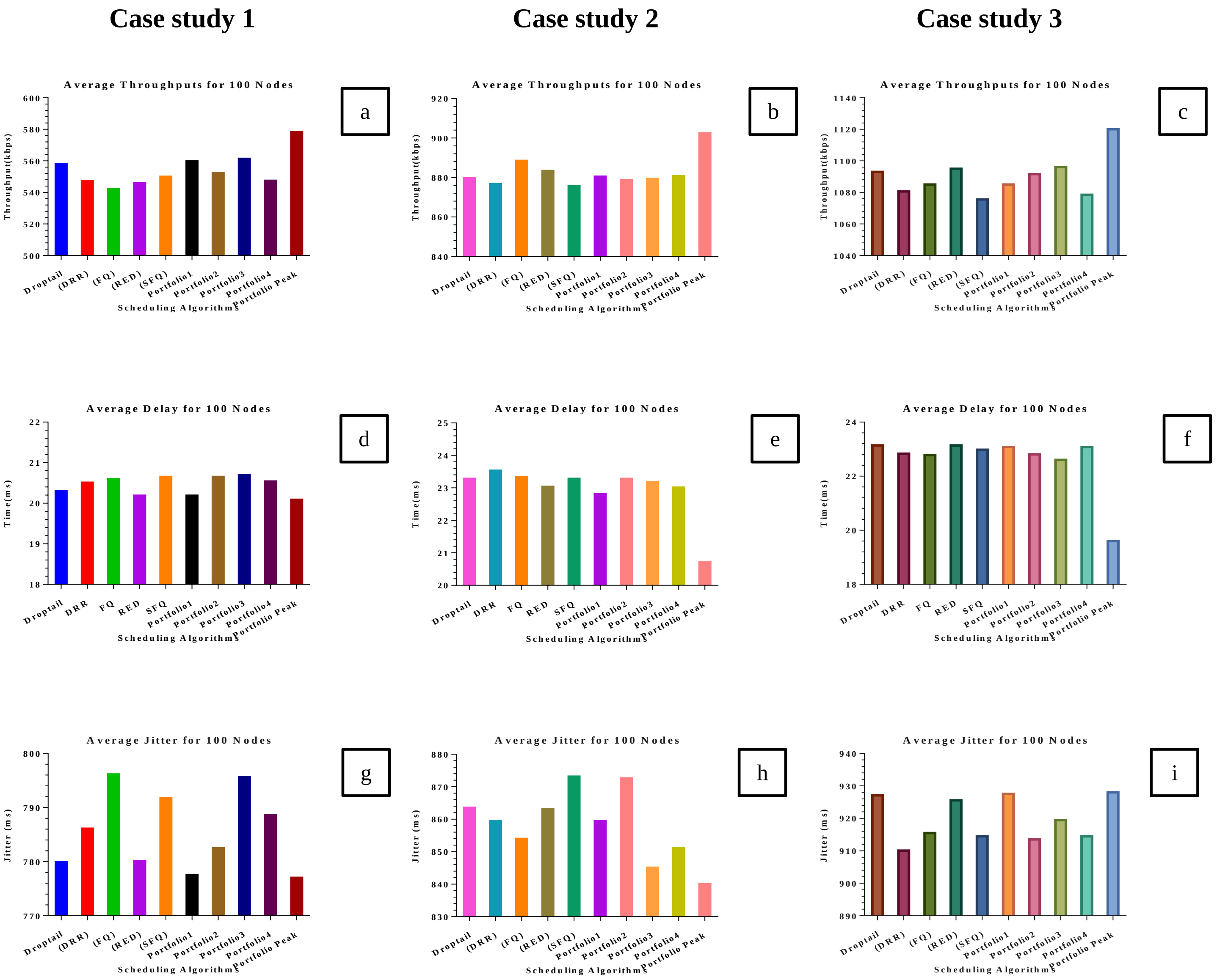


Figure 5.13: Average Performance for 100 Nodes

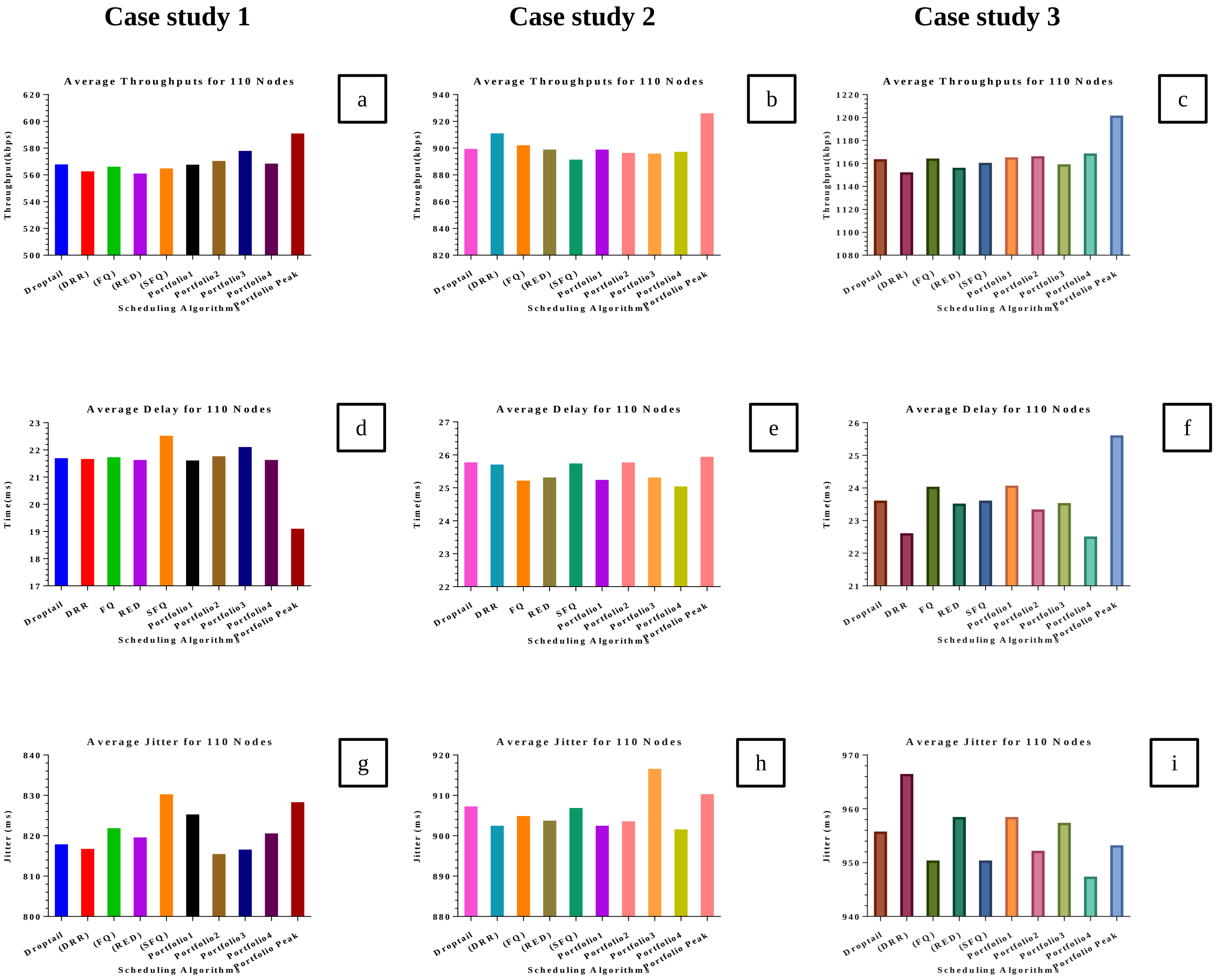


Figure 5.14: Average Performance for 110 Nodes

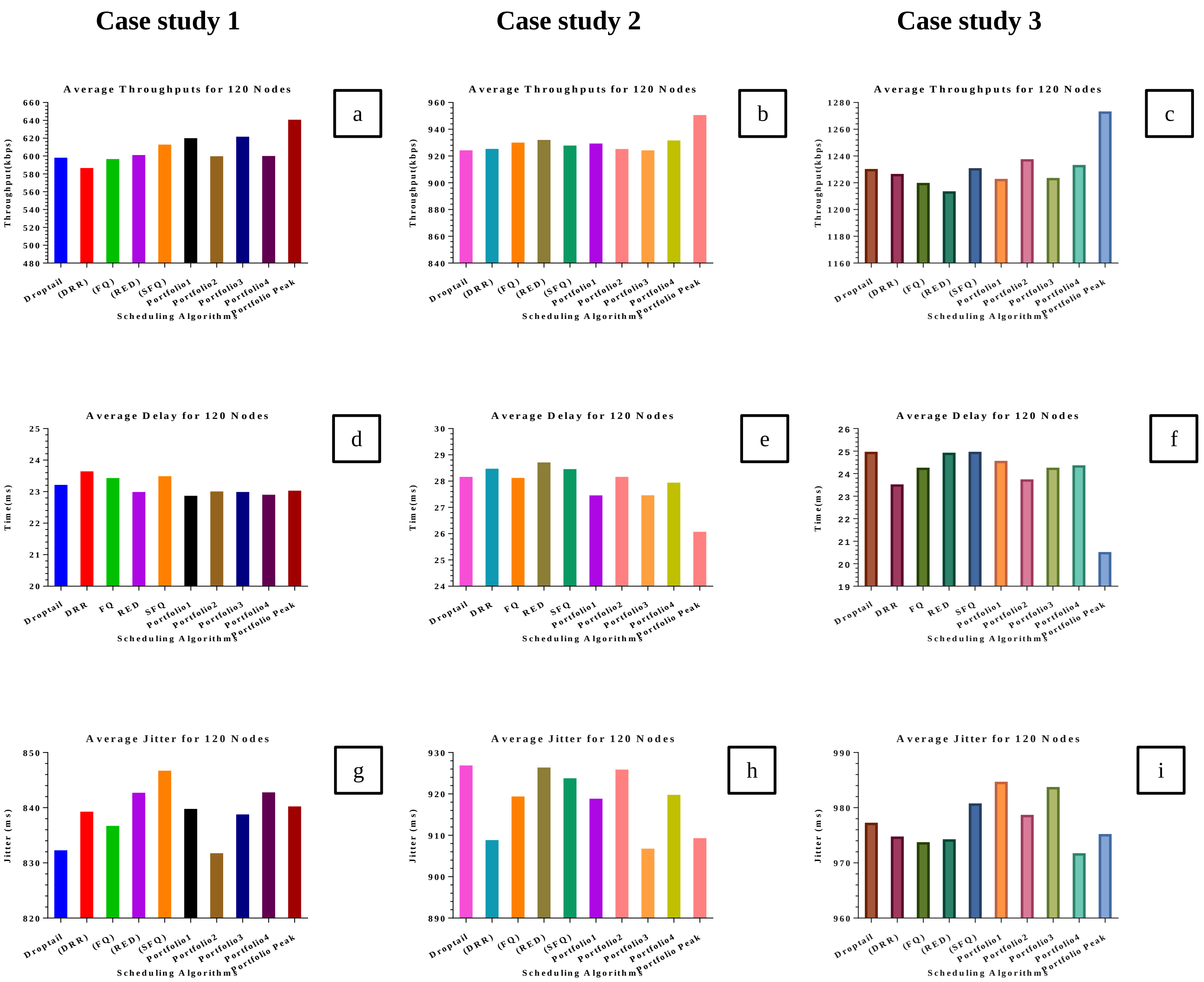


Figure 5.15: Average Performance for 120 Nodes

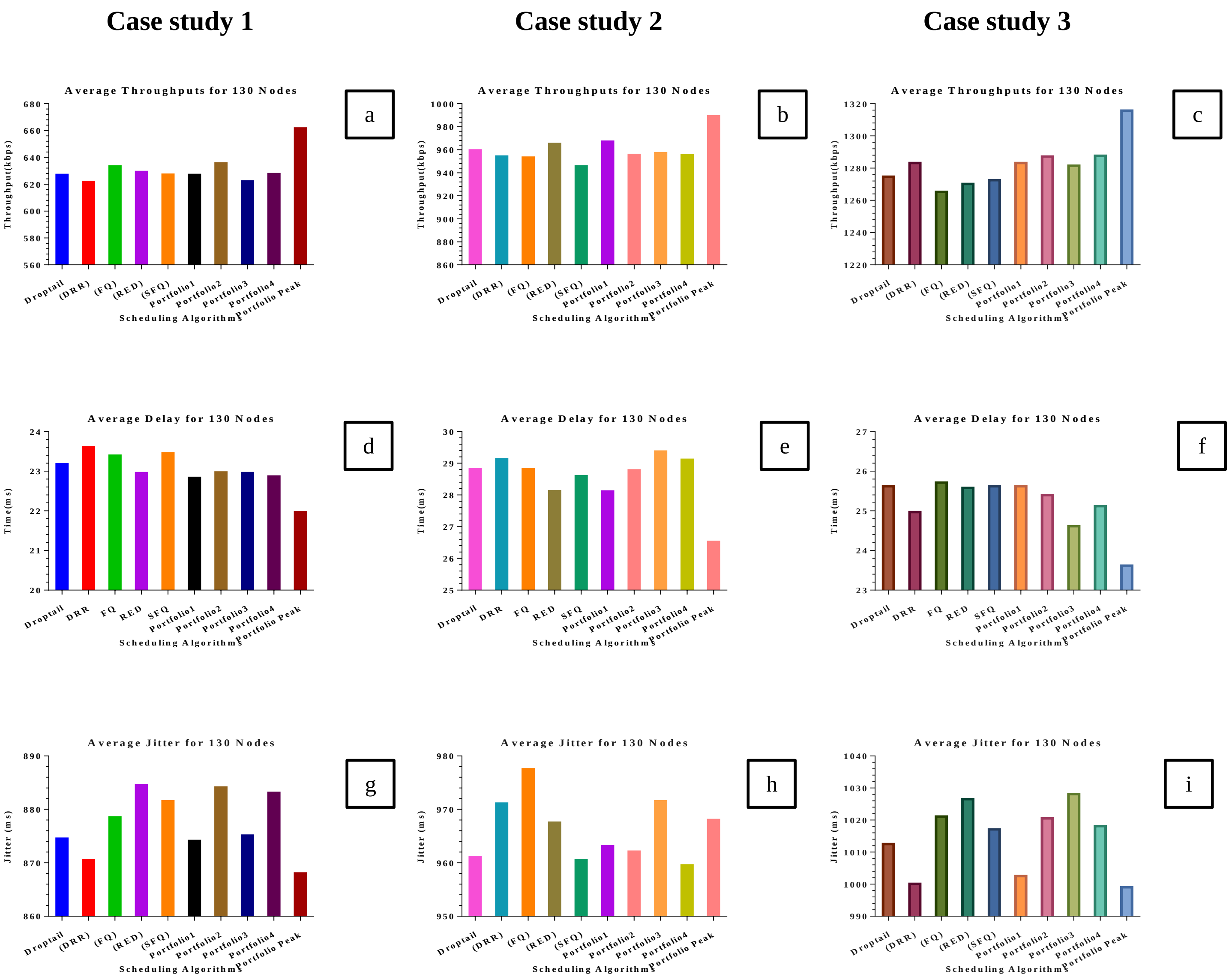


Figure 5.16: Average Performance for 130 Nodes

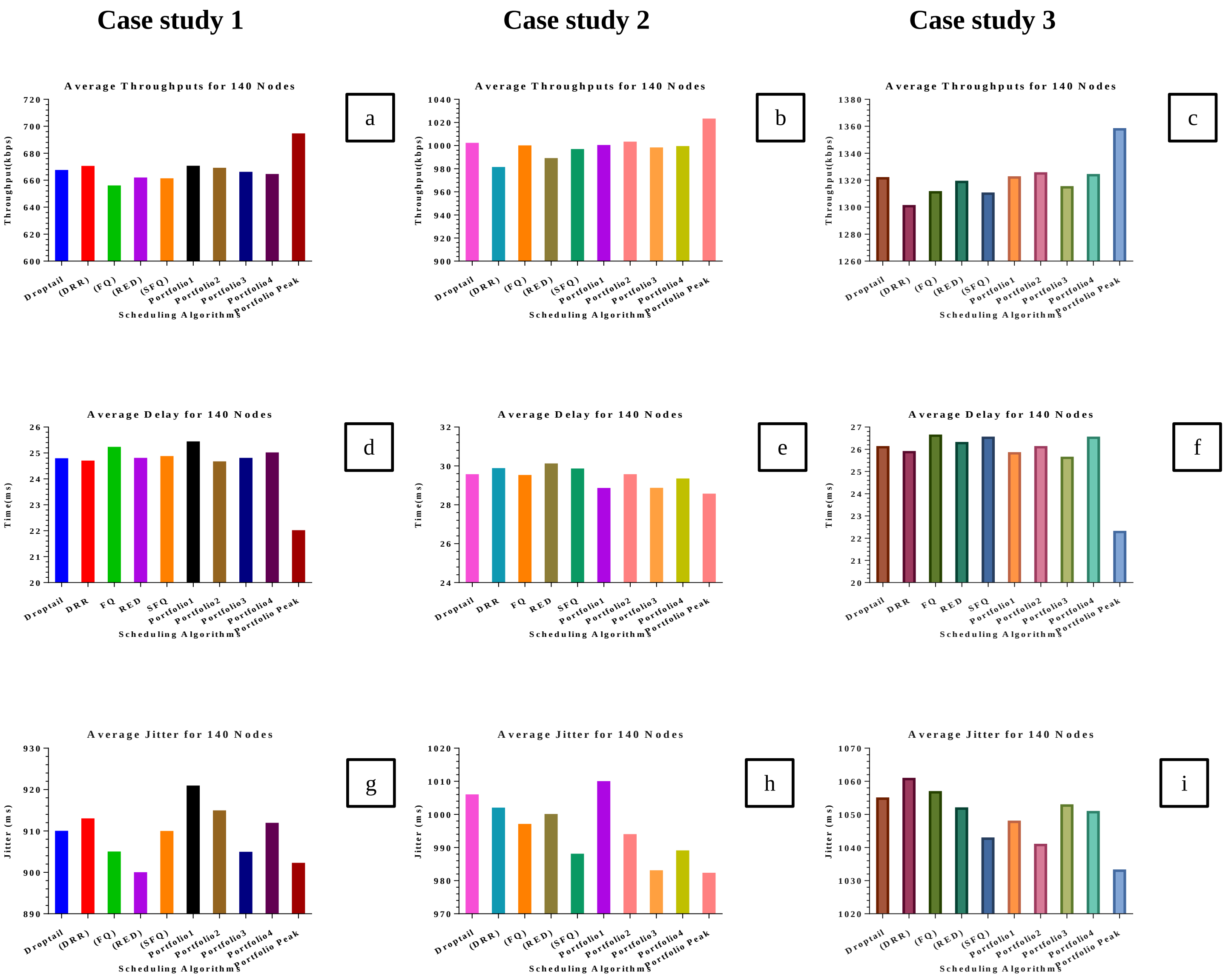


Figure 5.17: Average Performance for 140 Nodes

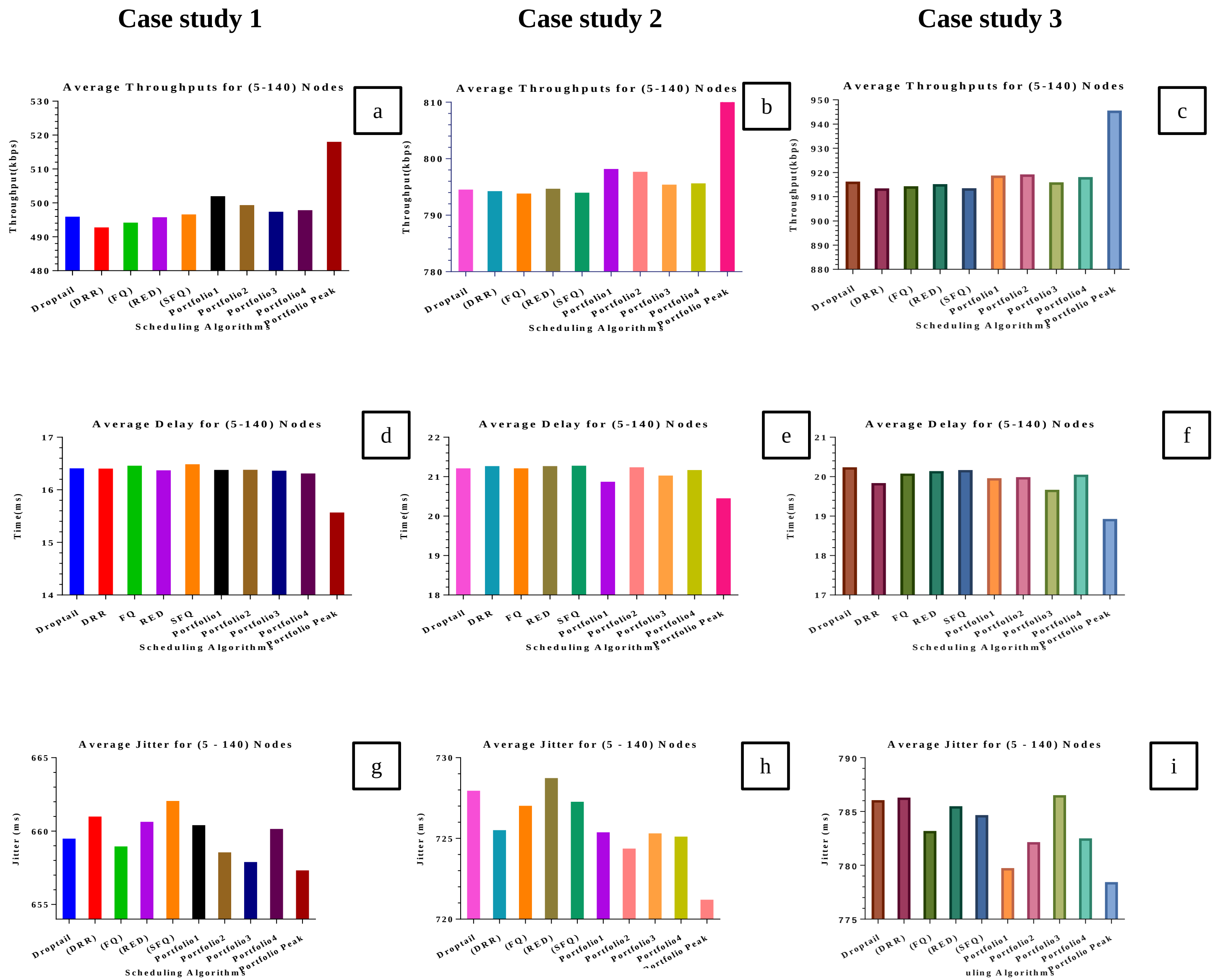


Figure 5.18: Average Performance for (5-140) Nodes

Mean performance for 140 node shows that the proposed PPA out-performs all conventional algorithms in terms of network Throughput, Delay and Jitter.

This occurs because the proposed PPA can obtain optimal scheduling even if unbalanced traffic occurs within the network and that by switching to the best scheduling that has higher performance. In addition, the proposed PPA out-performs the other four proposed fixed series of portfolio algorithms as a result of the PPA schedules being measured every second to enable switching to the best performance of the algorithm in a time, which reduces the occurrence of the congestion problem. In addition, the throughputs of the PPA significantly increase compared to conventional algorithms.

5.5 Summary

This chapter presented the proposed Portfolio Peak Algorithms (PPA) for WiMAX Networks. The researcher considered the cases (Case 1, Case 2, and Case 3) in terms of maximising the throughput to achieve optimal performance of the IEEE 802.16 standard. The proposed algorithm has been implemented and simulated in the NS2 simulator and its performance results have been compared with those conventional algorithms and to proposed fixed Portfolio algorithms. The simulation results showed that the proposed PP-algorithm achieved higher performance in terms of system throughput against all other algorithms. Moreover, in terms of average delay and average Jitter for the simulation scenarios of 5 Nodes to 140 Nodes, PPA achieved the best performance as shown in Figure 5.19.

