RESOURCE ALLOCATION IN WIMAX MESH NETWORKS

STEPHEN ATAMBIRE NSOH
Bachelor of Science, Kwame Nkrumah University of Science and Technology, 2009

A Thesis
Submitted to the School of Graduate Studies
of the University of Lethbridge
in Partial Fulfillment of the
Requirements for the Degree

MASTER OF SCIENCE

Department of Mathematics and Computer Science
University of Lethbridge
LETHBRIDGE, ALBERTA, CANADA

Abstract

The IEEE 802.16 standard popularly known as WiMAX is at the forefront of the technological drive. Achieving high system throughput in these networks is challenging due to interference which limits concurrent transmissions. In this thesis, we study routing and link scheduling in WiMAX mesh networks. We present simple joint routing and link scheduling algorithms that have outperformed most of the existing proposals in our experiments. Our session based routing and links scheduling produced results approximately 90% of a trivial lower bound.

We also study the problem of quality of service (QoS) provisioning in WiMAX mesh networks. QoS has become an attractive area of study driven by the increasing demand for multimedia content delivered wirelessly. To accommodate the different applications, the IEEE 802.16 standard defines four classes of service. In this dissertation, we propose a comprehensive scheme consisting of routing, link scheduling, call admission control (CAC) and channel assignment that considers all classes of service. Much of the work in the literature considers each of these problems in isolation. Our routing schemes use a metric that combines interference and traffic load to compute routes for requests while our link scheduling ensures that the QoS requirements of admitted requests are strictly met. Results from our simulation indicate that our routing and link scheduling schemes significantly improve network performance when the network is congested.

Acknowledgments

"If I have seen farther than other men, it is because I have stood on the shoulders of giants."

Sir. Isaac Newton

In reference to the above citation, I will like to render my profound gratitude to all who have contributed in making this thesis a success. I am first of all thankful to the Almighty God for providing me with the knowledge, strength and good health to pursue graduate studies. I am highly indebted to my supervisor Dr. Robert Benkoczi for his support and directions throughout my masters degree. His thoughtful encouragements and constructive criticisms all culminated in the success of this thesis. He has shown great command in the field of optimization and I am very glad to have benefited from his expertise.

I wish to also express my appreciation to my committee members: Dr. John Zhang and Dr. Saurya Das for their invaluable contributions towards the success of this thesis. Many thanks to all the faculty and staff of the Mathematics and Computer Science department especially Dr. Shahadat Hossain for his course in Advanced Algorithms. I am thankful to my colleagues in the Optimization Research Lab: Mr. Ben Burnett and Mr. Shah Khaled and all my friends in graduate studies especially Miss Fouzia Ashraf Khan and Miss Adriana Appau for their support during my graduate studies.

My sincere thanks go to my dear mum, brothers and sisters and to my uncle Mr. Wilbert Nam-Katoti who have supported me in several ways throughout my studies. For all who have supported me and for whom I am unable to mention because of space, I say thanks to you all.

Contents

List of Tables

List of Figures

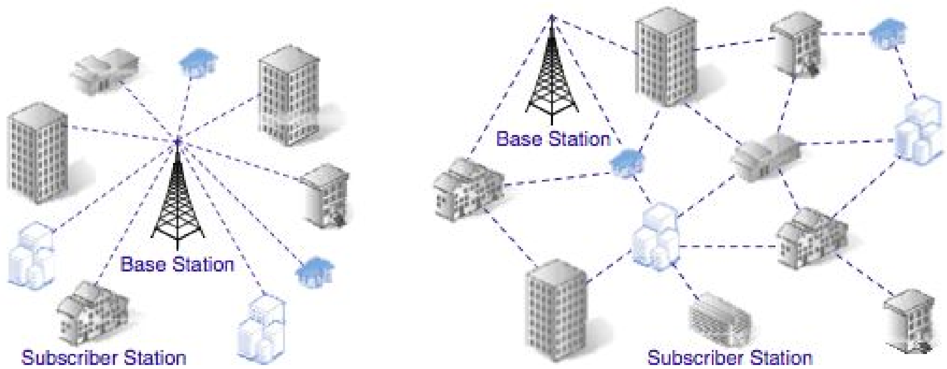
Chapter 1

Introduction

1.1 WiMAX Background and Motivation

The upsurge in the demand for broadband wireless access (BWA) has necessitated more advanced and efficient radio access technologies that can support high bandwidth real-time applications, such as voice over internet protocol (VoIP) and multimedia services over long distances. Also, most parts of the world lack the needed infrastructure for wired networks. In response to this, the IEEE 802.16 working group was set up in 1999 to develop broadband wireless standards. The IEEE 802.16 standard [2] popularly known as WiMAX, an acronym for world interoperatability for microwave access, holds the promise of delivering higher data rates over longer distances. Wireless mesh networks have become a viable alternative for reducing the cost of network deployment. For example, internet service providers who own a core network (backhaul) can interconnect several access points through a mesh topology to provide service to a wider coverage. This is particularly important in remote areas that lack wired infrastructure. Existing wired backhaul technologies such as DSL, T1 or T3 can be very expensive to install and are becoming bottleneck as wireless access speeds increase [3].WiMAX provides an inexpensive alternative and can enable the delivery of last mile BWA even to remote areas that lack wired connection.

A WiMAX network consists of a base station (BS) and several subscriber stations (SSs). The BS provides coverage to about 5 miles with line of sight (LOS) and supports a data rate of about 70mps. Typically the BS is backhauled to the core network while SSs act as access points for end users. The IEEE 802.16 working group has published several series of the standard. The original standard known as 802.16 which was published in 2002 only supported point to multipoint (PMP) operation. In this mode, SSs communicate directly



(a) PMP Architecture (b) Mesh Architecture

Figure 1.1: IEEE 802.16 operating modes

with the BS through LOS and vice-versa.

In 2003, a new standard 802.16a was published. This amendment provided spectrum extension to enhance system performance. To provide wider coverage and enhance scalability, the IEEE 802.16 working group released the 802.16d standard in 2004. This version was regarded as describing a fixed mesh network. The mesh mode allows SSs that are far away from the BS to communicate with the BS through other SSs providing a multihop transmission. In 2005, 802.16e was published and this made provision for mobility support. IEEE 802.16j was published in 2009 to increase the service area and decrease the deployment cost of 802.16 networks. In this later amendment, in addition to BS and SSs, a third type of node called relay station (RS) is introduced. An RS connects an SS to BS, RS to BS and RS to RS. In this work, our focus is on 802.16 d mesh mode. Fig 1.1 shows the network topology of a WiMAX mesh network. WiMAX based mesh networks offer several advantages over current mesh networks that use 802.11 technology which is a decade old and does not suit well for mesh networks.

Despite significant advances in wireless networks, providing QoS guarantees is still a challenging task. The wireless medium has limited bandwidth, higher packet error rate, and high packet overheads that limit the capacity of the network to offer QoS guarantees [4].

Significant modification has been made to the legacy IEEE 802.11 standards to facilitate QoS provisioning, however, design constraints at several layers of IEEE 802.11 restrict its capacity to deliver guaranteed QoS. IEEE 802.16 introduces QoS in mesh through time division multiple access (TDMA) MAC technology. The IEEE 802.16 standard defines the MAC and PHY layer specifications for mesh mode. Among the various proposals specified in the PHY layer, the Orthogonal Frequency Division Multiplexing (OFDM) mode and Scalable OFDMA (SOFDMA) mode have sparkled the most interest as access solutions for the deployment of cellular Wireless Metropolitan Area Network (WMAN). The standard does not however provide any details of these algorithms and leaves the task to vendors to design desired schemes to achieve desired goals. This has become an active area of research in the wireless network community.

Despite the benefits promised by this emerging technology, their design poses serious challenges. The standard provides specifications for MAC and PHY layers but gives no detailed algorithms. Researchers are faced with designing efficient algorithms for routing and scheduling which are decisive to system performance. Ensuring high system performance in mesh networks is a challenging task because of interference inherent in multihop transmissions. In WiMAX mesh networks, the problem is even more complicated since stations have to be scheduled to transmit in an interference-free manner. In this study, we first study the problem of routing in WiMAX mesh networks and provide two novel routing schemes. Our routing schemes aim to increase concurrent transmissions by constructing routes that have less interference. In our first scheme, we provide a tree based routing that selects least interference routes while ensuring load sharing in the network. In the second approach, we take a deviation from the classical tree based route construction by constructing session-based routes. Secondly, we propose a time-slot scheduling algorithm for uplink traffic. Our scheduling algorithm aims to reduce the length of schedule by keeping the BS busy in each time slot. Next, we study QoS provisioning in WiMAX mesh networks. We investigate the impact of routing, link scheduling, channel assignment and CAC in QoS provisioning. We provide a comprehensive framework that considers routing, link scheduling and channel assignment.