In all performance parameters, the proposed PP algorithm exhibited high efficiency under examined 450 scenarios (simulation experiments), with the five mobile Nodes/stations or more contained in each scenario. The simulation results also demonstrated that the proposed algorithm is more efficient and more adaptable to different scenarios than the conventional algorithms.

In conclusion, altogether the simulation results indicated that the proposed PPA out-performs the existing algorithms in terms of network throughput in all cases, as well as in the total of the average number of 5 to 140 Nodes of Delay and Jitter as shown in figure 5.19. It has also been shown that performance improves by using PPA in all scenarios with a large or short number of Mobil nodes.

As a result, all data can be transmitted rapidly by applying the proposed PPA in terms of throughput maximising. Thus, Portfolio Peak algorithms are among the best current algorithms.

CHAPTER 6

CONCLUSION AND FUTURE WORK

6.1 Introduction

In recent times the increasing popularity of broadband wireless access networks and multimedia applications with their diverse quality of service (QoS) requirements has continuously fuelled the needs for ever more powerful networks. One example of such a network is called WiMAX which is driven by WiMAX Forum based on IEEE 802.16 Wireless MAN standard. However, IEEE 802.16 standard does not specify a bandwidth allocation algorithm to guarantee QoS. This is purposely done in order to allow service providers and vendors to innovate in this area and distinguish their products.