1.2 Wireless Mesh Networks

In this section, we present the peculiarities of wireless mesh networks (WMN). WMNs have become popular network systems because of several advantages such as low set-up cost, easy network maintenance, robustness and reliable service coverage [5]. They are already deployed to extend the communication capabilities of existing systems in the mining, telecommunication industry, health care, defense etc. We envision an important role for mesh networks as the infrastructure capable of sustaining ubiquitous connectivity. WMNs are dynamically self-organized and self-configured, with the nodes in the network automatically establishing an ad-hoc network and maintaining the mesh connectivity [1]. This type of infrastructure is relatively inexpensive and very reliable because of several alternative routes for data transmission, and more resilient because each node needs transmit only as far as the next node [6]. Nodes act as repeaters to transmit data from nearby nodes to peers that are too far away to reach, resulting in a network that can span large distances especially over rough or difficult terrain. Furthermore, multi-hopping allows us to reduce the distances over which access points need to transmit over the backhaul links. This can help increase network throughput due to lower path loss and better spatial reuse of resources [3]. Path-loss is the reduction in power intensity of an electromagnetic signal as it propagates through space. The mesh network is composed of two kinds of nodes: mesh routers and gateway routers.The nodes are mostly stationary though they could be mobile in some cases. Gateway routers have the routing capability for gateway/bridge functions in addition to routing functions to support mesh networking. They form the mesh backbone. The mesh

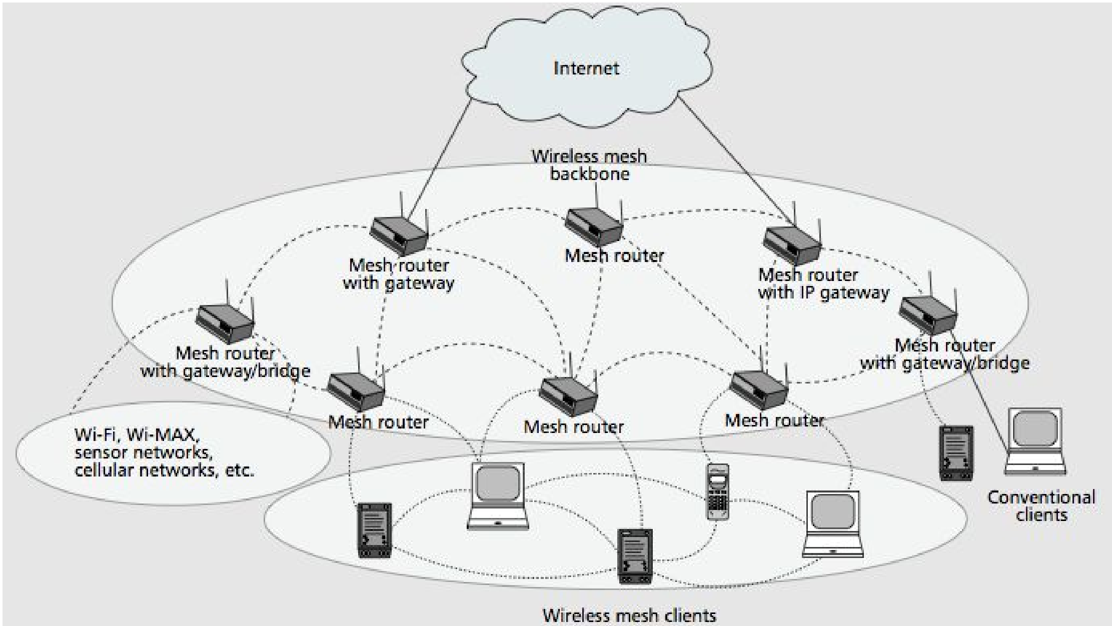


Figure 1.2: Architecture of WMN [1]

is connected to the internet through the gateway routers. The meshing among the mesh and gateway routers creates a wireless backhaul communication system, which provides each mobile user with a low cost and high-bandwidth.

Figure 1.2 illustrates the WMN architecture highlighting the various components. WMNs differ from ad-hoc networks in several ways. Whereas ad-hoc is purely self-organized, in WMN, there is at least one station to provide a bridge to the outside of the system. Also, routers in WMN are fixed or support low mobility but the topology of ad-hoc networks is unstable and dynamic.

1.2.1 Applications of WMN

WMNs have several applications ranging from the provision of wireless internet services for remote areas, to intensive and real-time applications on notebooks and other mobile devices. The list of applications that exploit the mesh networking paradigm is non-exhaustive.

Hence we only identify some common applications in this section.

One of the most common applications of WMN is public internet access. WMNs have become the ideal solution to provide both indoor and outdoor broadband wireless connectivity in urban, sub-urban, and rural environments without the need for extremely costly wired network infrastructure [7]. A large number of mesh routers could be deployed to provide ubiquitous internet service to an entire metropolitan area. Another application area is intelligent transportation system. Public transport services in most cities are using this service. Buses are equipped with mesh nodes that creates a mesh network, allowing bus drivers to deliver information to passengers on a real time basis [7]. This system could be used to address and alleviate transportation congestion problems and improve transportation safety and security. WMNs have also become a viable solution for rescue operations. The 9/11 events and other major disasters have heightened the need for an infrastructure that can allow rescue teams to communicate effectively in delivering their rescue operations and WMN along with ad-hoc networks may prove useful.

1.3 IEEE 802.16 Specification

The IEEE 802.16 standard defines the medium access control (MAC) and physical (PHY) layers, operating in licensed spectrum between 10 and 66 GHz [8]. The 802.16a published in 2003, defines additional PHYs for the 2-11 GHz licensed and unlicensed spectrum and provides enhancements to the MAC for supporting mesh topology. The protocol stack is presented in Figure 1.3

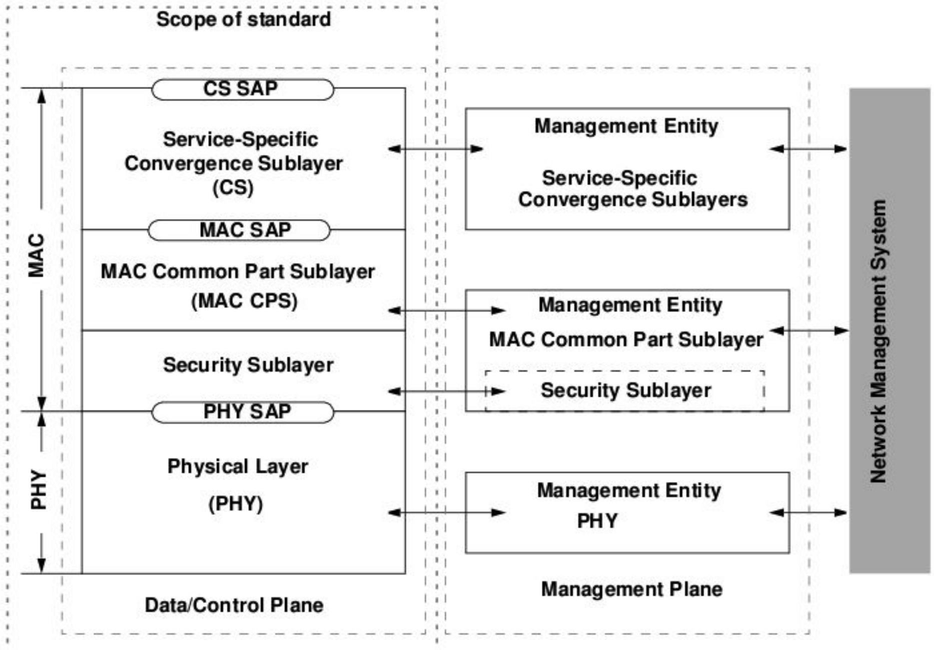


Figure 1.3: IEEE 802.16 protocol stack [2]

1.3.1 MAC Layer

The MAC layer is responsible for coordinating the sharing of the radio channel resources among multiple users. The MAC layer is divided into three sublayers: the service-specific convergence sublayer, common part sublayer and security sublayer. The primary task of the service-specific convergence sublayer is to classify external service data units (SDU) and associate each of them with a proper MAC service flow identifier and connection identifier [9]. The common part sublayer is the core of the 802.16 MAC. It is responsible for the fragmentation and segmentation of each MAC SDU into MAC protocol data units (PDUs), bandwidth allocation, routing, connection management etc. The security sublayer handles authentication, secure key exchange and encryption.

1.3.2 PHY Layer

The PHY layer is responsible for the physical access to the radio channel. It is responsible for the electrical and mechanical processes involved in transmission. The IEEE 802.16 is a TDMA based MAC protocol built on an othorgonal frequency division multiplexing (OFDM) PHY layer (see section 2.5.2). OFDM transforms blocks of bits into constant duration symbols carried on a set of frequency orthogonal pilot carriers [10]. The OFDM symbols are grouped into frames of equal length that repeat over time. WiMAX OFDM features multiple sub carriers ranging in number from 256 up to 2048 [4]. Each subcarrier can be modulated with BPSK, QPSK, 16 QAM, or 64 QAM modulation. Othorgonality offers the advantage of minimizing self interference, a major source of error in wireless communications. The duration of OFDM symbols depends on the total bandwidth occupied by the carriers. The IEEE 802.16 uses OFDM in the 5GHz band with hardware operating at 10MHz bandwidth in the licensed frequency bands and hardware with 20MHz bandwidth in the license-exempt frequency band.

1.4 WiMAX Framing

WiMAX uses a TDMA-based MAC where links are scheduled in a scheduling period known as frame. The frame structure of a WiMAX mesh network is illustrated in Figure 1.4. A frame refers to the set of timeslots for which links are scheduled. A timeslot is the period of time during which a link is scheduled to transmit. The mesh frame consists of a control and data subframe. The control subframe serves as a network control subframe or a schedule control subframe. Frames with network control subframe occur periodically while all other frames have a schedule control subframe [2]. The network control subframe is used to communicate with the network while the schedule control subframe is respon-

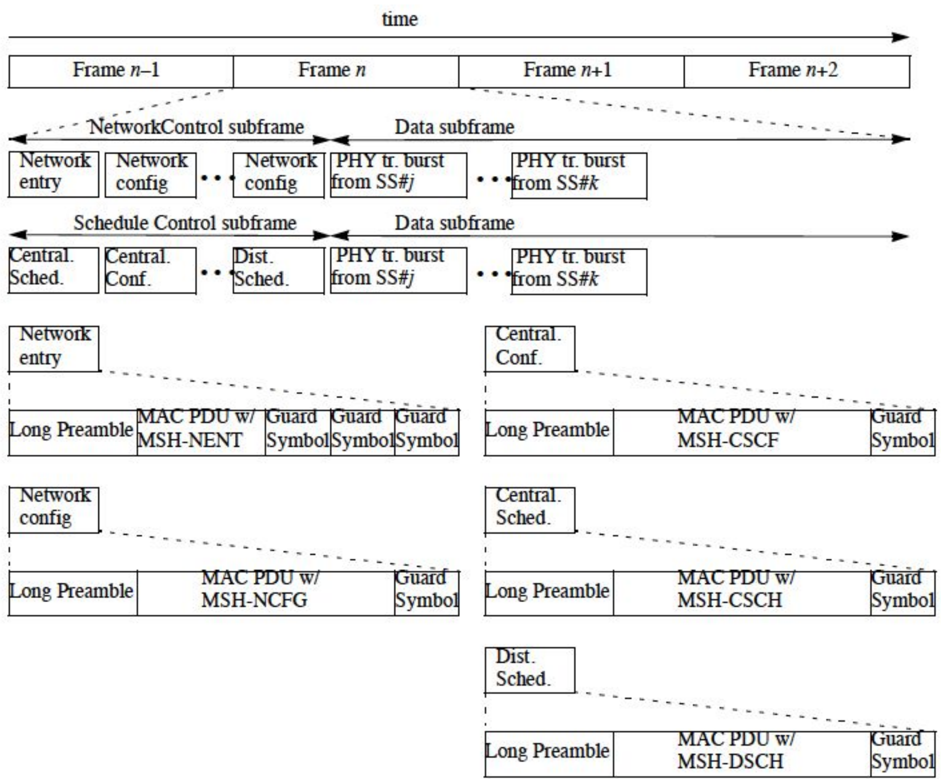


Figure 1.4: IEEE 802.16 mesh frame structure

sible for coordinating the scheduling of data transfers. The control subframe is divided into transmission opportunities each of which is 7 OFDM symbols long. In each subframe transmission, 3 OFDM symbols are guard symbols and 4 symbols are used for data transmission [10]. In the network control subframe, the first time slot is used for the transmission of mesh network entry (MSH-NENT) messages while the remaining slots are used to transmit mesh network configuration (MSH-NCFG) messages. The schedule control subframe is made up of centralized scheduling (MCH-CSCH) and distributed (MSH-DSCH) control units which are used in centralized and distributed scheduling respectively.

The data subframe is divided into slots which are used for transmitting and receiving data according to the scheduling results. Depending on the frame size and the size of the control subframe, the data subframe may contain fewer than 256 transmission opportunities. The IEEE 802.16 standard restricts the data subframe to contain a maximum of 256 slots.

1.5 Scheduling Modes

IEEE 802.16 mesh protocol specifies two TDMA scheduling protocols: centralized and distributed scheduling protocols. In this section, we briefly introduce these schemas.

1.5.1 Centralized Scheduling

In the mesh centralized scheduling, the BS is responsible for controlling and coordinating the scheduling for the entire network. First, SSs monitor the traffic from their wireless terminals and use this information to request end-to-end bandwidth from the BS. The BS then uses these requests to assign bandwidth to all links in the network. Two control messages: mesh centralized scheduling (MSH-CSCH) and mesh centralized scheduling configuration

(MSH-CSCF) are used in the centralized scheduling protocol. The BS uses the MSH-CSCF packets to distribute the network topology [10]. MSH-CSCH packets are used by SSs and BS to request end-to-end bandwidth and assign end-to-end bandwidth respectively (section 1.5.

SSs request bandwidth from the BS by sending MSH-CSCH to their parents in the routing tree. The routing tree is the union of all paths from each SS to the BS. The BS uses all requests from SSs to determine the bandwidth for each link in the network and multicasts the schedule to SSs using MSH-CSCH messages which are multicast by intermediate nodes until all nodes in mesh receive the schedule. The new schedule takes effect after the last node receives the MSH-CSCH message from its parent. Centralized schedule is more efficient and suits well for multihop networks [11] and it is the subject of this work. The scheduling procedure is fairly simple, however the connection setup delay is significant because of the request and grant procedure involved. Hence, centralized scheduling is not suitable for occasional traffic needs [8]. In chapter 4, we present details about centralized scheduling algorithms.

1.5.2 Distributed Scheduling

In distributed scheduling, mesh nodes themselves determine the schedule of data transmissions. First, a node wishing to request for transmission opportunities sends a request to its neighbours. One or more neighbours respond with a range of available transmission opportunities. The node then chooses a subrange of these transmission opportunities and confirms that it will use them. This scheme usually involves a three-way handshake - request, grant and confirm messages during which transmission slots are selected. Distributed scheduling does not ensure fairness and QoS guarantees since link bandwidth depends on grants from their neighbours. Distributed scheduling is however more flexible and efficient for connection setup and data transmission. There are two forms of distributed scheduling: coordinated and uncoordinated distributed scheduling. We briefly describe these in the next sections.

Coordinated Distributed scheduling

In coordinated distributed scheduling, nodes compete for channel access using a pseudo-random election algorithm based on scheduling information of their two-hop neighbours. Nodes compete for transmitting scheduling packets (MSH-DCSH) in the scheduling control subframe so that there is no contention in the data time slots [8]. The scheduling control subframe is divided into transmission opportunities and nodes contend to transmit MSH-DSCH messages in these transmission opportunities using a distributed election algorithm. When a node wins the election algorithm, it sets the temporary transmission opportunity as its transmission time and broadcasts it to the neighbours.

Uncoordinated Distributed Scheduling

This scheme is used for fast setup of new, temporary data bursts between neighbours. A node wishing to request for transmission opportunities first observes the idle slots of the current schedule by looking at the control messages and uses a random-access algorithm. The request message contains a list of the neighbours with whom it wants to transmit data to. The message also lists the idle slots in its neighborhood [8]. The node granting the request must also make sure that transmission of the grant message does not cause any collision in its own neighborhood. Upon receiving this grant message, the requesting node must confirm the schedule by sending another MSH-DCSH message after which data transfer then takes place.

1.6 Thesis Contribution

The IEEE 802.16 standard provides specification for the MAC layer but does not provide any implementation details. In this thesis, we study the resource allocation problem in WiMAX mesh networks and make the following novel contributions.

- We propose routing algorithms for centralized scheduling that incorporate interference and load sharing in the network.
- We present a link scheduling and channel assignment scheme that have outperformed most of the existing proposals in the literature through our simulations.
- We also investigate the impact of routing, link scheduling, CAC and channel assignment on QoS. We provide a comprehensive framework for QoS guarantees in WiMax mesh networks and we show evidence that load sharing is very crucial in QoS provisioning when the network is congested.

The work in this thesis has led to the following publications:

1. S. Nsoh and R. Benkoczi, *Towards global connectivity by joint routing and scheduling in wireless mesh networks*, in 3rd International Conference on Mobile, Ubiquitous and Intelligent Computing, 2012. pp. 214-219, IEEE, 2012.
2. S. Nsoh and R. Benkoczi, *Routing and Link Scheduling with QoS in IEEE 802.16 mesh networks*, In IEEE Communication and Wireless Networking Conference, 2013, China. (Under review)

1.7 Thesis Organization

The remainder of this thesis is organized as follows:

Chapter 2 outlines the system model used in this research. We present various popular system models and elaborate on their practicability. Assumptions used in this research are also discussed in this section. In chapter 3, we describe our routing schemes. First, we provide a detailed specification of IEEE 802.16 mesh routing. Next we give a summary of existing research on routing. Finally, we present our routing schemes which provide a striking balance between interference and load sharing. In our first scheme, we consider tree based routing while in our second scheme we consider multipath routing and quantify its gains.

Chapter 4 presents IEEE 802.16 centralized scheduling and channel assignment. We outline details of the scheduling problem. We summarize related research on scheduling and propose our throughput aware scheduling and channel assignment scheme. WiMAX QoS provisioning is described in chapter 5. We examine the QoS requirements of different service classes and provide the problem definition. A summary of existing related research is presented. Next, we propose novel routing schemes to improve QoS provisioning. Our CAC scheme is then outlined. We provide simulations to evaluate our model and present results from the experiments

Chapter 6 presents a summary of our conclusions and discussions about some future work of this research.

Chapter 2

System Model

2.1 Introduction

In this chapter, we provide an overview of popular system models used in the WiMAX architecture. We compare various system models and explain the rationale behind the chosen models in this work.

2.2 Radio Models

A radio, also known as a transceiver, is a key device in wireless communication. In wireless communication, a radio is an electronic device capable of transmitting/receiving signals to/from the wireless medium through an antenna and transforming the signals between the wired circuit and the wireless medium. A radio can be classified as half duplex or full duplex.

2.2.1 Half Duplex Radio

A node equipped with a half duplex radio can only be tuned to transmitting or receiving. In this mode, a node cannot be transmitting and receiving at the same time even if different channels are used for transmitting and receiving. An incoming link of a node will interfere with an outgoing link of the node. However, with orthogonal channels, it is possible to have multiple incoming links or outgoing links activated at the same time. When a node is in transmitting mode, it has to be switched to receiving mode before it can receive signals. There is usually a time lapse of about 100μs when the radio is switched from one mode to another. In this work, we ignore the switching time and only focus on the time taken for packets to tranverse (move) from source to destination. We assume nodes to be half duplex and hence cannot transmit and receive at the same time.

2.2.2 Full Duplex Radio

Full duplex on the other hand allows both transmission and receiving simultaneously. This means multiple links irrespective of outgoing or incoming can be active concurrently provided they use different channels. It is worth pointing out that full duplex radios are expensive and complex as compared to half duplex radios.