WiMAX has the ability to provide QoS requirements for a variety of services as it is developed, to meet the anticipated growth in the worldwide market for high bandwidth and real-time services. However, provisioning QoS mainly depends on the scheduling algorithm at both medium access control (MAC) and physical layer (PHY), respectively. Moreover, QoS provisioning of real-time and non-real-time applications are frequently unstable due to insufficient allocation of bandwidth, which, in turn, leads to the deterioration of overall system throughput.

This chapter provides some concluding observations. In addition, it highlights some possible future research directions in order to guide and offer other researchers an insight into extending this work.

6.2 Evaluation of The Thesis Contributions

Within this framework four principle contributions are proposed:

Firstly, a new mode of scheduling algorithms has been developed which uses the requesting and granting of permission based on performance as a method of scheduling algorithms.

Results show that the new scheduling algorithms achieve best performance in terms of throughput and delay.

Secondly, the proposed algorithms ensure that the scheduling transmissions of each user are maintained by assigning and switching to the highest scheduling algorithm of performance in each time frame which is proportional to the users' application demands.

Simulation results demonstrate that the proposed peak algorithm serves application users' demands by achieving the highest performance due to its ability to assign and switch to the highest algorithm of performance, thus outperforming the fixed algorithm.

Thirdly, the transmission opportunity for each user will be assigned based on Quality of Service (QoS).

This is challenging to achieve due to unpredictable channel conditions such as signal fading and frequency interference. Different types of application traffic require a variety of qualities from a network. Some types will need a higher priority over others. In order to support these requirements, QoS parameters are utilised. WiMAX assures guaranteed QoS by including several mechanisms at the MAC layer such as admission control and scheduling algorithms.

Finally, the new proposed algorithms aim at reducing the resulting delay and maximise throughput in IEEE 802.16 standard.

Results obtained from the comparison between conventional and portfolio peak algorithms demonstrate substantial reduction in delay.

6.3 Conclusion

This thesis focused on scheduling algorithm performance evaluation for the WiMAX Networks. Fixed portfolio algorithms and portfolio peak algorithms have been proposed, implemented and evaluated to provide the best QoS support for real-time and non-real-time services. The performance of the proposed algorithms in the WiMAX Network has been carefully investigated and evaluated using the NS2 simulator. The conventional Droptail, DRR, FQ, RED and SFQ algorithms have been utilised to evaluate and investigate the system performance with different numbers of Mobile Nodes. The performance of the proposed Portfolio and Portfolio Peak algorithms has been compared with those of conventional algorithms under three case studies within several scenarios.

The simulation results showed that the proposed Portfolio algorithms could noticeably improve the WiMAX network performance with diverse numbers of mobile stations (MS). Performances of the proposed Portfolio Peak Algorithms achieved optimal performance of the IEEE 802.16 standard in terms of throughput. The simulation results highlighted that the proposed PP-algorithm achieved higher performance in terms of system throughput against all other algorithms in real-time and non-real time applications, including this portfolio and portfolio peak which proved to result in less average delay and jitter and proved to be capable of outperforming.

Both proposed algorithms (Fixed Portfolio and Portfolio Peak) sustain the efficiency of the reduced delay and Jitter that plague the legacy of IEEE 802.16 and guarantee the maximising of throughput.

The NS2 was used to conduct the evaluation of the proposed portfolio. The simulation results confirmed our expectations and demonstrated that the proposed portfolio algorithms significantly improve the throughput and that improvement is achieved by minimising delay and Jitter compared to the conventional algorithms.

Both Portfolio algorithms aim to satisfy QoS requirements of the diverse traffic type demands and, at the same time, maximise bandwidth efficiency. However, these algorithms have been developed with different approaches.

The Proposed Portfolio Peak Algorithms (PPPA) perform best with all of the scenarios and case studies which increase and maximise throughput. The throughput with PPA is greater than the fixed Portfolio algorithms.
On the whole, the proposed algorithms confirmed the earlier hypothesis of the researcher's investigations and substantiate the claim that within WIMAX systems performance can be significantly amplified by using Portfolio of algorithms.

Finally, the results of this research are valuable for the development and standardisation of the next-generation wireless communication systems. The scheduling algorithms have attracted a lot of attention in the communication industry and the academic world in the areas of WiMAX network research and development.

6.4 Future Work

In this thesis, performances of the proposed algorithms have been evaluated and intensively investigated together with bandwidth allocation of the conventional algorithms. Scheduling algorithms have been developed by many researchers to enhance the efficiency of resource allocation and transmission opportunities for maximising the throughput in the WiMAX network systems to achieve higher performance.

However, it is suggested here that any related work in the future to should consider further design and evaluation of WiMAX network for Fixed Portfolio algorithms and Portfolio Peak algorithms using a fuzzy logic-based approach that grants the optimal bandwidth required by each service, based on maximum throughput and minimising delay metrics.

References

Borin, J. F. and N. L. da Fonseca (2008). "Simulator for WiMAX networks." Simulation Modelling Practice and Theory **16**(7): 817-833.

Chen, Y.-W., Y.-Y. Chu and I.-H. Peng (2014). "Energy-saving centric uplink scheduling scheme for broadband wireless access networks." EURASIP Journal on Wireless Communications and Networking **2014**(1): 70.

Fall, K. and K. Varadhan (2005). "The ns Manual (formerly ns Notes and Documentation)." The VINT project **47**: 19-231.

Group, I. W. (2004). "IEEE standard for local and metropolitan area networks-Part 16: Air interface for fixed broad-band wireless access systems." IEEE Std. 802.16-2004.

Manolis Katevenis, Stefanos Sidiropoulos and Costas Courcoubetis (1991). "Weighted round-robin cell multiplexing in a general-purpose ATM switch chip." IEEE Journal on selected Areas in Communications **9**(8): 1265-1279.

Nguyen, T. M., T. Yim, Y. Jeon, Y. Kyung and J. Park (2014). "QoS-aware dynamic resource allocation for wireless broadband access networks." EURASIP Journal on Wireless Communications and Networking **2014**(1): 104.

Sayenko, A., O. Alanen and T. Hämäläinen (2008). "Scheduling solution for the IEEE 802.16 base station." Computer Networks **52**(1): 96-115.

Shimonishi, H., M. Yoshida, R. Fan and H. Suzuki (1997). An improvement of weighted round robin cell scheduling in ATM networks. GLOBECOM 97. IEEE Global Telecommunications Conference. Conference Record, IEEE.

Sikora, A. and V. F. Groza (2005). Coexistence of IEEE802. 15.4 with other Systems in the 2.4 GHz-ISM-Band. 2005 IEEE Instrumentationand Measurement Technology Conference Proceedings, IEEE.

Wang, C., W.-J. Yan and H.-K. Lo (2012). "Dynamic admission control and bandwidth reservation for IEEE 802.16 e mobile WiMAX networks." EURASIP Journal on Wireless Communications and Networking **2012**(1): 143.

Wei Nie, Houjun Wang and Jong Hyuk Park (2011). "Packet Scheduling with QoS and Fairness for Downlink Traffic in WiMAX Networks." JIPS **7**(2): 261-270.

Yoon, H.-H., H. Kim, C. Oh and K. Kim (1999). A queue length-based scheduling scheme in ATM networks. Proceedings of IEEE. IEEE Region 10 Conference. TENCON 99.'Multimedia Technology for Asia-Pacific Information Infrastructure'(Cat. No. 99CH37030), IEEE.

AKASHDEEP, KAHLON, K. & KAUSHAL, M. 2016. Analysis of a queue length aware and latency guaranteed fuzzy-based adaptive resource allocator for WiMAX networks. *Optik-International Journal for Light and Electron Optics,* 127**,** 357-367.

AL-HAWAWREH, M. S. & ZREIKAT, A. I. 2017. Performance Analysis of a WIMAX Network in Different Propagation Models. *International Journal of Computer Science and Information Security,* 15**,** 603.

ALAVI, H. S., MOJDEH, M. & YAZDANI, N. A quality of service architecture for IEEE 802.16 standards. 2005 Asia-Pacific Conference on Communications, 2005. IEEE, 249-253.

ALI-YAHIYA, T., BEYLOT, A.-L. & PUJOLLE, G. 2010. Downlink resource allocation strategies for OFDMA based mobile WiMAX. *Telecommunication Systems,* 44**,** 29-37.

ALLIANCE December 2008. NGMN Recommendation on SON and O&M Requirements. *Next Generation Mobile Networks*, *White paper*.

ALLIANCE May 2007. NGMN Use Cases related to Self Organising Network, Overall Description. *Next Generation Mobile Networks*.

ANCEAUME, E., DÉFAGO, X., GRADINARIU, M. & ROY, M. Towards a theory of self-organization. International Conference On Principles Of Distributed Systems, 2005. Springer, 191-205.

ANDREWS, J. G., GHOSH, A. & MUHAMED, R. 2007. *Fundamentals of WiMAX: understanding broadband wireless networking*, Pearson Education.

ANOUARI, T. & HAQIQ, A. Comparative study and analysis of voip traffic over wimax using different service classes. 2012 Next Generation Networks and Services (NGNS), 2012. IEEE, 87-93.

ATHANASOPOULOS, N., TSIAKAS, P., VOUDOURIS, K., MANOR, D., MOR, A. & AGAPIOU, G. An IEEE802. 16j prototype relay station architecture. Melecon 2010-2010 15th IEEE Mediterranean Electrotechnical Conference, 2010. IEEE, 1247-1252.

BACIOCCOLA, A., CICCONETTI, C., EKLUND, C., LENZINI, L., LI, Z. & MINGOZZI, E. 2010. IEEE 802.16: History, status and future trends. *Computer Communications,* 33**,** 113-123.

BAI, B., CHEN, W., CAO, Z. & LETAIEF, K. B. 2010. Uplink cross-layer scheduling with differential QoS requirements in OFDMA systems. *EURASIP Journal on Wireless Communications and Networking,* 2010**,** 168357.

BALL, C. F., TREML, F., GAUBE, X. & KLEIN, A. Performance analysis of temporary removal scheduling applied to mobile WiMAX scenarios in tight frequency reuse. 2005 IEEE 16th International Symposium on Personal, Indoor and Mobile Radio Communications, 2005. IEEE, 888-894.

BAROOAH, M., CHAKRABORTY, S., NANDI, S. & KOTWAL, D. 2013. An architectural framework for seamless handoff between IEEE 802.11 and UMTS networks. *Wireless networks,* 19**,** 411-429.

BENNETT, J. C. & ZHANG, H. WF2Q: worst-case fair weighted fair queueing. Proceedings of IEEE INFOCOM'96. Conference on Computer Communications, 1996. IEEE, 120-128.

BRAHMIA, M. E. A., ABOUAISSA, A. & LORENZ, P. 2014. Adaptive scheduling mechanism for IPTV over WiMAX IEEE 802.16 j networks. *International Journal of Communication Systems,* 27**,** 1009-1019.

BRESLAU, L., ESTRIN, D., FALL, K., FLOYD, S., HEIDEMANN, J., HELMY, A., HUANG, P., MCCANNE, S., VARADHAN, K. & XU, Y. 2000. Advances in network simulation. *Computer,* 33**,** 59-67.

CAO, M., MA, W., ZHANG, Q. & WANG, X. 2007. Analysis of IEEE 802.16 mesh mode scheduler performance. *IEEE Transactions on Wireless Communications,* 6**,** 1455-1464.

CAO, M., MA, W., ZHANG, Q., WANG, X. & ZHU, W. Modelling and performance analysis of the distributed scheduler in IEEE 802.16 mesh mode. Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing, 2005. ACM, 78-89.

CAO, Y. & LI, V. O. 2001. Scheduling algorithms in broadband wireless networks. *Proceedings of the IEEE,* 89**,** 76-87.

CAO, Y., LIU, Z. & YANG, Y. A centralized scheduling algorithm based on multi-path routing in WiMAX mesh network. 2006 International Conference on Wireless Communications, Networking and Mobile Computing, 2006. IEEE, 1-4.

CHAN, Y.-C., WU, C.-L. & LAI, C.-Y. Congestion-aware downlink scheduling for IEEE 802.16 j multi-hop relay networks. Proceedings of the 4th Annual International Conference on Wireless Internet, 2008. ICST (Institute for Computer Sciences, Social-Informatics and …, 36.

CHEN, Y.-W., CHU, Y.-Y. & PENG, I.-H. 2014. Energy-saving centric uplink scheduling scheme for broadband wireless access networks. *EURASIP Journal on Wireless Communications and Networking,* 2014**,** 70.

CHENG, S.-M., LIN, P., HUANG, D.-W. & YANG, S.-R. A study on distributed/centralized scheduling for wireless mesh network. Proceedings of the 2006 international conference on Wireless communications and mobile computing, 2006. ACM, 599-604.

CHOI, K. W., JEON, W. S. & JEONG, D. G. 2008. Resource allocation in OFDMA wireless communications systems supporting multimedia services. *IEEE/ACM transactions on networking,* 17**,** 926-935.

CICCONETTI, C., ERTA, A., LENZINI, L. & MINGOZZI, E. 2007. Performance evaluation of the IEEE 802.16 MAC for QoS support. *IEEE Transactions on mobile computing,* 6.

CICCONETTI, C., LENZINI, L., MINGOZZI, E. & EKLUND, C. 2006. Quality of service support in IEEE 802.16 networks. *IEEE network,* 20**,** 50-55.

COMMITTEE, I. L. M. S. 2004. IEEE standard for local and metropolitan area networks part 16: Air interface for fixed broadband wireless access systems. *IEEE Std 802.16^< TM>-2004.*

DANIELE TARCHI, ROMANO FANTACCI & MARCO BARDAZZI. Quality of service management in IEEE 802.16 wireless metropolitan area networks. 2006 IEEE International Conference on Communications, 2006. IEEE, 1789-1794.

DENG, D. J., CHANG, L. W., KE, C. H., HUANG, Y. M. & CHANG, J. M. 2009. Delay constrained uplink scheduling policy for rtPS/ertPS service in IEEE 802.16 e BWA systems. *International Journal of Communication Systems*, 22**,** 119-133.

DIXIT, S. 2004. Self-organization of complex networks applied to wireless world systems. *Wireless Personal Communications,* 29**,** 63-70.

DU, P., JIA, W., HUANG, L. & LU, W. Centralized scheduling and channel assignment in multi-channel single-transceiver wimax mesh network. 2007 IEEE Wireless Communications and Networking Conference, 2007. IEEE, 1734-1739.

EKLUND, C., MARKS, R. B., STANWOOD, K. L. & WANG, S. 2002. IEEE Standard 802.16: A Technical Overview of the WirelessMAN Air Interface for Broadband Wireless Access. *IEEE communications magazine,* 40**,** 98-107.

ELGEREDA, J., MOGHADDAMA, A. S. & VEDDERB, B. 2010 &. How Quality of Service (QoS) is achieved in WiMAX (IEEE 802.16).

ERGEN, M., COLERI, S. & VARAIYA, P. 2003. QoS aware adaptive resource allocation techniques for fair scheduling in OFDMA based broadband wireless access systems. *IEEE Transactions on broadcasting,* 49**,** 362-370.

ESMAILPOUR, A. & NASSER, N. 2011. Dynamic QoS-Based Bandwidth Allocation Framework for Broadband Wireless Networks. *IEEE Transactions on Vehicular Technology,* 60**,** 2690-2700.

FALL, K. & VARADHAN, K. 2005. The ns Manual (formerly ns Notes and Documentation). *The VINT project,* 47**,** 19-231.

FALL, K. & VARADHAN, K. 2011. The ns Manual (formerly ns Notes and Documentation). *The VINT project***,** 1-434.

FONG, B., ANSARI, N., FONG, A. C. M. & HONG, G. Y. 2004. On the scalability of fixed broadband wireless access network deployment. *IEEE Communications Magazine,* 42**,** S12-S18.

FU, L., CAO, Z. & FAN, P. Spatial reuse in IEEE 802.16 based wireless mesh networks. IEEE International Symposium on Communications and Information Technology, 2005. ISCIT 2005., 2005. IEEE, 1358-1361.

GAN, W., XIAN, J., XIE, X. & RAN, J. A cross-layer designed scheduling algorithm for WiMAX uplink. 2009 9th International Conference on Electronic Measurement & Instruments, 2009. IEEE, 1-122-1-127.

GENC, V., MURPHY, S., YU, Y. & MURPHY, J. 2008. IEEE 802.16 J relay-based wireless access networks: an overview. *IEEE wireless communications,* 15**,** 56-63.

GHOSH, D., GUPTA, A. & MOHAPATRA, P. 2008. Scheduling in multihop WiMAX networks. *ACM SIGMOBILE Mobile Computing and Communications Review,* 12**,** 1-11.

GHOSH, D., GUPTA, A. & MOHAPATRA, P. Adaptive scheduling of prioritized traffic in IEEE 802.16 j wireless networks. 2009 IEEE International Conference on

Wireless and Mobile Computing, Networking and Communications, 2009. IEEE, 307-313.

GIRICI, T., ZHU, C., AGRE, J. R. & EPHREMIDES, A. 2010. Proportional fair scheduling algorithm in OFDMA-based wireless systems with QoS constraints. *Journal of communications and networks,* 12**,** 30-42.

GKELIAS, A. & LEUNG, K. K. A distributed opportunistic scheduling scheme for wireless network coding. 2009 6th International Symposium on Wireless Communication Systems, 7-10 Sept. 2009 2009. 313-317.

GLISIC, S. & MAKELA, J.-P. Advanced wireless networks: 4G technologies. 2006 IEEE Ninth International Symposium on Spread Spectrum Techniques and Applications, 2006. IEEE, 442-446.

GOLESTANI, S. J. A self-clocked fair queueing scheme for broadband applications. Proceedings of INFOCOM'94 Conference on Computer Communications, 1994. IEEE, 636-646.

GROUP, I. W. 2004. IEEE standard for local and metropolitan area networks-Part 16: Air interface for fixed broad-band wireless access systems. *IEEE Std. 802.16-2004*.

HAIDAR SAFA, HASSAN ARTAIL, MARCEL KARAM, RAWAN SOUDAH & SAMAR KHAYAT. New Scheduling Architecture for IEEE 802.16 Wireless Metropolitan Area Network. 2007 IEEE/ACS International Conference on Computer Systems and Applications, 13-16 May 2007 2007. 203-210.

HAMITI, S. 2009. IEEE 802.16 m System Description Document (SDD). *IEEE 802.16 m-08/003r7*.

HAMZA, J. B., NG, C. K., NOORDIN, N., RASID, M. & ISMAIL, A. 2010. Review of minimizing a vertical handover in a heterogeneous wireless network. *IETE technical review,* 27**,** 97-106.

HAN, B., JIA, W. & LIN, L. 2007. Performance evaluation of scheduling in IEEE 802.16 based wireless mesh networks. *Computer Communications,* 30**,** 782-792.

HO, L. T., SAMUEL, L. G. & PITTS, J. M. 2003. Applying emergent self-organizing behavior for the coordination of 4G networks using complexity metrics. *Bell Labs Technical Journal,* 8**,** 5-25.

HOYMANN, C. 2005. Analysis and performance evaluation of the OFDM-based metropolitan area network IEEE 802.16. *Computer networks,* 49**,** 341-363.

HSIEH, I. P., CHANG, F. M. & KAO, S. J. 2012. A slot-based BS scheduling with maximum latency guarantee and capacity first in 802.16 e networks. *International Journal of Communication Systems,* 25**,** 55-66.

HUA WANG & LARS DITTMANN 2010. Downlink resource management for QoS scheduling in IEEE 802.16 WiMAX networks. *Computer Communications,* 33**,** 940-953.

HUANG, F., GENG, J., WEI, G., WANG, Y. & YANG, D. Performance analysis of distributed and centralized scheduling in two-hop relaying cellular system. 2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications, 2009. IEEE, 1337-1341.

IEEESTANDARD, G. 2018. IEEE Standard for Air Interface for Broadband Wireless Access Systems. *IEEE Std 802.16-2017 (Revision of IEEE Std 802.16-2012)***,** 1-2726.

IKBAL CHAMMAKHI MSADAA, FETHI FILALI & FAROUK KAMOUN. An Adaptive QoS Architecture for IEEE 802.16 Broadband Wireless Networks. 2007 IEEE International Conference on Mobile Adhoc and Sensor Systems, 8-11 Oct. 2007 2007. 1-3.

JERRY, B. 2005. *Discrete event system simulation*, Pearson Education India.

JIANFENG CHEN, WENHUA JIAO & HONGXI WANG. A service flow management strategy for IEEE 802.16 broadband wireless access systems in TDD mode. IEEE International Conference on Communications, 2005. ICC 2005. 2005, 16-20 May 2005 2005. 3422-3426 Vol. 5.

K. VINAY, N. SREENIVASULU, D. JAYARAM & D. DAS. Performance evaluation of end-to-end delay by hybrid scheduling algorithm for QoS in IEEE 802.16 network. 2006 IFIP International Conference on Wireless and Optical Communications Networks, 11-13 April 2006 2006. IEEE, 5 pp.-5.

KASSAB, M., BONNIN, J.-M. & MAHDI, M. WiMAX Simulation module with management architecture and signaling exchanges. Proceedings of the Fourth International ICST Conference on Performance Evaluation Methodologies and Tools, 2009. ICST (Institute for Computer Sciences, Social-Informatics and …, 6.

KATIDIOTIS, A., TSAGKARIS, K. & DEMESTICHAS, P. 2010. Performance evaluation of artificial neural network-based learning schemes for cognitive radio systems. *Computers & Electrical Engineering,* 36**,** 518-535.

KAUR, H. & SINGH, G. 2011. Performance Analysis of WiMAX Scheduling Algorithms. *International Journal of Advanced Research in Computer Science,* 2.

KIM, D. & GANZ, A. Fair and efficient multihop scheduling algorithm for IEEE 802.16 BWA systems. 2nd International Conference on Broadband Networks, 2005., 2005. IEEE, 833-839.

KIM, S., LEE, M. & YEOM, I. Simulating IEEE 802.16 Uplink Scheduler Using NS-2. Simulation Works. Industry Track to The First International Conference on Simulation Tools and Techniques for Communications, Networks and Systems, ICST, 2008. Citeseer.

KIM, S. & YEOM, I. 2007. TCP-aware uplink scheduling for IEEE 802.16. *IEEE Communications letters,* 11**,** 146-148.

KITTI WONGTHAVARAWAT & AURA GANZ. IEEE 802.16 based last mile broadband wireless military networks with quality of service support. IEEE Military Communications Conference, 2003. MILCOM 2003., 2003a. IEEE, 779-784.

KITTI WONGTHAVARAWAT & AURA GANZ 2003b. Packet scheduling for QoS support in IEEE 802.16 broadband wireless access systems. *international journal of communication systems 16,* 16**,** 81-96.

KUMAR, A. & GARG, A. K. 2011. Queuing Algorithem Based Quality of Service (Qos) For Scheduling Environment Model in Wimax Network with Opnet Modeler. *Global Journal of Researches in Engineering (GJRE),* 11**,** 35-40.

KWAK, R. & CIOFFI, J. M. Resource-allocation for OFDMA multi-hop relaying downlink systems. IEEE GLOBECOM 2007-IEEE Global Telecommunications Conference, 2007. IEEE, 3225-3229.

LAIAS, E. & AWAN, I. 2010. An interactive QoS framework for fixed WiMAX networks. *Simulation Modelling Practice and Theory,* 18**,** 291-303.

LAKKAKORPI, J., SAYENKO, A. & MOILANEN, J. Comparison of different scheduling algorithms for wimax base station: Deficit round-robin vs. proportional fair vs. weighted deficit round-robin. 2008 IEEE Wireless Communications and Networking Conference, 2008. IEEE, 1991-1996.

LAW, A. M. How to build valid and credible simulation models. Simulation Conference, 2008. WSC 2008. Winter, 2008. IEEE, 39-47.

LEE, J., YOON, H. & YEOM, I. 2010. Distributed fair scheduling for wireless mesh networks using IEEE 802.11. *IEEE transactions on vehicular technology,* 59**,** 4467-4475.

LIN, Y.-N., WU, C.-W., LIN, Y.-D. & LAI, Y.-C. A latency and modulation aware bandwidth allocation algorithm for WiMAX base stations. 2008 IEEE Wireless Communications and Networking Conference, 2008. IEEE, 1408-1413.

LIU, S., FENG, S., YE, W. & ZHUANG, H. 2009. Slot allocation algorithms in centralized scheduling scheme for IEEE 802.16 based wireless mesh networks. *Computer Communications,* 32**,** 943-953.

LIU, Y., CILIAX, B. J., BORGES, K., DASIGI, V., RAM, A., NAVATHE, S. B. & DINGLEDINE, R. Comparison of two schemes for automatic keyword extraction from MEDLINE for functional gene clustering. Proceedings. 2004 IEEE Computational Systems Bioinformatics Conference, 2004. CSB 2004., 2004. IEEE, 394-404.

MA, M., LU, J. & FU, C. 2010. Hierarchical scheduling framework for QoS service in WiMAX point-to-multi-point networks. *IET communications,* 4**,** 1073-1082.

MANOLIS KATEVENIS, STEFANOS SIDIROPOULOS & COSTAS COURCOUBETIS 1991. Weighted round-robin cell multiplexing in a general-purpose ATM switch chip. *IEEE Journal on selected Areas in Communications,* 9**,** 1265-1279.

MARDINI, W. & ALFOOL, M. A. 2011. Modified WRR Scheduling Algorithm for WiMAX Networks. *Network Protocols & Algorithms,* 3**,** 24-53.