2.3 Antennas

An antenna is an electrical conductor or system of conductors used either for radiating electromagnetic energy or for collecting electromagnetic energy [12]. During transmission, radio-frequency electrical energy from the transmitter is converted into electromagnetic energy by the antenna and radiated into the surrounding environment. The reverse process happens during the reception of a signal. An antenna radiates power in all directions but typically, does not perform equally well in all directions. In reality, the antenna gain is a function of the relative angle to the antenna in a 3-dimensional space. However, radiation patterns are almost always depicted as a two-dimensional cross section of the tree-dimensional pattern which is frequently considered constant. Antenna gain is a measure of directionality. It is defined as the power output, in a particular direction, compared to that produced in any direction. Antennas can be classified as directional or omnidirectional.

2.3.1 Omnidirectional Antennas

Omnidirectional antennas radiate power uniformly in a plane with a homogeneous antenna gain in all directions. They are also called non-directional antennas because they do not favor any particular direction. These antennas are the cheapest and simplest to model. Omnidirectional antennas are the most dominant type of antennas in WiMAX and mesh networks. However, omnidirectional antennas create more interference in the network by radiating power in all directions. In this work, we also assume the antenna model to be omnidirectional.

2.3.2 Directional Antennas

Directional antennas radiate power only in directions toward intended receivers. Because the amount of radio-frequency energy is the same, but distributed over less area, the apparent signal strength is higher. This apparent increase in signal strength is the antenna gain. These antennas create less interference and are suitable for near LOS coverage such as hallways and long corridors. However, they provide less coverage as compared to omnidirectional antennas.

2.4 Transmission Models

The transmission model defines the network topology of a set of wireless devices. The ability of two nodes to communicate with each other depends on the Euclidean distance between the nodes, the transmitting power used by the transmitter to send the signal and the surrounding environment.

Assume there is a set $V = \{v_1, v_2, ...v_n\}$ of n wireless terminals deployed in a region.

Denote by E the set of directional communicating links between pairs of the nodes. The graph $G = \{V, E\}$ is called the communication graph. A graph $G = (V, E)$ is a collection comprising a set V of vertices or nodes and a set E of edges connecting the vertices. A graph can be directed or undirected. In a directed graph, there is a direction associated with each edge while edges in an undirected graph have no directions associated with them. The communication graph could be directed or undirected depending on the power levels of the transmitters. When the transmitters transmit at different power levels, then the resulting communication graph is directed, otherwise it is undirected. In this work, although we assume the transmitters transmit with the same power level, we still model the communication graph as directional. This allows us to deal separately with uplink and downlink traffic. Uplink traffic refers to traffic directed towards the BS while down link traffic is directed away from the BS.

Every wireless terminal v_i has a transmission range r which denotes the maximum distance within which data transmitted by v_i can be successfully received by the receiver. Given the transmission range r of a node, the transmission region of the node is a segment of length $2r$ centered at the node for one-dimensional networks, a disk of radius r for two-dimensional networks or a sphere of radius r for three-dimensional networks. The transmission range of a node is dependent on the transmission power and the propagation model [6].

2.4.1 Unit Disk

When all nodes transmit with the same power level, then each node will have the same transmission range R_T. The resulting communication graph is called a unit disc graph (UDG). In this case, $(v_i, v_j) \in E$ if $\| v_i - v_j \| \leq r$. Usually, the transmission range of nodes is normalized to one unit and two nodes can communicate directly with each other if the distance between them is no more than one unit. In unit disk graphs, the resulting communication graph is unidirectional. This model is widely used because of its simplicity which makes it easy to analyze. In our experiments, we model our graphs as UDG.

2.4.2 *Geometric Intersection*

When different nodes transmit with different power levels, the resulting graph consists of disks with different sizes. In this model, there exits a directed edge (v_i, v_j) if and only if v_j is within the transmission region of v_i. Since Geometric graphs are represented as directed graphs, they are more intricate to analyze as compared to UDG.

2.5 Channel Models

The medium access control (MAC), is the mechanism responsible for determining which device has access to the transmission medium at a given time. It selects the physical channels and establishes or releases connections on those channels. The channel model depicts how the available spectrum is used. A spectrum is defined as the range of frequencies that signals contain. The entire spectrum could be used as a whole thereby forming a single channel or it can be divided into multiple orthorgonal subchannels through frequency division multiplexing (FDM).

2.5.1 *Single Channel*

In the single channel mode, the entire frequency band is assigned to a single user. Multiple users can transmit on the channel as long as they do not interfere with each other. This model is the simplest to study and forms the foundation of all other channel models.

2.5.2 *Orthogonal Frequency Division Multiple Access (OFDMA)*

Orthogonal frequency division multiplexing (OFDM) is a MAC technique for transmitting large amounts of digital data over a radio wave. The technology works by splitting the radio signal into multiple smaller sub-signals that are then transmitted simultaneously over several subcarriers at different frequencies. OFDMA technique is similar to OFDM, however, whereas in OFDM, all subchannels are dedicated to one user, OFDMA distributes the subchannels to multiple users. In OFDMA, the entire spectrum is divided into subcarriers which are grouped into subchannels through sub-channelization. Sub-channelization defines subchannels that can be allocated to SSs depending on their channel condition and data requirements. Two forms of sub-channelization exist: diversity permutation and contiguous permutation. In the former, subcarriers that form a subchannel are chosen randomly from the entire frequency band while in the latter, a subchannel is made up of adjacent subcarriers. The diversity permutation mode is recommended for highly mobile users, or for users with very low signal-to-interference-plus-noise ratio (SINR) [13]. We adopt OFDMA channel model in this work. Each node can be assigned several subchannels (channels) to increase throughput as long primary and secondary interference are not violated.

2.6 Interference

Interference remains a major issue in wireless communication. If two links cannot be established at the same time on the same channel, we say there is some interference between them. A link refers to a directed edge between the source for data and its intended recipient. Interference hampers coverage and capacity and limits the effectiveness of new and existing systems. Every terminal in a wireless network has an interference range $R_I(v_i)$ such that every node v_j within that range is interfered when v_j is not the intended receiver. Most researchers, for simplicity, treat the transmission range of a node as its interference range. However, in practice, the interference range is not necessarily the same as the transmission range. Typically, $R_T(v_i) < R_I(v_i) \leq cR_T(v_i)$ for some constant $c > 1$ [6]. In practice, $2 \leq c \leq 4$. In this work, we also consider the transmission range to be the same as the interference range. There are two kinds of interference: primary interference and secondary interference.

2.6.1 Primary Interference

Primary interference occurs between links that share a neighbour. Figure 2.1 illustrates primary interference. In Figure 2.1(a), there is interference when a node receives from multiple neighbours at the same time. We assume each node has only one transceiver. In Figure 2.1(b), a node transmitting to multiple neighbours causes interference while in Figure 2.1(c) a node cannot transmit and receive at the same time because of interference. Even with unlimited number of channels, it is impossible to eliminate primary interference.

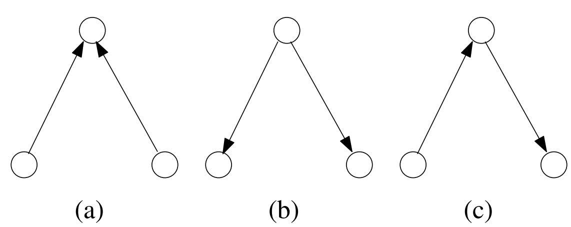


Figure 2.1: Primary interference

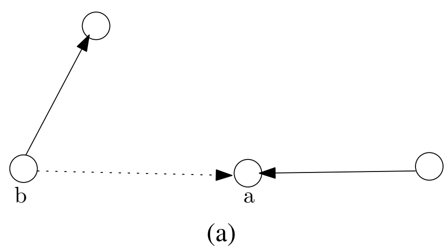


Figure 2.2: Secondary interference

2.6.2 Secondary Interference

Secondary interference occurs when a node a is in the interference region of another node b and b is transmitting while a is not the intended receiver. Secondary interference is illustrated in Figure 2.2. However, only Figure 2.2(a) needs to be considered in TDMA networks [14]. Although Figure 2.2(b) exists, a sender can still transmit while other transmissions are ongoing.

2.6.3 Interference Models

In practice, the degree of interference can be measured, but in theoretical studies, several models based on the relative position of the nodes have been adopted.

Protocol Interference

In this model, a transmission by a node v_i is successfully received by node v_j if and only if the intended receiver v_j is sufficiently apart from any source of any other simultaneous transmission i.e., $\|v_k - v_j\| \geq (1+\eta)\|v_i - v_j\|$ for any $v_k \neq v_i$ where $\eta > 0$ is a constant [6]. The protocol interference model does not necessarily provide a comprehensive view of reality because of the aggregate effect of interference from multiple simultaneous transmissions in a wireless network. However, it does provide some good estimations of interference and enables a theoretical performance analysis of a number of protocols designed in the literature.

Physical Interference

In the physical interference model, signal to interference noise ratio (SINR) is used to aggregate the interference in a network. This is because in reality, a link will receive interference from all simultaneously active links on the same channel and those signals all contribute to the interference part of the SINR computation. Assume node i is transmitting to node j and let $S_{i,j}$ denote the signal strength from i at node j. Assume $N_{-i,j}$ is the aggregate signal strength at j by all nodes other than i that are transmitting on the same channel at the same time. The SINR of the transmission from i to j is expressed as:

$$SINR_{i,j} = \frac{S_{i,j}}{N_{-i,j} + N_0} \tag{2.1}$$

where N_0 is the ambient-noise power level. The transmission from i to j can be successfully received if the SNIR is above some threshold. In this work, we model interference using the protocol interference model. We treat the transmission range as the same as the interference range of the node. Two nodes thus interfere if they are within the interference range of each other. For secondary interference, the links can be active simultaneously provided they use different channels. A link can be assigned more channels to increase throughput. However, no node can be involved in communication with more than one neighbour.