MARKS, R. 2002. A technical overview of the WirelessMAN air interface for broadband wireless access. *IEEE C802***,** 16-02.

MARKS, R. IEEE standard 802.16 for global broadband wireless access. ITU Telecom World 2003 Forum, 2003.

MEHTA, V. & GUPTA, D. N. 2012. Performance analysis of qos parameters for wimax networks. *International Journal of Engineering and Innovative Technology (IJEIT) Volume,* 1**,** 105-110.

MICHEL, M. & PAUTET, M.-B. 1992. The GSM system for mobile communications. *Telecom Publishing*.

MSADAA, I. C., FILALI, F. & KAMOUN, F. An 802.16 model for NS2 simulator with an integrated QoS architecture. Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, 2008. ICST (Institute for Computer Sciences, Social-Informatics and …, 29.

MUKHERJEE, S. & VISWANATHAN, H. Analysis of throughput gains from relays in cellular networks. GLOBECOM'05. IEEE Global Telecommunications Conference, 2005., 2005. IEEE, 6 pp.-3476.

NGUYEN, T.-D. & HAN, Y. 2006. A proportional fairness algorithm with QoS provision in downlink OFDMA systems. *IEEE Communications Letters,* 10**,** 760-762.

NGUYEN, T. M., YIM, T., JEON, Y., KYUNG, Y. & PARK, J. 2014. QoS-aware dynamic resource allocation for wireless broadband access networks. *EURASIP Journal on Wireless Communications and Networking,* 2014**,** 104.

NIYATO, D. & HOSSAIN, E. 2005a. Call admission control for QoS provisioning in 4G wireless networks: issues and approaches. *IEEE network,* 19**,** 5-11.

NIYATO, D. & HOSSAIN, E. Queue-aware uplink bandwidth allocation for polling services in 802.16 broadband wireless networks. GLOBECOM'05. IEEE Global Telecommunications Conference, 2005., 2005b. IEEE, 5 pp.-3706.

NIYATO, D. & HOSSAIN, E. 2006. A Queuing-Theoretic and Optimization-Based Model for Radio Resource Management in IEEE 802.16 Broadband Wireless Networks. *IEEE Transactions on Computers,* 55**,** 1473-1488.

NUAYMI, L. 2007. *WiMAX: technology for broadband wireless access*, John Wiley & Sons.

PAREEK, D. 2006. *The Business of WiMAX*, John Wiley & Sons.

PAREKH, A. K. & GALLAGER, R. G. 1993. A generalized processor sharing approach to flow control in integrated services networks: the single-node case. *IEEE/ACM transactions on networking***,** 344-357.

PATEL, G. & DENNETT, S. 2000. The 3GPP and 3GPP2 movements toward an all-IP mobile network. *IEEE Personal Communications,* 7**,** 62-64.

PATEL, Z. & DALAL, U. 2014. Design and implementation of low latency weighted round Robin (LL-WRR) scheduling for high speed networks. *International Journal of Wireless & Mobile Networks,* 6**,** 59-71.

PENG, M., LI, M. & WANG, W. A unified architecture and the tree level based scheduling technique for multi-radio multi-channel in IEEE 802.16 standard based wireless mesh networks. 2006 First International Conference on Communications and Networking in China, 2006. IEEE, 1-6.

RATH, H. K., BHORKAR, A. & SHARMA, V. NXG02-4: An opportunistic uplink scheduling scheme to achieve bandwidth fairness and delay for multiclass traffic in Wi-Max (IEEE 802.16) broadband wireless networks. IEEE Globecom 2006, 2006. IEEE, 1-5.

REHMAN, A. & SHAKIR, M. 2014. *Comparative Analysis of Scheduling Algorithms in IEEE 802.16 WiMAX*, University of Bahrain: Scientefic Publishing Center.

SADRI, Y. & KHANMOHAMADI, S. 2013. An intelligent scheduling system using fuzzy logic controller for management of services in WiMAX networks. *The Journal of Supercomputing***,** 1-13.

SAFFER, Z., ANDREEV, S. & KOUCHERYAVY, Y. 2011. Performance evaluation of uplink delay-tolerant packet service in IEEE 802.16-based networks. *EURASIP Journal on Wireless Communications and Networking,* 2011**,** 549492.

SALAH, A. A., ALI, B. M., SAQER, A. & ODEH, N. An efficient resource allocation algorithm for OFDMA cooperative relay networks with fairness and QoS guaranteed. 2010 Second International Conference on Network Applications, Protocols and Services, 2010. IEEE, 188-192.

SARKAR, M. & SACHDEVA, H. A QoS aware packet scheduling scheme for WiMAX. Proceedings of IAENG Conference on World Congress on Engineering and Computer Science, Berkeley, 2009.

SAYENKO, A., ALANEN, O. & HÄMÄLÄINEN, T. 2008. Scheduling solution for the IEEE 802.16 base station. *Computer Networks,* 52**,** 96-115.

ŞEKERCIOĞLU, Y. A., IVANOVICH, M. & YEĞIN, A. 2009. A survey of MAC based QoS implementations for WiMAX networks. *Computer Networks*, 53**,** 2517-2536.

SEMERIA, C. 2001. Supporting differentiated service classes: queue scheduling disciplines. *Juniper networks***,** 11-14.

SHANG, Y. & CHENG, S. An enhanced packet scheduling algorithm for QoS support in IEEE 802.16 wireless network. International Conference on Networking and Mobile Computing, 2005. Springer, 652-661.

SHETIYA, H. & SHARMA, V. Algorithms for routing and centralized scheduling in IEEE 802.16 mesh networks. IEEE Wireless Communications and Networking Conference, 2006. WCNC 2006., 2006. IEEE, 147-152.

SHIMONISHI, H., YOSHIDA, M., FAN, R. & SUZUKI, H. An improvement of weighted round robin cell scheduling in ATM networks. GLOBECOM 97. IEEE Global Telecommunications Conference. Conference Record, 1997. IEEE, 1119-1123.

SHREEDHAR, M. & VARGHESE, G. 1996. Efficient fair queuing using deficit round-robin. *IEEE/ACM Transactions on networking***,** 375-385.

SIKORA, A. & GROZA, V. F. Coexistence of IEEE802. 15.4 with other Systems in the 2.4 GHz-ISM-Band. 2005 IEEE Instrumentationand Measurement Technology Conference Proceedings, 2005. IEEE, 1786-1791.

SINGLA, S. & KAMBOJ, M. S. Analysis of packet scheduling in WiMAX network. International Conference on Recent Advances and Future Trends in Information Technology, 2012.

SRINIVASAN, R. 2008. IEEE 802.16 m evaluation methodology document (EMD). *IEEE 802.16 m-08/004r3*.

STALLINGS, W. 2009. *Wireless communications and networks,* Pearson Education India.

SUN, J., YAO, Y. & ZHU, H. Quality of service scheduling for 802.16 broadband wireless access systems. Vehicular Technology Conference, 2006. VTC 2006-Spring. IEEE 63rd, 2006. IEEE, 1221-1225.

SUNDARESAN, K., WANG, X. & MADIHIAN, M. Low-overhead scheduling algorithms for OFDMA relay networks. Proceedings of the 4th Annual International Conference on Wireless Internet, 2008. ICST (Institute for Computer Sciences, Social-Informatics and …, 24.

TAO, Z., TEO, K. H. & ZHANG, J. Aggregation and concatenation in IEEE 802. 16j mobile multihop relay (MMR) networks. 2007 IEEE Mobile WiMAX Symposium, 2007. IEEE, 85-90.

TARHINI, C. & CHAHED, T. 2012. QoS-oriented resource allocation for streaming flows in IEEE802. 16e Mobile WiMAX. *Telecommunication Systems,* 51**,** 65-71.

TEO, K. H., TAO, Z. & ZHANG, J. 2007. The mobile broadband WiMAX standard [standards in a nutshell]. *IEEE Signal Processing Magazine,* 24**,** 144-148.

WANG, C., YAN, W.-J. & LO, H.-K. 2012. Dynamic admission control and bandwidth reservation for IEEE 802.16 e mobile WiMAX networks. *EURASIP Journal on Wireless Communications and Networking,* 2012**,** 143.

WANG, H. & JIA, W. Effective traffic control in IEEE 802.16 j WiMAX networks. 2010 IEEE 18th International Workshop on Quality of Service (IWQoS), 2010. IEEE, 1-5.

WEHRLE, K., GÜNES, M. & GROSS, J. 2010. *Modeling and tools for network simulation*, Springer Science & Business Media.

WEI, H.-Y., GANGULY, S., IZMAILOV, R. & HAAS, Z. J. Interference-aware IEEE 802.16 WiMax mesh networks. 2005 IEEE 61st Vehicular Technology Conference, 2005. IEEE, 3102-3106.

WEI NIE, HOUJUN WANG & JONG HYUK PARK 2011. Packet Scheduling with QoS and Fairness for Downlink Traffic in WiMAX Networks. *JIPS,* 7**,** 261-270.

WIMAX-PART, M. Mobile WiMAX – Part I: A Technical Overview and Performance Evaluation. WiMAX forum, 2006.

XIAO, Y. 2006. QoS guarantee and provisioning at the contention-based wireless MAC layer in the IEEE 802.11 e wireless LANs. *IEEE Wireless Communications,* 13**,** 14-21.

XIAOJING, M. 2007. *An efficient scheduling for diverse QoS requirements in WiMAX*. University of Waterloo.

YADAV, A. & CHAUHAN, I. V. 2016. Study of Different WiMAX QoS Scheduling Algorithms. *IJEAST*, 1**,** 162-168.

YOON, H.-H., KIM, H., OH, C. & KIM, K. A queue length-based scheduling scheme in ATM networks. Proceedings of IEEE. IEEE Region 10 Conference. TENCON 99.'Multimedia Technology for Asia-Pacific Information Infrastructure'(Cat. No. 99CH37030), 1999. IEEE, 234-237.

ZENG, H. & ZHU, C. Resource allocation in 802.16 j multi-hop relay systems with the user resource fairness constraint. 2009 IEEE Wireless Communications and Networking Conference, 2009. IEEE, 1-6.

ZHANG, S. & LAU, V. K. Distributed resource allocation for ofdma system with half-duplex relay using rateless code. 2007 IEEE International Symposium on Information Theory, 2007. IEEE, 1921-1925.

Appendix

This code for connection manager

```
#include "connectionmanager.h"

#include "mac802_16.h"


/**

 * Create the manager for the given mac

 * @param mac The Mac where the manager is located

 */

ConnectionManager::ConnectionManager (Mac802_16 * mac)

{

  assert (mac!=NULL);

  mac_ = mac;


  //init list

  LIST_INIT (&i_con_list_);

  LIST_INIT (&o_con_list_);

}




/**

 * Add a connection to the list
```

```
 * @param con The connection to add
 * @param out true if it is an outgoing connection
 */
void ConnectionManager::add_connection (Connection* con, bool out) {
  assert (con!=NULL);
  assert (!get_connection (con->get_cid(), out)); //check duplicate
  mac_->debug ("At %f in %d adding %s connection %d\n", \
		 NOW, mac_->addr(), out?"out":"in", con->get_cid());
  if (out)
    con->insert_entry (&o_con_list_);
  else
    con->insert_entry (&i_con_list_);

  con->setManager(this);
}

/**
 * Remove a connection
 * @param The connection to remove
 */
void ConnectionManager::remove_connection (Connection* con) {
  assert (con !=NULL);
  mac_->debug ("At %f in %d removing connection %d\n", \
		 NOW, mac_->addr(), con->get_cid());
```

```
  con->remove_entry ();
}


/**
 * Remove connection by CID, both directions.
 * @param cid The connection id
 */
void ConnectionManager::remove_connection (int cid)
{
  Connection *con = get_connection (cid, IN_CONNECTION);
  if (con)
    remove_connection (con);
  con = get_connection (cid, OUT_CONNECTION);
  if (con)
    remove_connection (con);
}
  /**
 * Return the connection that has the given CID
 * @param cid The connection ID
 * @param out The direction
 * @return the connection or NULL
 */
Connection* ConnectionManager::get_connection (int cid, bool out) {
  //search throught the list
```

```
  for (Connection *n=out?o_con_list_.lh_first:i_con_list_.lh_first;
       n; n=n->next_entry()) {
    if (n->get_cid ()==cid)
      return n;
  }
  return NULL;
}


/**
 * Flush the queues. This can be called after switching BS.
 */
void ConnectionManager::flush_queues () {
  mac_->debug ("At %f in %d Flushing queues\n", NOW, mac_->addr());
  for (Connection *n=o_con_list_.lh_first; n; n=n->next_entry()) {
    int i = n->flush_queue();
    mac_->debug ("\tFreed %d packet in queue for connection %d\n", i, n->get_cid());
  }
}
```

This code for WiMAX MAC

```
#include "mac802_16pkt.h"


/**
 * Return the size of the MOB_NBR-ADV frame
```

```
 * @param frame The frame
 */
int     Mac802_16pkt::getMOB_NBR_ADV_size(mac802_16_mob_nbr_adv_frame
*frame)
{
  int size = 4; //min size
  if ((frame->skip_opt_field & 0x1) == 0)
    size +=3; //add operator id

  for (int i = 0 ; i < frame->n_neighbors ; i++) {
    size += 4; //min size for neighbor info
    if (frame->nbr_info[i].phy_profile_id.FAindex & 0x1)
      size++;
    if (frame->nbr_info[i].phy_profile_id.bs_eirp & 0x1)
      size++;
    if ((frame->skip_opt_field & 0x2) == 0)
      size +=3; //contain neighbor bs id
    if ((frame->skip_opt_field & 0x4) == 0)
      size ++; //contain HO process optimization
    if ((frame->skip_opt_field & 0x8) == 0)
      size ++; //contain neighbor bs id
    if (frame->nbr_info[i].dcd_included)
      size += 2+GET_DCD_SIZE (frame->nbr_info[i].dcd_settings.nb_prof);
    if (frame->nbr_info[i].ucd_included)
```

```
      size += 2+GET_UCD_SIZE (frame->nbr_info[i].ucd_settings.nb_prof);
    if (frame->nbr_info[i].phy_included)
      size += 2;
  }

  return size;
}
/**
 * Return the size of the MOB_SCN-REQ
 * @param frame The frame
 */
int Mac802_16pkt::getMOB_SCN_REQ_size(mac802_16_mob_scn_req_frame
*frame)
{
  int size=6;

  if (frame->n_recommended_bs_index != 0)
    size ++;
  int tmp = 11*(frame->n_recommended_bs_index+frame-
>n_recommended_bs_full);
  size += tmp/8;
  if ((tmp%8)!=0)
    size ++;
```

```
  return size;
}

/**
 * Return the size of the MOB_SCN-RSP
 * @param frame The frame
 */
int        Mac802_16pkt::getMOB_SCN_RSP_size(mac802_16_mob_scn_rsp_frame
*frame)
{
  int size=6;

  if (frame->scan_duration!=0) {
    size += 3;
    if (frame->n_recommended_bs_index!=0)
      size ++;
    int tmp = 0;
    for (int i = 0 ; i < frame->n_recommended_bs_index ; i++) {
      tmp+=11;
      if (frame->rec_bs_index[i].scanning_type==2 ||
              frame->rec_bs_index[i].scanning_type==3)
            tmp+=24;
    }
    for (int i = 0 ; i < frame->n_recommended_bs_index ; i++) {
```

```
      tmp+=51;
      if (frame->rec_bs_index[i].scanning_type==2 ||
              frame->rec_bs_index[i].scanning_type==3)
            tmp+=24;
    }
    size += tmp/8;
    if ((tmp%8)!=0)
      size ++;
  }

  return size;
}

/**
 * Return the size of the MOB_MSHO-REQ
 * @param frame The frame
 */
int
Mac802_16pkt::getMOB_MSHO_REQ_size(mac802_16_mob_msho_req_frame
*frame)
{
  int size=4;
  int tmp, tmpB;
```

```
tmp = 0;

tmpB = 0;

if (frame->n_new_bs_index !=0)

  size++;


if (frame->report_metric & 0x1) tmp++;

if (frame->report_metric & 0x2) tmp++;

if (frame->report_metric & 0x4) tmp++;


for (int i = 0 ; i < frame->n_new_bs_index ; i++) {

  tmpB += 20 + 8*tmp;

  if (frame->bs_index[i].arrival_time_diff_ind & 0x1)

    tmpB += 4;

}

//n_new_bs_full

for (int i = 0 ; i < frame->n_new_bs_full ; i++) {

  tmpB += 20 + 8*tmp;

  if (frame->bs_full[i].arrival_time_diff_ind & 0x1)

    tmpB += 4;

}

tmpB += 4;

//N_current

if (frame->report_metric & 0x8) tmp++;

for (int i = 0 ; i < frame->n_current_bs ; i++) {
```

```
    tmpB += 4 + 8*tmp;
  }
  //increase size
  size += tmp/8;
  if ((tmp%8)!=0)
    size ++;  //includes padding

  return size;
}
/**
 * Return the size of the MOB_MSHO-REQ
 * @param frame The frame
 */
int   Mac802_16pkt::getMOB_BSHO_RSP_size(mac802_16_mob_bsho_rsp_frame
*frame)
{
  int size=4;

  return size;
}

/**
 * Return the size of the MOB_HO-IND
 * @param frame The frame
```

```
 */
int           Mac802_16pkt::getMOB_HO_IND_size(mac802_16_mob_ho_ind_frame
*frame)
{
  int size=4;

  return size;
}
```

This code for WiMAX classifier

```
#include "sduclassifier.h"

/*
 * Create a classifier in the given mac
 */
SDUClassifier::SDUClassifier ()
{
  //set default priority
  priority_ = 0;
}

/*
```

```
 * Interface with the TCL script
 * @param argc The number of parameter
 * @param argv The list of parameters
 */
int SDUClassifier::command(int argc, const char*const* argv)
{
  if (argc == 3) {
    if (strcmp(argv[1], "set-priority") == 0) {
      priority_ = atoi(argv[2]);
      return TCL_OK;
    }
  }
  return TCL_ERROR;
}

/**
 * Classify a packet and return the CID to use (or -1 if unknown)
 * @param p The packet to classify
 * @return The CID or -1
 */
int SDUClassifier::classify (Packet * p) {
  return -1;
}
```

/* This WiMAX code contains the control agent located in IEEE 802.16 BS responsible for synchronization between BSs.