Chapter 3

Interference and Load Aware Routing

3.1 Introduction

In WiMAX mesh network, the BS serves as the gateway to the external network and all other nodes have to send their requests through the BS through multihop transmission. For resource allocation, the IEEE 802.16 standard specifies that the assignment of link bandwidth should result in a tree rooted at the BS. The resource allocation problem consists of computing a routing tree and scheduling links in the tree to maximize system performance. These two operations can either be performed separately or jointly. To reap the full benefits promised by this emerging technology, researchers have to produce efficient algorithms for the resource allocation problem. In this chapter, we study the problem of centralized routing and tackle the scheduling problem in Chapter 4.

Routing in WMNs poses several design challenges. The issues that need to be addressed in the routing design include both short and long time scales. A good mesh routing scheme has to ensure both long-term route stability and achieve short-term opportunistic performance [15]. The routing design has to also ensure robustness against a wide range of soft and hard failures, ranging from transient channel outages, links with intermediate loss rates to failing nodes. In addition, the routing scheme has to take into consideration the interference and traffic load in the network. A routing scheme of less interference allows spatial reuse which increases system throughput. The routing problem is thus decisive in system performance.

The standard provides specifications for centralized routing but does not give any detailed routing algorithm except a random routing in which SSs randomly select their parents while building a tree. In this chapter, we present two novel routing schemes. First, we propose a tree based routing which produces less interference while ensuring load balance in the network. Ensuring load balance in mesh routing is crucial since traffic congestion is a major source of delay in mesh networks. In our second routing scheme, we take a deviation from the classical tree routing and present a session based routing. In this scheme, we exploit multipath routing to effectively ensure load sharing in the network. We do a quantitative study in Chapter 5 to investigate how much more efficiently the resources are used with the session based routing.

3.2 Related Work

Centralized routing in mesh networks has attracted much attention from the research community. Quassem *et al.* [15] provide an excellent survey on routing in mesh networks. Several works like [16, 17, 18] aim to find shortest routes using breadth first search (BFS) approach. In [3], a modified Dijkstra algorithm for shortest path is used for routing tree construction. In this work, the authors define $1/C(e)$ as the distance along an edge where $C(e)$ is the capacity of the edge. The drawback to shortest path routing is that every node will choose the shortest route and this may lead to bottlenecks in the network. Also, the shortest path does not account for interference and traffic load which are essential metrics. In [19], the authors try to take both traffic load and hop count into account during the routing tree construction. The routing tree is constructed in a bottom-up approach where an SS with the highest degree is first attached to its neighbours with less or equal hop-count. The SS with the least traffic load is then chosen as the parent node. This scheme may lead to longer routes as SSs may choose neighbours with the same hop-count. Yang *et al.* [20] present a multipath routing scheme by multiplying the interference, data traffic and traffic class to obtain the best route.

In [21], the routing tree is constructed by having SSs choose parent nodes with the least neighbors along the path to the BS. When an SS joins the tree, it may affect the degrees of nodes in the tree which could cause some nodes to reselect their parent node. This may lead to an infinite looping. Fanchun *et al.* [22] propose a maximal parallelism routing and a *Min Max* BFS routing schemes. In the former, the goal is to maximize the sum of the weights on pairwise non-interfering edges connecting two consecutive layers of nodes using traffic load as weights on edges. In the latter, the authors aim to minimize the degree of a BFS tree. This approach does not guarantee load sharing since node degrees do not necessarily reflect the traffic demands on links.

An interference aware routing scheme is proposed in [23]. The goal is to reduce the level of interference and maximize concurrent transmissions. In this work, the concept of blocking value and blocking metric are introduced. The blocking value (B) of a node i is the number of nodes that are interfered when i is transmitting while the blocking metric (Bk) of a route is the sum of blocking values of the source node as well as the intermediate nodes on the route. When a node is considered to join the partially constructed routing tree, it selects the parent node with the least blocking metric. Although this work captures the interference in the network, it does not take into account the traffic load in the network. As a result, every node will choose the route with the least blocking metric and this will lead to congestion. To overcome this, [22] extend the work of [23] to capture traffic load. In this work, the authors define blocking value as the degree of the node in the communication graph multiplied by the number of packets (traffic demand). This scheme takes into account the traffic load in the network but can still lead to congestion. For example, without updating the blocking metrics of nodes, any node joining the network has no information of the already existing children of the parent node. When an SS joins a parent node, the blocking metric of the parent node needs to be updated to reflect the increase in traffic load along that route else more SSs are likely to join the same parent node. This is the motivation behind our routing schemes.

3.3 Problem Definition

We are given a directed communication graph $G = (V, E)$, where V denotes the set of nodes (BS and SSs) and E the set of directed edges connecting the nodes. We model the traffic demands of SSs as a discrete number of packets denoted by $demand(i)$. Each node i has a set of nodes that are interfered when node i is transmitting. We want to construct a route from every SS to the BS such that the total interference level is minimized while ensuring load sharing in the network.

3.4 Proposed Scheme

In this section, we present two routing schemes for mesh networks. We propose a tree based routing in section 3.4.1 that updates the blocking metrics of nodes while constructing the tree. In section 3.4.2, we present another novel routing scheme that exploits multipath routing to effectively ensure load sharing.

3.4.1 Interference and Load Aware Routing (ILR)

The choice of routing tree in mesh networks is crucial in achieving high throughput performance. In order to achieve maximum concurrent transmission, we construct a routing tree that minimizes interference in the network. Without a load sharing mechanism, some links will be congested and become the bottleneck of the network. To overcome this, we extend the work of [23] to capture both interference and traffic load.

Definition 1 *Blocking value* (B) *of a node i is the number of nodes that are interfered when node i is transmitting.*

Definition 2 *Blocking metric* (Bk) *of a path is the sum of blocking values and demands of*

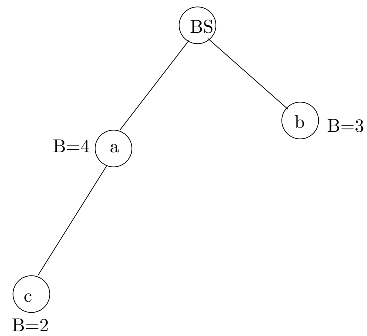


Figure 3.1: Example graph with blocking values

nodes on the path

Consider the diagram in figure 3.1 showing the blocking values of the nodes. For ease of exposition, we denote $Bk(i)$ to be the blocking metric of the path from i to the BS. Assume the normalized demands of nodes a,b and c are d_a,d_b and d_c respectively. We compute the blocking metrics as follows:

$Bk(a) = B(a) + d_a + d_b = 4 + d_a + d_b$

$Bk(b) = B(b) + d_b = 3 + d_b$

$Bk(c) = Bk(a) + B(c) = Bk(a) + 2$

The construction of the routing starts with the BS and builds up by adding SSs one at a time. At each point, we have a partially constructed routing tree and a node is selected to join the tree. When a node is considered to join the tree, it selects the parent node with the least blocking metric. After the new node joins the tree, we need to update the blocking metric of the parent node to reflect the increase in traffic on the path. We update the blocking metric by adding the quantity $\lambda \times demand(p)/Max$ (normalized demand) to the blocking metric where Max is the maximum traffic demand in the network and $demand(p)$ is the traffic demand of node p. Since the blocking metric of a node depends on the blocking metric of its parent, any time we update the blocking metric of a node, we need to also update the blocking metrics of all its children in the tree. For example, in Figure 3.2(a), node h is considered to join the tree and it chooses node a as the parent node (assume node a has the least blocking metric). After node h joins the tree, we update the blocking metric of node a by adding the quantity $\lambda \times demand(h)/Max$. This will cause a change in the blocking metrics of nodes h, d, g and c which will also need to be updated as well by adding the same quantity as shown in Figure 3.2(b).

Updating the blocking metrics is the crux of this algorithm since we are combining two metrics: interference and traffic load. We tried several quantities and we obtained the best results by setting λ to 2. Since traffic demands of nodes change over time, the routing needs to be performed at the beginning of each frame. Details of the routing algorithm are

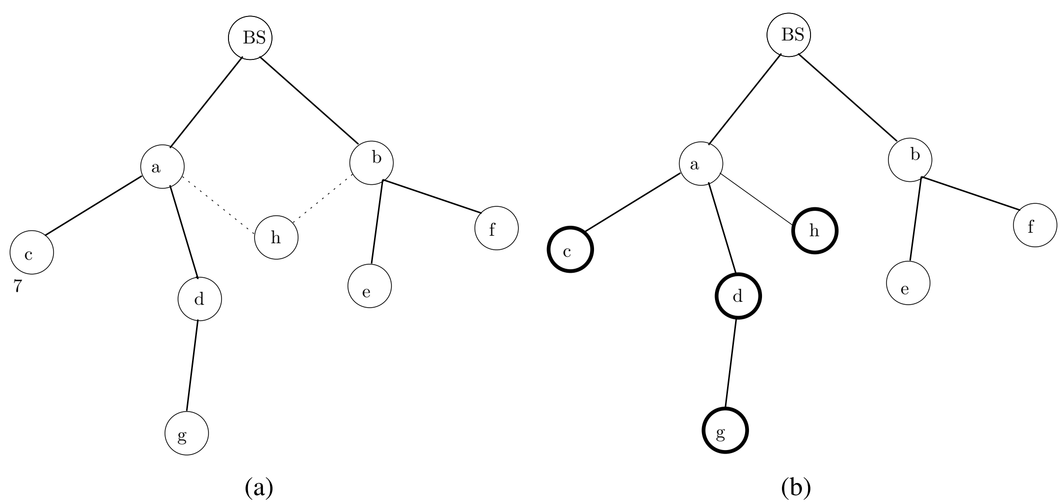


Figure 3.2: Updating blocking metrics

presented in Algorithm 1.