***/**

```
#include "wimaxctrlagent.h"

#include "mac802_16BS.h"

#include "bsscheduler.h"

#include "wimaxneighborentry.h"


#define MYNUM       Address::instance().print_nodeaddr(addr())



int hdr_wimaxbs::offset_;

/**

 * Tcl hook for Packet definitions

 */

static class WimaxBSHeaderClass : public PacketHeaderClass {

public:

          WimaxBSHeaderClass() : PacketHeaderClass("PacketHeader/WIMAXBS",

                                                   sizeof(hdr_wimaxbs)) {

            bind_offset(&hdr_wimaxbs::offset_);

      }

} class_wimaxbshdr;
```

```
/**
 * Tcl hook for agent
 */
static class WimaxCtrlAgentClass : public TclClass {
public:
	WimaxCtrlAgentClass() : TclClass("Agent/WimaxCtrl") {}
	TclObject* create(int, const char*const*) {
		return (new WimaxCtrlAgent());
	}
} class_wimaxctrlagent;

/*
 * Handler for timer expiration
 */
void UpdateTimer::expire (Event*)
{
  a_->sendUpdate();
}

/*
 * Handler for response timer expiration
 */
void ScanRspTimer::expire (Event*)
```

```
{
  a_->agent()->send_scan_response(a_->cid());
}



/*
 * Creates a Wimax controler agent
 * Initializes the agent and bind variable to be accessible in TCL
 */
WimaxCtrlAgent::WimaxCtrlAgent()    :    Agent(PT_WIMAXBS),    mac_(0),
updatetimer_ (this)
{
  nbmapentry_=0;
  LIST_INIT (&scan_req_head_);

  //bind attributes
  bind ("adv_interval_", &adv_interval_);
  bind ("default_association_level_", &defaultlevel_);
  bind ("synch_frame_delay_", &synch_frame_delay_);

  //schedule first update
  updatetimer_.sched (Random::uniform(0, UPDATE_JITTER));
}
```

```
/*
 * Interface with TCL interpreter
 * @param argc The number of elements in argv
 * @param argv The list of arguments
 * @return TCL_OK if everything went well else TCL_ERROR
 */
int WimaxCtrlAgent::command(int argc, const char*const* argv)
{
  //Tcl& tcl= Tcl::instance();

  if (argc == 3) {
    // set the Minimum interval between two RAs
    if (strcmp(argv[1], "set-mac") == 0) {
      mac_ = (Mac802_16BS*) TclObject::lookup(argv[2]);
      mac_->setCtrlAgent (this);
      return TCL_OK;
    }
  } else if (argc == 4) {
    if (strcmp(argv[1], "add-neighbor") == 0) {
      //the parameter is the mac, and we also extract the node
      Mac802_16 *tmp = (Mac802_16 *) TclObject::lookup(argv[2]);
      if (nbmapentry_ == MAX_MAP_ENTRY) {
          fprintf (stderr, "Table size exceeding. Increase MAX_MAP_ENTRY\n");
      }
```

```
      // The following is not supported outside of our mobility
      // package so we provide a workaround
      //tcl.evalf ("%s get-node", argv[2]);
      //Node *tmpNode = (Node *) TclObject::lookup(tcl.result());
      Node *tmpNode = (Node *) TclObject::lookup(argv[3]);

      //add entry
      maptable_[nbmapentry_][0] = tmp->addr();
      maptable_[nbmapentry_][1] = tmpNode->address();
      debug ("Adding neighbor %s (mac %d) in %s\n",
Address::instance().print_nodeaddr(tmpNode->address()),
                  tmp->addr(), MYNUM);
      nbmapentry_++;
      return TCL_OK;
    }
  }

  return (Agent::command(argc, argv));
}

/*
 * Send an update (DCD/UCD) to all neighboring BSs
 */
void WimaxCtrlAgent::sendUpdate ()
```

```
{
  //get the DCD/UCD message to include in the update
  Packet *dcd = mac_->getMap()->getDCD();
  Packet *ucd = mac_->getMap()->getUCD();

  //allocate data to store information
  mac802_16_dcd_frame *dcdframe = (mac802_16_dcd_frame*) dcd->accessdata();
  mac802_16_ucd_frame *ucdframe = (mac802_16_ucd_frame*) ucd->accessdata();

  Packet *p = allocpkt();
  hdr_ip *iph = HDR_IP(p);
  hdr_wimaxbs *rh = HDR_WIMAXBS(p);
  hdr_cmn *hdrc = HDR_CMN(p);

  rh->getType() = WIMAX_BS_ADV;
  hdrc->size() = HDR_CMN(dcd)->size()+HDR_CMN(ucd)->size(); //TBD: remove
double header
  //set content
  rh->macAddr() = mac_->addr();
  p->allocdata (sizeof (mac802_16_dcd_frame)+sizeof (mac802_16_ucd_frame));
  unsigned char *data = p->accessdata();
  memcpy (data, dcdframe, sizeof (mac802_16_dcd_frame));
  memcpy (data+sizeof (mac802_16_dcd_frame), ucdframe, sizeof
(mac802_16_ucd_frame));
```

```
  Packet *tmpPkt;
  for (int i = 0; i < nbmapentry_ ; i++) {
    tmpPkt = p->copy();
    iph = HDR_IP(tmpPkt);
    //set packet destination
    iph->daddr() = maptable_[i][1];
    iph->dport() = port();
    debug ("At %f in node %s, send update to node %s\n", NOW,
MYNUM,Address::instance().print_nodeaddr(iph->daddr()));
    debug ("frame number=%d\n", dcdframe->frame_number);
    send(tmpPkt, 0);
  }

  //reschedule timer
  updatetimer_.resched (adv_interval_);
}

/*
 * Process received packet
 * @param p The packet received
 * @param h The handler that sent the packet
 */
void WimaxCtrlAgent::recv(Packet* p, Handler *h)
```

```
{
  assert (p);


  hdr_wimaxbs *rh = HDR_WIMAXBS(p);
  switch (rh->getType()) {
  case WIMAX_BS_ADV:
    processUpdate (p);
    break;
  case WIMAX_BS_SYNCH_REQ:
    process_synch_request (p);
    break;
  case WIMAX_BS_SYNCH_RSP:
    process_synch_response (p);
    break;
  default:
    fprintf (stderr, "Unknown message type in WimaxCtrlAgent\n");
  }
  Packet::free (p);
}

/*
 * Process received packet
 * @param p The update received
 * @param h The handler that sent the packet
```

```
 */
void WimaxCtrlAgent::processUpdate(Packet* p)
{
  debug ("At %f in node %s, WimaxCtrlAgent received update message from %s\n",
NOW, MYNUM,
            Address::instance().print_nodeaddr(HDR_IP(p)->saddr()));

  hdr_wimaxbs *rh = HDR_WIMAXBS(p);
  WimaxNeighborEntry *entry = mac_->nbr_db_->getNeighbor(rh->macAddr());

  //check if we know about this neighbor
  bool found = false;
  for (int i = 0; i < nbmapentry_ ; i++) {
    if (maptable_[i][1]==HDR_IP(p)->saddr())
      found = true;
  }
  assert (found);

  if (entry==NULL) {
    debug ("\tNew neighbor detected...add entry for mac %d\n", rh->macAddr());
    entry = new WimaxNeighborEntry (rh->macAddr());
    mac_->nbr_db_->addNeighbor(entry);
  }
  //update entry
```

```
  unsigned char *data = p->accessdata();
  mac802_16_dcd_frame *dcdframe = (mac802_16_dcd_frame *)malloc (sizeof (mac802_16_dcd_frame));
  mac802_16_ucd_frame *ucdframe = (mac802_16_ucd_frame *)malloc (sizeof (mac802_16_ucd_frame));

  memcpy (dcdframe, data, sizeof (mac802_16_dcd_frame));
  memcpy (ucdframe, data+sizeof (mac802_16_dcd_frame), sizeof (mac802_16_ucd_frame));
  debug ("\tframe number=%d ccc=%d\n", dcdframe->frame_number,ucdframe->config_change_count);
  mac802_16_dcd_frame *dcdtmp = entry->getDCD();
  if (dcdtmp)
    free(dcdtmp);
  mac802_16_ucd_frame *ucdtmp = entry->getUCD();
  if (ucdtmp)
    free(ucdtmp);
  entry->setDCD (dcdframe);
  entry->setUCD (ucdframe);
  //free (p);
}

/**
 * Process scanning request
```

```
 * @param p The request
 */
void WimaxCtrlAgent::process_scan_request (Packet *req)
{
  hdr_mac802_16 *wimaxHdr_req = HDR_MAC802_16(req);
  gen_mac_header_t header_req = wimaxHdr_req->header;
  mac802_16_mob_scn_req_frame *req_frame;
  req_frame = (mac802_16_mob_scn_req_frame*) req->accessdata();

  mac_->debug ("At %f in Mac %d received scanning request from %d\n", NOW, mac_->addr(), header_req.cid);

  //for first implementation we disregard the information
  //sent by MN. Just use default association mechanisms

  //should check if there is already pending request: TBD
  Scan_req *entry = NULL;

  for (entry = scan_req_head_.lh_first ; entry && (entry->cid() != header_req.cid); entry=entry->next_entry());

  if (entry) {
    mac_->debug ("\tDuplicate requests. Skip...\n");
    return;
```

```
 } else {
  entry = new Scan_req (this, synch_frame_delay_*mac_->getFrameDuration(),
header_req.cid, req_frame);
  entry->insert_entry (&scan_req_head_);
  entry->start_frame()=mac_->getFrameNumber();
 }

 switch (defaultlevel_){
 case 0: //Scan without association
  entry->response()->scan_duration = req_frame->scan_duration;
  entry->response()->start_frame = 2;
  entry->response()->report_mode = 0; //no report for now
  entry->response()->interleaving_interval = req_frame->interleaving_interval;
  entry->response()->scan_iteration = req_frame->scan_iteration;
  entry->response()->n_recommended_bs_index = 0;
  entry->response()->n_recommended_bs_full = nbmapentry_;
  for (int i = 0; i < nbmapentry_ ; i++) {
   entry->response()->rec_bs_full[i].recommended_bs_id = maptable_[i][0];
   entry->response()->rec_bs_full[i].scanning_type =
SCAN_WITHOUT_ASSOC;
  }
  //send response
  mac_->send_scan_response (entry->response(), header_req.cid);
  //clean data
```

```
  entry->remove_entry();
  delete entry;
  break;

case 1: //Association without coordination
  entry->response()->scan_duration = req_frame->scan_duration;
  entry->response()->start_frame = 2;
  entry->response()->report_mode = 0; //no report for now
  entry->response()->interleaving_interval = req_frame->interleaving_interval;
  entry->response()->scan_iteration = req_frame->scan_iteration;
  entry->response()->n_recommended_bs_index = 0;
  entry->response()->n_recommended_bs_full = nbmapentry_;
  for (int i = 0; i < nbmapentry_ ; i++) {
    entry->response()->rec_bs_full[i].recommended_bs_id = maptable_[i][0];
    entry->response()->rec_bs_full[i].scanning_type = SCAN_ASSOC_LVL0;
  }
  //send response
  mac_->send_scan_response (entry->response(), header_req.cid);
  //clean data
  entry->remove_entry();
  delete entry;
  break;

case 2: //Association with coordination
```

```
 //init data
 entry->response()->n_recommended_bs_index = 0;
 entry->response()->n_recommended_bs_full = 0;
 entry->pending_rsp () = 0;

 //send request to neighbors
 for (int i = 0; i < nbmapentry_ ; i++) {
   Packet *p = allocpkt();
   hdr_ip *iph = HDR_IP(p);
   hdr_wimaxbs *rh = HDR_WIMAXBS(p);
   hdr_cmn *hdrc = HDR_CMN(p);

   rh->getType() = WIMAX_BS_SYNCH_REQ;
   hdrc->size() = 30; //We need to define proper size
   //set content
   iph = HDR_IP(p);
   iph->daddr() = maptable_[i][1];
   iph->dport() = port();

   rh->macAddr() = mac_->getCManager()->get_connection(header_req.cid, true)-
>getPeerNode()->getAddr();
   rh->cid = header_req.cid;
   rh->scanning_type = (wimax_scanning_type) defaultlevel_;
   //we suggest a rendez-vous time at the beginning of each
```

```
    //scan iteration
    rh->current_frame = mac_->getFrameNumber();
    //if we want to start scanning 2 frames after, then add one (i.e 3) because
    //the message will be sent on the next frame
    printf ("scan_duration=%d, scan_interval=%d\n", req_frame->scan_duration,
            req_frame->interleaving_interval);
    rh->rdvt          =          (i+1)*(req_frame->scan_duration+req_frame-
>interleaving_interval)+synch_frame_delay_+START_FRAME_OFFSET+1;
    rh->rendezvous_time = NOW+rh->rdvt*mac_->getFrameDuration();
    printf ("Request: current frame=%d, rdv frame=%d, rdv time=%f\n",
            rh->current_frame, rh->rdvt, rh->rendezvous_time);

    entry->pending_rsp ()++;
    send (p,0);

    entry->response()->n_recommended_bs_full = i + 1;
    entry->response()->rec_bs_full[i].recommended_bs_id = maptable_[i][0];
    entry->response()->rec_bs_full[i].scanning_type = rh->scanning_type;
    entry->response()->rec_bs_full[i].rdv_time = rh->rdvt-synch_frame_delay_;
  }

  //continue initializing response
  entry->response()->scan_duration = req_frame->scan_duration;
  entry->response()->start_frame = START_FRAME_OFFSET;
```

```
    entry->response()->report_mode = 0; //no report for now
    entry->response()->interleaving_interval = entry->request()->interleaving_interval;
    entry->response()->scan_iteration = entry->request()->scan_iteration;
    entry->response()->n_recommended_bs_index = 0;

    //printf ("Response: current frame=%d (now=%f), start in %d frame (t=%f)\n",
    //      mac_->getFrameNumber (), NOW, entry->response()->start_frame, NOW+(entry->response()->start_frame*mac_->getFrameDuration()));
    //send response
    //((BSScheduler*)mac_->getScheduler())->send_scan_response (entry->response(), entry->cid());

    break;
  case 3: //Network Assisted Association reporting

    break;
  default:
    break;
  }
}

/**
 * Process synchronization request
```

```
 * @param req The request
 */
void WimaxCtrlAgent::process_synch_request (Packet *req)
{
  debug ("At %f in node %s, WimaxCtrlAgent received synch request from %s\n",
NOW, MYNUM,
	 Address::instance().print_nodeaddr(HDR_IP(req)->saddr()));

  //schedule rendez-vous time
  //schedule sending of Fast-ranging-IE
  mac_->addNewFastRanging (HDR_WIMAXBS(req)->rendezvous_time,
HDR_WIMAXBS(req)->macAddr());



  //send response
  Packet *p = allocpkt();
  hdr_ip *iph = HDR_IP(p);
  hdr_wimaxbs *rh = HDR_WIMAXBS(p);
  hdr_cmn *hdrc = HDR_CMN(p);

  rh->getType() = WIMAX_BS_SYNCH_RSP;
  hdrc->size() = 30; //We need to define proper size
  //set content
  iph = HDR_IP(p);
```

```
  iph->daddr() = HDR_IP(req)->saddr();
  iph->dport() = port();

  //we accept what the serving BS sent
  rh->cid = HDR_WIMAXBS(req)->cid;
  rh->scanning_type = HDR_WIMAXBS(req)->scanning_type;
  rh->current_frame = HDR_WIMAXBS(req)->current_frame;
  rh->rdvt = HDR_WIMAXBS(req)->rdvt;
  rh->rendezvous_time = HDR_WIMAXBS(req)->rendezvous_time;
  //send (p,0);
}

/**
 * Process synchronization response
 * @param p The response
 */
void WimaxCtrlAgent::process_synch_response (Packet *p)
{
  debug ("At %f in node %s, WimaxCtrlAgent received synch response from %s\n",
NOW, MYNUM,
	 Address::instance().print_nodeaddr(HDR_IP(p)->saddr()));

  hdr_wimaxbs *rh = HDR_WIMAXBS(p);
  int i;
```

```
  //update information
  Scan_req *entry;
  for (entry = scan_req_head_.lh_first ; entry && (entry->cid() != rh->cid);
entry=entry->next_entry());

  assert (entry);

  i = entry->response()->n_recommended_bs_full;
  entry->response()->n_recommended_bs_full = i + 1;
  entry->response()->rec_bs_full[i].recommended_bs_id = maptable_[i][0];
  entry->response()->rec_bs_full[i].scanning_type = rh->scanning_type;
  entry->response()->rec_bs_full[i].rdv_time = rh->rdvt;
}

/**
 * Send a scan response to the MN that has the given CID
 * @param cid The CID of the MN
 */
void WimaxCtrlAgent::send_scan_response (int cid)
{
  Scan_req *entry;
  for (entry = scan_req_head_.lh_first ; entry && (entry->cid() != cid); entry=entry-
>next_entry());
```

```
  assert (entry);


  entry->response()->scan_duration = entry->request()->scan_duration;
  printf ("Response: current frame=%d, start frame=%d diff=%d\n",
        mac_->getFrameNumber(), entry->start_frame(), 100-(mac_-
>getFrameNumber()-entry->start_frame()));
  printf ("Response: current frame=%d (now=%f), start in %d frame (t=%f)\n",
        mac_->getFrameNumber (), NOW, entry->response()->start_frame,
NOW+(entry->response()->start_frame*mac_->getFrameDuration()));
  //send response
  mac_->send_scan_response (entry->response(), entry->cid());
  //clean data
  entry->remove_entry();
  delete entry;
}
```

This code for Ul sub frame timer

```
#include "ulsubframetimer.h"

#include "framemap.h"

#include "subframe.h"

#include "wimaxscheduler.h"

#include "contentionslot.h"


/**

 * Creates a timer to handle the subframe transmission

 * @param subframe The UlSubframe

 */

UlSubFrameTimer::UlSubFrameTimer (UlSubFrame *subframe): pdu_(0),
newphy_(true), mac_(0)

{

  assert (subframe);

  subframe_ = subframe;

}


/**

 * Reset the timer

 */

void UlSubFrameTimer::reset ()
```

```
{
  pdu_= NULL;
  newphy_ = true;
  if (status()==TIMER_PENDING)
    cancel();
}

/**
 * When it expires, the timer will handle the next packet to send
 * @param e not used
 */
void UlSubFrameTimer::expire( Event* e )
{
  if (!mac_) {
    mac_= subframe_->map_->getMac();
  }

  //mac_->debug ("At %f in Mac %d UlsubFrameTimer expires\n", NOW, mac_-
>addr());
  int iuc;
  if (newphy_) {
    if (pdu_==NULL){
      //printf ("\ttake first pdu\n");
      //get the first pdu
```

```
  pdu_ = subframe_->getFirstPdu();
  if (!pdu_)
        return; //this means there was no uplink burst allocated
} else {
  //printf ("\tcontinue pdu\n");
  iuc = pdu_->getBurst(0)->getIUC();
  //check if this is a contention slot
  if (iuc == UIUC_INITIAL_RANGING) {
        //stop ranging timer
        //printf ("\tpause ranging\n");
        subframe_->ranging_->pauseTimers ();
  } else if (iuc == UIUC_REQ_REGION_FULL) {
        //stop bw request timers
        //printf ("\tpause bw requests\n");
        subframe_->bw_req_->pauseTimers();
  }
  pdu_ = pdu_->next_entry();
}

if (pdu_->getBurst(0)->getIUC()==UIUC_END_OF_MAP){
  pdu_=NULL; //reset for next frame
  //mac_->debug ("\tend of map\n");
  if (mac_->getNodeType()==STA_BS) {
        mac_->getPhy()->setMode (OFDM_SEND);
```

```
    } else {
          mac_->getPhy()->setMode (OFDM_RECV);
    }
    return; //end of subframe
  }

  //change the modulation
  UlBurst *burst =  (UlBurst*)pdu_->getBurst(0);
  iuc = burst->getIUC();
  //printf ("Searching for IUC=%d\n", iuc);
  if      (iuc      ==      UIUC_EXT_UIUC      &&      burst->getExtendedUIUC()==
UIUC_FAST_RANGING) {
    iuc = burst->getFastRangingUIUC ();
    //printf ("Searching for IUC=%d\n", iuc);
  }
  Ofdm_mod_rate rate = subframe_->getProfile (iuc)->getEncoding();
  mac_->getPhy()->setModulation (rate);
  //check if this is a contention slot
  if (iuc == UIUC_INITIAL_RANGING) {
    //resume ranging timer
    //printf ("\tresume ranging\n");
    subframe_->ranging_->resumeTimers();
  } else if (iuc == UIUC_REQ_REGION_FULL) {
    //resume bw request timers
```

```
    //printf ("\tresume bw requests\n");
    subframe_->bw_req_->resumeTimers();
  }
}
//check if packet to send
Packet *p = pdu_->getBurst(0)->dequeue();

if (p) {
  newphy_= false;
  double txtime = HDR_CMN(p)->txtime();
  //printf ("\tPacket to send\n");
  //schedule for next packet
  mac_->transmit (p);
  if (pdu_->getBurst(0)->getQueueLength()!=0) {
    //mac_->debug ("\treschedule in %f (%f)\n", txtime, NOW+txtime);
    resched (txtime); //wait transmition time + GAP
    return;
  }
}

newphy_= true;
double stime=0.0;
assert (pdu_->next_entry());
```

```
  stime = subframe_->map_->getStarttime();
  //mac_->debug("\tstart frame=%f\n", stime);
  stime += subframe_->getStarttime()*mac_->getPhy()->getPS();
  //mac_->debug ("\tulstart = %f\n", stime);
  stime += pdu_->next_entry()->getBurst(0)->getStarttime()*mac_->getPhy()->getSymbolTime();
  //mac_->debug ("\tnext pdu start=%d\n", pdu_->next_entry()->getBurst(0)->getStarttime());
  //mac_->debug ("\t%f Next burst %d at %f\n", NOW, pdu_->next_entry()->getBurst(0)->getIUC(), stime);
  resched (stime-NOW);

}
```