3.4.2 Interference and Load Aware Multipath Routing (ILMR)

Although the IEEE 802.16 standard specifies that the routing should form a tree rooted at the base station, we take a deviation from this to investigate the gains of multipath approach in session based routing. The traditional tree-based routing forces each node to use only

Algorithm 1 ILR

Input: $G = (V,E), traffic demands$

Output: $T = (V,E')$

$S \leftarrow \{1\}$ set of selected nodes (starts with base station)

$N_s \leftarrow \{2,3,...,N\}$

$W \leftarrow Neighbour(1)$

while $N_s \neq \oslash$ **do**

$(i,j) \leftarrow \underset{j \in S,\, i \in Neighbour(j):\, i \in W}{\text{argmin}} B(i) + Bk(j)$

Add i to S

$Parent(i) = j$

$Bk(i) \leftarrow Bk(j) + B(i)$

$Bk(j) \leftarrow Bk(j) + demand(i)/Max \times \lambda$

propagate the update through the tree.

$W \leftarrow W \cup Neighbour(i)$

$N_s \leftarrow N_s - \{i\}$

end while

one path for forwarding all data. This means some links in the network are never used for communication and this does not exploit alternative paths which may lead to better resource utilization. The use of multiple alternative paths is crucial in ensuring effective load sharing in the network. To achieve this, we modify our routing scheme described in the previous section to allow nodes to choose a complete path instead of just attaching to a parent node. For example, in Figure 3.3, node 10 is considered to join the network and it only has connection with node 6. Node 8 at this time has the least blocking metric but has no connection with node 10. However, there exists an edge between node 6 and node 8 which means node 6 can forward data from node 10 to node 8. Considering the already existing traffic load along the route from node 6 to the BS, it will be better for node 10 to choose the route $6-8-7-BS$.

This scheme can effectively ensure load sharing and produces results approximately 90% of the lower bound as shown in section 3.5. After a node joins the routing, we update the blocking metric of all nodes on the route and propagate the update in the network as

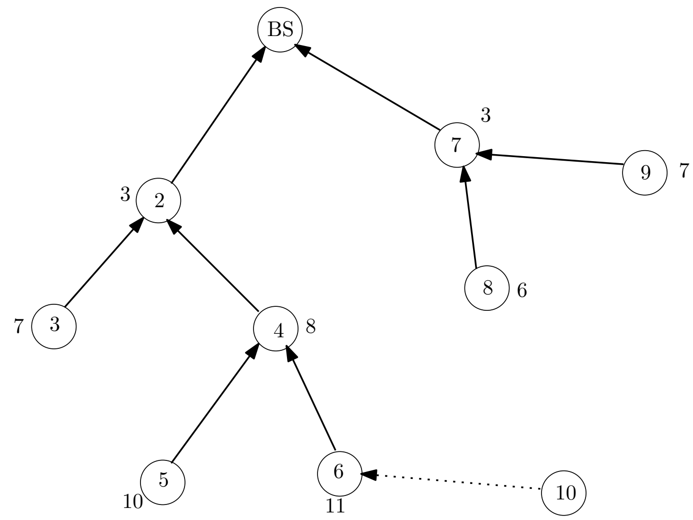


Figure 3.3: multipath routhing

shown in Figure 3.4. Details of ILMR is presented in Algorithm 2.

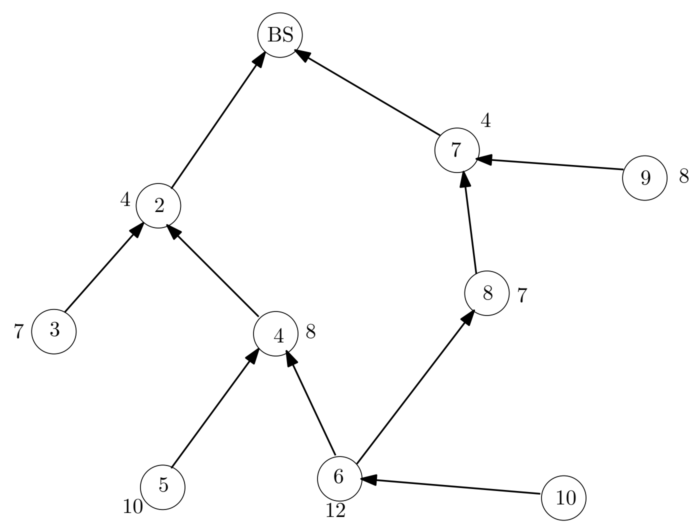


Figure 3.4: multipath routhing

Algorithm 2 ILMR

Input: $G = (V,E), trafficdemands$
Output: $T = (V,E')$
  $S \leftarrow \{1\}$ set of selected nodes (starts with base station)
  $N_s \leftarrow \{2,3,...,N\}$
  $W \leftarrow Neighbour(1)$
  while $N_s \neq \oslash$ **do**
    $(i,j) \longleftarrow \underset{j \in S,\, i \in Neighbour(j):\, i \in W}{\text{argmin}}\, B(i) + Bk(j)$
    find route with minimum blocking metric from *i* to *BS*
    update blocking metrics of nodes
    Add *i* to *S*
    $W \leftarrow W \cup Neighbour(i)$
    $N_s \leftarrow N_s - \{i\}$
  end while

3.5 Experiments

3.5.1 Experimental Setup

In this section, we provide simulation results to evaluate the routing schemes proposed in this chapter. We compare our routing schemes with shortest path routing (SP) and the scheme proposed in [23] which we label as IR. We use the maximum degree first select (MDFS) algorithm proposed in [16] for the scheduling. This scheme processes links in the order of their degree (number of children). The experiments were preformed with square grid and random graphs with varying number of nodes. The demands of nodes were randomly generated between 1 and 8. For the random graphs, nodes are randomly placed in a 100×100 square area and two nodes are connected by an edge if they are within the transmission range of each other. The transmission range and interference range are set the same in the experiments and the number of channels was set to 1 to force bottleneck in the network sooner. In each case, the experiments were repeated with varying demands and the average length of schedule recorded. We use two metrics: length of schedule and channel utilization ratio to compare the different schemes. These metrics are defined in section 3.5.2 and 3.5.3 respectively. Figure 3.5 - 3.7 illustrate the results from the experiments.

3.5.2 *Length of Schedule*

The length of schedule is the total number of timeslots needed for all packets to reach the BS. Depending on the routes used for data transmission, packets experience varying delays. Figures 3.5 and 3.6 illustrate the length of schedule obtained with varying topologies. The results indicate that our routing schemes produced the best results in all the topologies. ILMR produced the least scheduling length for all cases followed ILR with IR outperforming SP. By effectively distributing the load over the network, more links can be scheduled concurrently leading to shorter lengths of schedule for ILR and ILMR. ILR outperformed IR by approximately 4% while ILMR outperformed IR by about 8%.

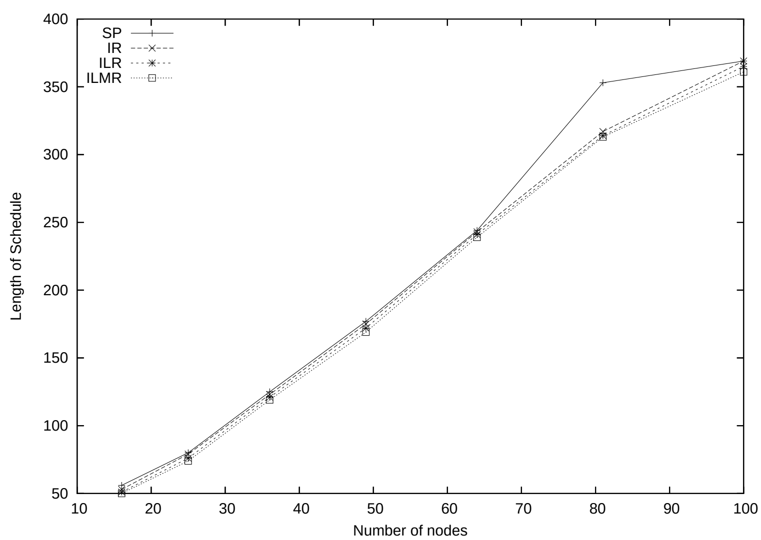
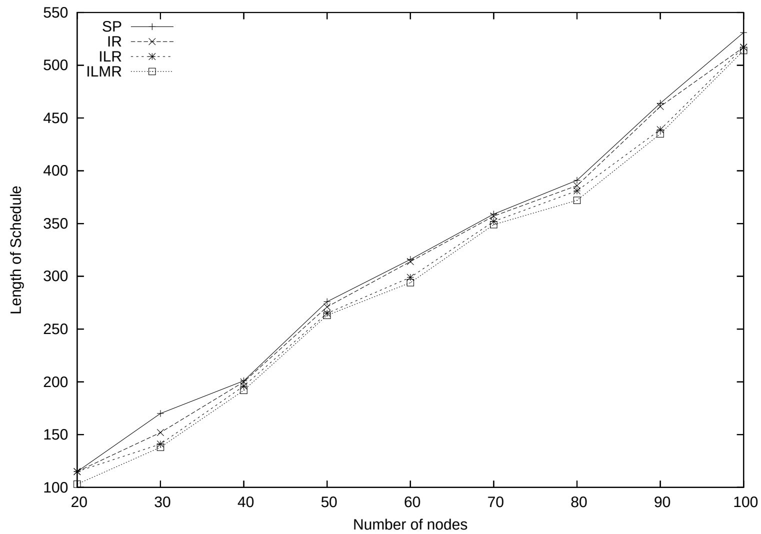
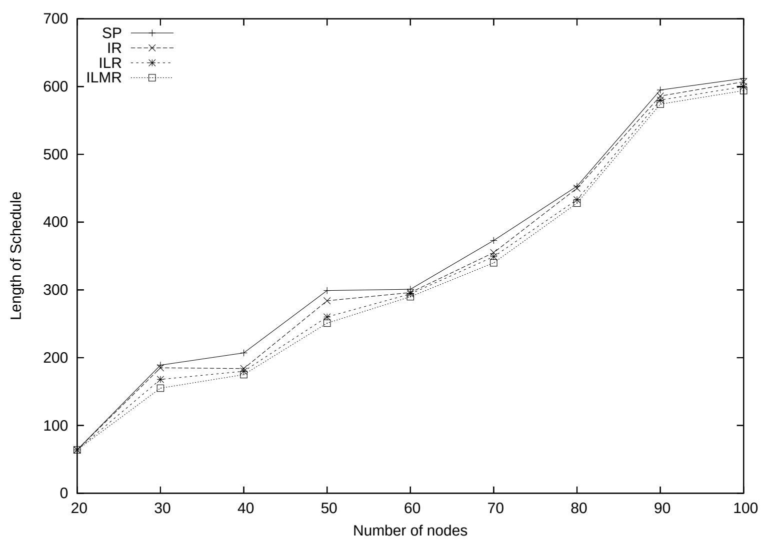


Figure 3.5: Regular square grid graph



(a) Dense graphs



(b) sparse graphs

Figure 3.6: Random graphs

3.5.3 Channel Utilization Ratio

In this section, we investigate the channel utilization ratio (CUR) in the routing schemes. Channel utilization ratio is a measure of concurrent transmissions in the network and can be computed from the expression:

$$CUR = \frac{\text{number of minislots used by all nodes to transmit data packets}}{\text{number of nodes} \times \text{number of minislots}} \tag{3.1}$$

A minislot refers to a slot in channel-timeslot grid. A 100% CUR is obtained when every node transmits on all channels in every timeslot. Figure 3.7 shows the results obtained from

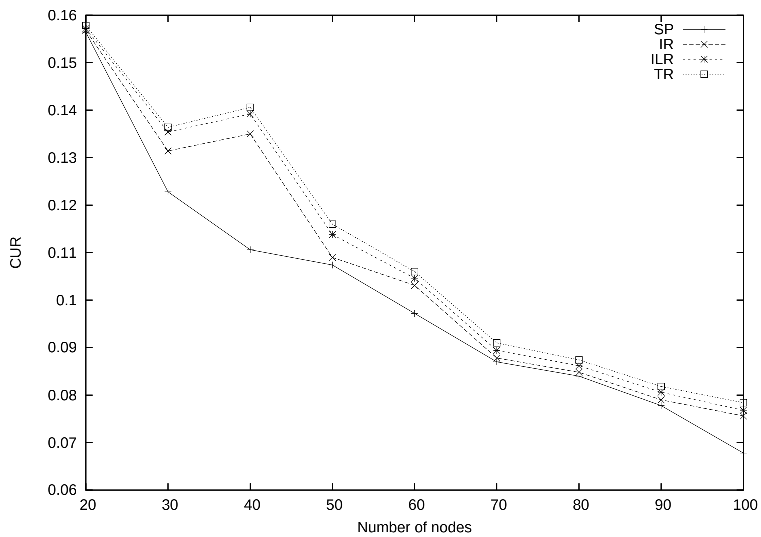


Figure 3.7: CUR vs number of nodes

the experiments. Our scheme uses less interfering routes and at the same time distributing the traffic load across the network. As a result, more links are scheduled concurrently and that accounts for the better performance in terms of channel utilization ratio. The results follow the same trend with ILMR producing the best performance followed ILR. IR also uses less interfering routes but does not ensure load sharing. IR outperforms SP which only uses shortest paths without regard to interference. The results indicate that our routing schemes outperform IR and SP and achieves about 5% throughput enhancement.

Chapter 4

Throughput Aware Scheduling

4.1 Introduction

Packet scheduling remains an important research area in TDMA networks. The scheduling algorithm has to address issues such as throughput, fairness, QoS constraints and channel utilization ratio among others. In this chapter, we only consider throughput and leave QoS provisioning for chapter 5. Also, we do not consider fairness since SSs are granted their full bandwidth requests. Instead, the goal of the scheduling algorithm is to find a minimum length schedule for uplink traffic. Finding the minimum number of centralized scheduling transmission opportunities in the data sub-frame is an important provisioning question for 802.16 mesh networks [17]. Most researchers translate the problem into a maximum clique problem where the goal is to find the maximum number of concurrent interference-free transmissions. The maximum clique problem is however NP hard, hence a quicker solution is more desirable. Hence we propose a sub-optimal heuristic for the scheduling problem.

In this chapter, we discuss the 802.16 scheduling protocol in detail. First we describe the 802.16 mesh control packets used by the BS and SSs to negotiate end-to-end bandwidth, then we propose a model for bandwidth assignment. We evaluate our model through extensive simulation and compare it with other schemes proposed in the literature.

4.2 Bandwidth Request Mechanism

In WiMAX mesh network, SSs having data to transmit need to request for bandwidth from the BS through several ways. These include requests, grants, UGS (see section 5.2), polling etc. Vendors are allowed to employ different combinations of these schemes to optimize performance. Requests refer to mechanisms that SSs use to indicate to the BS that they require uplink allocation of bandwidth. Polling on the other hand is the process where the BS allocates bandwidth to an SS specifically for making bandwidth requests. Polling can be for a single SS (unicast polling) or a group of SSs (multicast polling).

Two control messages, MSH-CSCH and MSH-CSCF are used for coordination in centralized scheduling. The BS uses MSH-CSCF packets to distribute the network topology and MSH-CSCH packets to assign end-to-end bandwidth [10]. SSs request end-to-end bandwidth from the BS by sending MSH-CSCH packets to their parent nodes. When all the requests reach the BS, the BS uses the information to make a schedule and multicasts the schedule to SSs with new MSH-CSCH packets. The BS makes the schedule to fulfill bandwidth requests of all SSs. If the end-to-end bandwidth requests of SS are not fully satisfied, then it should result in a fair end-to-end bandwidth assignment.

4.3 Problem Definition

We model the mesh network as a directed graph $G = (V, E)$ where V denotes the set of nodes (BS and SSs) and E the set of directed communicating links between the nodes. We are also given the route from every SS to the BS and we represent the traffic demand of each SS as a discrete number of packets. Each node can only transmit one packet in each transmission opportunity. The goal of the centralized scheduling is to find a minimum length schedule for all data to reach the BS while satisfying the constraint that only non-interfering links can be scheduled to transmit concurrently.

4.4 Related Work

Several works [16, 17, 18, 23] have been proposed for multichannel single transceiver WiMAX centralized scheduling. The goal of these works is to maximize the number of concurrent transmissions in the network. In mesh network, links with higher number of children may accumulate traffic and become the bottleneck of the network hence [16] proposes a maximum degree first select (MDFS) algorithm which assigns channels to links in the order of decreasing degree (number of children). A nearest first select (NS) algorithm for uplink traffic is presented in [17]. The goal is to minimize the length of time needed for all data transfer. In this work, links are processed in the order of their closeness to the BS. Jun Xiao *et al.* [18] extend the work of [18] by equipping each node with two transceivers operating on different frequencies. Empowering nodes with multiple transceivers results in high throughput but also increases significantly the cost of deployment. In all these works, the authors do not employ any efficient channel assignment scheme. Instead they choose the minimum index channel from the set of available channels. In this work, we employ a novel technique for channel assignment. Given list of channels, we select the channel that causes the least interference to the links in the neighbourhood.

Raniwala *et al.* [24] propose a load-aware joint routing and scheduling for 802.11 base mesh networks. Their approach iteratively performs routing and channel assignment until no significant improvements are made. This scheme cannot however be applied directly to 802.16 WiMAX mesh because of the difference in technology. In [23], links are processed in the order of decreasing traffic demands while an optimal scheduling based on integer linear program is presented in [25]. A dynamic clique based link scheduling is presented in [26]. The essential idea here is to find the maximum number of non-interfering links to transmit concurrently while enforcing fairness. Finding a maximum clique is a hard problem and takes a lot of time. In WiMAX mesh network, there is limited time to perform the resource allocation hence a quicker solution is more desirable.

In TDMA scheduling, a major source of delay occurs when a packet arriving at an intermediate node must wait before being forwarded. To overcome this, Djukic *et al.* [14] propose a delay aware link scheduling. The scheduling delay is directly related to the transmission order so the authors present a $\{0,1\}$ integer formulation that finds a transmission order for which the maximal scheduling delay among all paths is minimized. Several works [27, 28, 29, 30] have been proposed for distributed Wimax scheduling. However, centralized scheduling is more efficient and suits well for multihop networks

4.5 Throughput Aware Scheduling

WiMAX centralized scheduling consists of uplink and downlink phases. For ease of exposition and without loss of generality, we consider the uplink phase. Since each link direction is a mirror of the other [16, 17], our model can be easily applied to the downlink phase as well. The resource allocation problem can be viewed as two phases: centralized scheduling and channel assignment. The scheduling algorithm determines which links are scheduled to transmit in a given timeslot while the channel assignment algorithm determines which channel to assign to a given link. We present details of these algorithms in the next sections.

4.5.1 Centralized Scheduling

The scheduling algorithm has to select a subset of non-interfering links to transmit in each timeslot. When a link is scheduled to transmit, its traffic demand decreases by one unit while the traffic demand of its parent increases by one unit. The goal is thus to minimize the number of timeslots required for all data to reach to BS. Due to primary interference, the BS can only receive one packet in each timeslot. This means for a network with a total of n packets, a trivial lower bound is n timeslots for all data to reach the BS. To achieve this bound means the BS has to receive data in each timeslot and this implies there must be data available at the one hop links in each timeslot. This is the central idea of our scheduling scheme. We aim to keep the BS busy in each timeslot while ensuring that there are always packets available at the one hop links to be forwarded to the BS. We define a k-hop link as a link that is k links away from the BS. A one-hop link is a link that has the BS as its destination.