This code for Profile code:

```
#include "profile.h"
#include "subframe.h"

/**
 * Creates a profile with the given frequency and encoding
 * @param f The frequency information for the profile
 * @param enc The encoding type
```

```
 */
Profile::Profile (SubFrame *subframe, int f, Ofdm_mod_rate enc) : iuc_(0)
{
  assert (subframe);
  subframe_ = subframe;
  frequency_ = f;
  encoding_ = enc;
}

/**
 * Return the encoding type
 * @return the encoding type
 */
Ofdm_mod_rate Profile::getEncoding( )
{
  return encoding_;
}
/**
 * Set the encoding type
 * @param enc the encoding type
 */
void Profile::setEncoding( Ofdm_mod_rate enc )
{
  if (encoding_ != enc)
```

```
    subframe_->incrCCC();
  encoding_ = enc;
}

/**
 * Return the frequency in unit of kHz
 * @return the frequency
 */
int Profile::getFrequency( )
{
  return frequency_;
}

/**
 * Set the frequency in unit of kHz
 * @param f the frequency
 */
void Profile::setFrequency( int f )
{
  if (frequency_ != f)
    subframe_->incrCCC();
  frequency_ = f;
}
/**
```

```
 * Return the frequency in unit of kHz
 * @return the frequency
 */
int Profile::getIUC( )
{
  return iuc_;
}

/**
 * Set the IUC number for this profile
 * @param iuc The IUC number for this profile
 */
void Profile::setIUC( int iuc )
{
  if (iuc_!=0 && iuc_!= iuc)
    subframe_->incrCCC();
  iuc_ = iuc;
}
```

This code for DL sub frame timer

```
#include "dlsubframetimer.h"
#include "framemap.h"
#include "subframe.h"
```

```
#include "wimaxscheduler.h"
#include "contentionslot.h"

/**
 * Creates a timer to handle the subframe transmission
 * @param subframe The DlSubframe
 */
DlSubFrameTimer::DlSubFrameTimer (DlSubFrame *subframe): burstIndex_(0),
newburst_(true), mac_(0)
{
  assert (subframe);
  subframe_ = subframe;
}

/**
 * Reset the timer
 */
void DlSubFrameTimer::reset ()
{
  burstIndex_ = 0;
  newburst_ = true;
  if (status()==TIMER_PENDING)
    cancel();
}
```

```
/**
 * When it expires, the timer will handle the next packet to send
 * @param e not used
 */
void DlSubFrameTimer::expire( Event* e )
{
  if (!mac_) {
    mac_= subframe_->map_->getMac();
  }

  //printf ("At %f in Mac %d DlsubFrameTimer expires\n", NOW, mac_->addr());
  int iuc;
  Burst *b = subframe_->getPdu()->getBurst(burstIndex_);
  if (newburst_) {
    //printf ("\tburst=%x type=%d\n", b,b->getIUC());
    if (b->getIUC()==DIUC_END_OF_MAP) {
      //printf ("\tend of subframe\n");
      burstIndex_=0;//reset for next frame
      if (mac_->getNodeType()==STA_MN) {
            mac_->getPhy()->setMode (OFDM_SEND);
      } else {
            mac_->getPhy()->setMode (OFDM_RECV);
      }
```

```
    return; //end of subframe
  }
  //change modulation
  iuc = b->getIUC();
  Ofdm_mod_rate rate = subframe_->getProfile (iuc)->getEncoding();
  mac_->getPhy()->setModulation (rate);
}
//check if packet to send
Packet *p = b->dequeue();
if (p) {
  newburst_ = false;
  double txtime = HDR_CMN(p)->txtime();
  //schedule for next packet
  mac_->transmit (p);
  if (b->getQueueLength()!=0) {
    //printf ("\tNext packet at %f(in %f)\n", NOW+txtime, txtime);
    resched (txtime); //wait transmition time + GAP
    return;
  }
}

//no packet to send...schedule for next phypdu
newburst_= true;
burstIndex_++;
```

```
  double stime=0.0;
  assert (b->next_entry());

  stime = subframe_->map_->getStarttime();
  stime += b->next_entry()->getStarttime()*mac_->getPhy()->getSymbolTime();
  //printf ("\tMap start time=%f Next burst at %f\n", subframe_->map_-
>getStarttime(), stime);
  resched (stime-NOW);

}
```

This code Connection:

```
//#include "contentionrequest.h"
#include "contentionslot.h"
#include "framemap.h"
#include "wimaxscheduler.h"
#include "random.h"
#include "mac802_16SS.h"

/*
 * Handling function for WimaxFrameTimer
 * @param e The event that occured
 */
```

```
void WimaxBackoffTimer::handle(Event *e)
{
  busy_ = 0;
  paused_ = 0;
  stime = 0.0;
  rtime = 0.0;

  HDR_CMN(c_->p_)->timestamp() = NOW; //add timestamp since it bypasses the queue
  mac->transmit (c_->p_->copy());

  //start timeout trigger
  c_->starttimeout ();
}

void WimaxBackoffTimer::pause()
{
	Scheduler &s = Scheduler::instance();

	//the caculation below make validation pass for linux though it
	// looks dummy

	double st = s.clock();
	double rt = stime;
```

```
          double sr = st - rt;


          assert(busy_ && ! paused_);


          paused_ = 1;

          rtime -= sr;


          assert(rtime >= 0.0);


          s.cancel(&intr);

}




void WimaxBackoffTimer::resume()

{

          Scheduler &s = Scheduler::instance();


          assert(busy_ && paused_);


          paused_ = 0;

          stime = s.clock();


          assert(rtime >= 0.0);

          s.schedule(this, &intr, rtime);
```

```
}

/*
 * Creates a contention slot for the given frame
 * @param s The contention slot
 * @param p The packet to send
 */
ContentionRequest::ContentionRequest (ContentionSlot *s, Packet *p)
{
  assert (s);
  assert (p);
  s_=s;
  mac_ = s_->map_->getMac();
  window_ = s_->getBackoff_start();
  nb_retry_ = 0;
  p_=p;
  backoff_timer_ = new WimaxBackoffTimer (this, mac_);
  timeout_timer_ = new ContentionTimer (this);
  int result = Random::random() % ((int)(pow (2, window_)+1));
  mac_->debug ("At %f in Mac %d Start contention in %f(backoff=%d, size=%d,
ps=%f)\n", NOW, mac_->addr(), result*s_->getSize()*mac_->getPhy()-
>getPS(),result,s_->getSize(),mac_->getPhy()->getPS());
  backoff_timer_->start (result*s_->getSize()*mac_->getPhy()->getPS());
  backoff_timer_->pause();
```

```
}


ContentionRequest::~ContentionRequest ()
{
  //printf ("canceling timeout\n");
  //the timeout timer need not be triggered
  //this can happen when the STA received bw allocation
  //when it is not waiting for one (or it's still in backoff)
  if (timeout_timer_->status()==TIMER_PENDING)
    timeout_timer_->cancel();
  if (backoff_timer_->busy())
    backoff_timer_->stop();
  delete backoff_timer_;
  delete timeout_timer_;
  assert (p_);
  Packet:: free (p_);
}


/*
 * Called when timeout expired
 */
void ContentionRequest::expire ()
{
```

```
}


/*
 * Called when timeout expired
 */
void ContentionRequest::starttimeout ()
{
  timeout_timer_->sched (timeout_);
}


/*
 * Pause the backoff timer
 */
void ContentionRequest::pause ()
{
  if (backoff_timer_->busy() && !backoff_timer_->paused())
    backoff_timer_->pause();
}


/*
 * Resume the backoff timer
 */
void ContentionRequest::resume ()
{
```

```
  if (backoff_timer_->paused() && timeout_timer_->status()==TIMER_IDLE)
    backoff_timer_->resume();
}

/*
 * Creates a contention slot for the given frame
 * @param frame The frame map
 */
RangingRequest::RangingRequest (ContentionSlot *s, Packet *p) :
ContentionRequest (s,p)
{
  type_ = WimaxT3TimerID;
  timeout_ = mac_->macmib_.t3_timeout;
}

/*
 * Called when timeout expired
 */
void RangingRequest::expire ()
{
  mac_->debug ("Ranging request expires\n");
  if (nb_retry_ == (int)mac_->macmib_.contention_rng_retry) {
    //max retries reached, inform the scheduler
    mac_->expire (type_);
```

```
  } else {
    if (window_ < s_->getBackoff_stop())
      window_++;
    nb_retry_++;
    int result = Random::random() % ((int)(pow (2, window_)+1));
    mac_->debug ("Start Ranging contention in %f(backoff=%d, size=%d, ps=%f)\n",
result*s_->getSize()*mac_->getPhy()->getPS(),result,s_->getSize(),mac_-
>getPhy()->getPS());
    backoff_timer_->start (result*s_->getSize()*mac_->getPhy()->getPS());
    backoff_timer_->pause();
  }
}

/*
 * Creates a contention slot for the given frame
 * @param frame The frame map
 */
BwRequest::BwRequest (ContentionSlot *s, Packet *p, int cid, int len) :
ContentionRequest (s,p)
{
  type_ = WimaxT16TimerID;
  timeout_ = mac_->macmib_.t16_timeout;
  cid_ = cid;
  size_ = len;
```

```
}

/*
 * Called when timeout expired
 */
void BwRequest::expire ()
{
  debug2 ("At %f in Mac %d Bw request expires (%d/%d)\n", NOW, mac_->addr(),
nb_retry_, (int)mac_->macmib_.request_retry);
  if (nb_retry_ == (int)mac_->macmib_.request_retry) {
    //max retries reached, delete the pdu that were waiting
    Connection *c = mac_->getCManager()->get_connection (cid_, true);
    int len = 0;
    debug2 ("Dropping packet because bw req exceeded\n");
    while (len < size_) {
      Packet *p = c->dequeue();
      assert (p);
      len += HDR_CMN(p)->size();
      //We want to know when the packet is dropped. Create a new entry
      ((Mac802_16SS*)mac_)->drop(p, "BWR");
      //Packet::free (p);
    }
    //must remove the request from the list
    ((BwContentionSlot*)s_)->removeRequest (cid_);
```

```
  } else {
    if (window_ < s_->getBackoff_stop())
      window_++;
    nb_retry_++;
    int result = Random::random() % ((int)(pow (2, window_)+1));
    debug2 ("Start BW contention in %f(backoff=%d, size=%d, ps=%f)\n", result*s_-
>getSize()*mac_->getPhy()->getPS(),result,s_->getSize(),mac_->getPhy()-
>getPS());
    backoff_timer_->start (result*s_->getSize()*mac_->getPhy()->getPS());
    backoff_timer_->pause();
  }
}
```