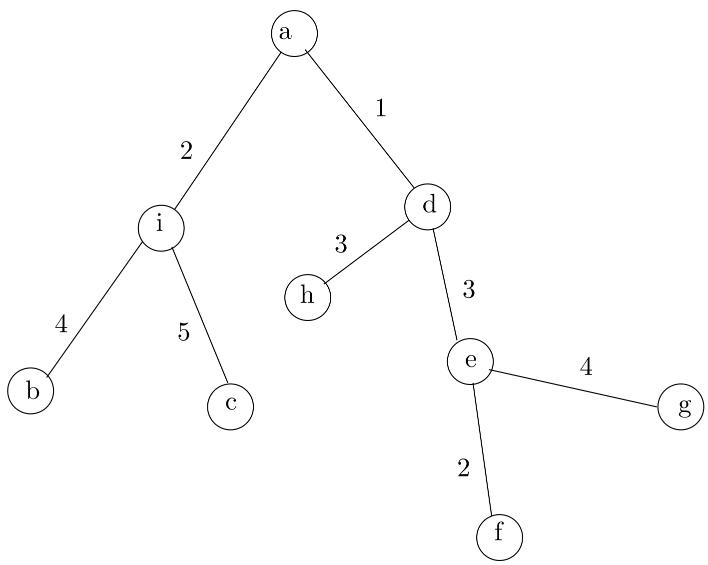


Figure 4.1: Example scheduling

In each timeslot, our scheduling algorithm first schedules a one hop link to transmit. In particular, we choose the one hop link whose children have the least amount of traffic. This is to allow the remaining one hop links to be able to receive data from their children without violating primary interference. As an example, consider the diagram shown in figure 4.1 with traffic demands indicated on the links. Assume node a is the BS. First we have to choose between links ia and da. For link ia, the total traffic for its children is $4+5=9$ while that for da is $3+3=6$. Link da will thus be selected since its children have the least traffic. In this way, it is possible for link *ai* to receive data from one of its children without violating primary interference. Next, we check to see if there are at least two packets available at the one-hop links. If this is not the case, we schedule the two hop link with the highest amount of data that does not conflict with the already scheduled link. This ensures that in the next timeslot, we have data available at the one hop links to be forwarded to the BS. From the diagram, if the traffic on link *ia* were zero, then link *ci* (with the highest traffic demand) will be chosen to transmit. Anytime a link is selected, we remove all its children from the candidate set of links because of primary interference. We define a metric p of a link as hopcount of a link multiplied by the data traffic. For example link fe has $p = 3 \times 2 = 6$ while that of ge is $3 \times 4 = 12$. We then process the remaining links in order of decreasing p. Details of the scheduling algorithm is presented in Algorithm 3. The outer loop in the algorithm chooses timeslots. In each timeslot, the set A holds the links that have data to transmit. From this set, the algorithm then finds a subset of links to be scheduled denoted as L_u based on the approach outlined above. Each time a link is selected, the channel assignment function detailed in Algorithm 4 is called to assign a channel to the link. If an available channel is found for the link, its children links are removed from the candidate set to account for primary interference.

4.5.2 Channel Assignment

In this section, we describe a novel approach we use to assign channels to links. The goal is to have more links transmit concurrently. Consider C_a as the available channel set and C_i as the set of channels that link i can transmit without causing any interference. At the start, each link can transmit on any channel. Assume link l is scheduled to transmit on channel c. This means all other links that interfere with l denoted as $S(l)$ cannot be allowed to transmit on c. As a result, all interfering links will have c removed from their available channel list.

Algorithm 3 Throughput Aware scheduling

Input: $G = (V, E)$, traffic demands
Output: schedule

$t \leftarrow 1$ // t is timeslot
while exists any $demand(j) > 0$ for any SS_j **do**
  $A \leftarrow$ set of links that have data to transmit
  $L_u \leftarrow \oslash$
  $s \leftarrow$one-hop link with data and whose children have least traffic
  $A \leftarrow A - \{s\}$
  if Channel Assignment(s) is true **then**
    $L_u \leftarrow L_u \cup \{s\}$
    $A \leftarrow A -$ children(s)
  end if
  if demand(one-hop links)< 2 **then**
    $l \leftarrow \underset{i \in A : hopcount(i)=2}{\text{argmax}}$ demand(i)
    $A \leftarrow A - \{l\}$
    if Channel Assignment(l) is true **then**
      $L_u \leftarrow L_u \cup \{l\}$
      $A \leftarrow A -$ children(l)
    end if
  end if
  while $A \neq \oslash$ **do**
    $l \leftarrow \underset{i \in A}{\text{argmax}}$ p(i)
    $A \leftarrow A - \{l\}$
    if Channel Assignment(l) is true **then**
      $L_u \leftarrow L_u \cup \{l\}$
      $A \leftarrow A -$ children(l)
    end if
  end while
  adjust demands for each $i \in L_u$
  $t \leftarrow t + 1$
end while

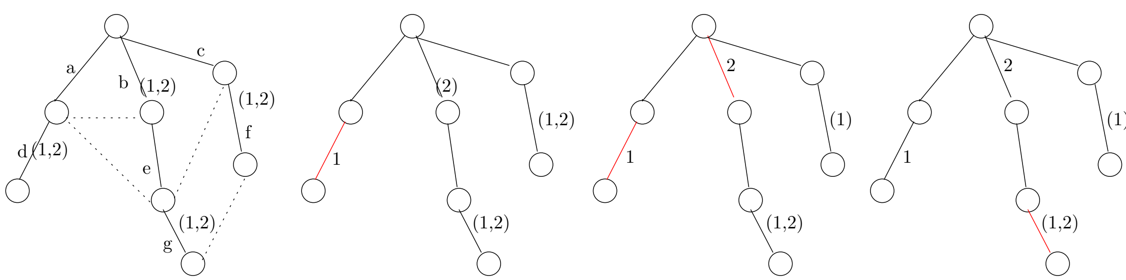
If l transmits on c and this causes n links to remove c from their channel list, we refer to n as the interference degree of c.

Definition 3 *Interference degree (In) of a channel c with respect to a link l is the number of links that are blocked from transmitting on c as a result of l transmitting on c.*

To assign a channel to a link, the goal is to choose the channel that has the least interference degree In. The channel assignment algorithm is presented in Algorithm 4.

Algorithm 4 Channel Assignment

Input: l // link to be assigned channel
Output: channel assignment
  if $C_l \neq \oslash$ **then**
    $c(l) \leftarrow \underset{ch \in C_l}{\text{argmin}}\, In(ch)$
    $C_i \leftarrow C_i - \{ch\} \forall i \in S(l)$
    return true
  else
    return false
  end if



(a) (b) (c) (d)

Figure 4.2: Example channel assignment

Consider the diagram shown in Figure 4.2 with the dotted lines indicating the interference between links. Assume we want to assign channels to the links d,b,g and f and that there are only two channels available. At the start, each link has in its link set two channels as indicated on Figure 4.2(a). When link d is considered, we have to choose between channel 1 or two. Since both channels have the same interference degree (1), we will assign channel 1 to link d. As a result, we remove channel 1 from the channel set of link b as indicated on Figure 4.2(b). Next, we consider link b and since there is only one channel available, channel 2 will be assigned. We then remove channel 2 from the channel set of link f as shown in Figure 4.2(c). The next link to be considered is link g which still has two channels in its channel set as shown in Figure 4.2(d). At this point, assigning channel 1 to link g will result in the channel set of link f being empty. However, since the interference degree of channel 2 is the least(zero), we will assign channel 2 to link g. This will allow link f to be assigned channel 1.

4.6 Experiments

In this section, we evaluate our scheduling framework through extensive experiments using random as well regular square grid graphs. The experiment set up is the same as described in section 3.5.1. In each experiment, we compute routes using ILMR and then perform scheduling using throughput aware scheduling (TS), maximum degree first select (MDFS) and nearest select (NS) [17]. We also record a trivial lower bound which is the total number of packets in the network. We compare the schemes using metrics such as length of schedule and channel utilization ratio.

4.6.1 Length of Schedule

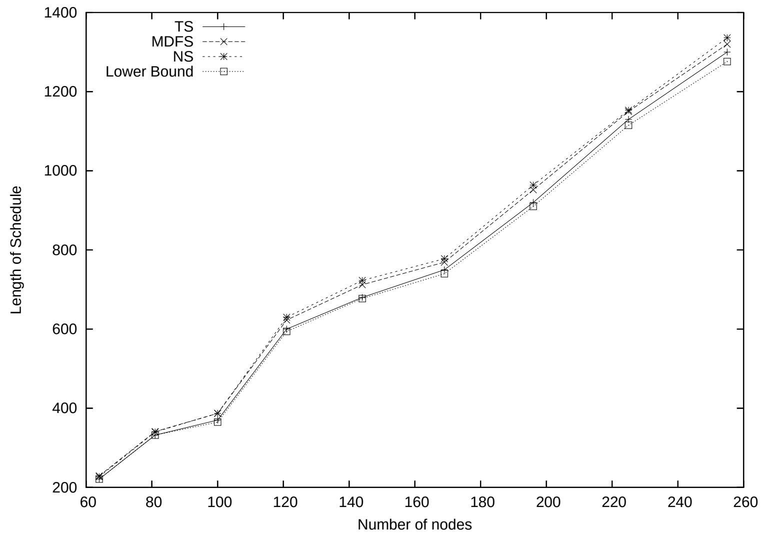
We compare the length of schedule produced by the different scheduling schemes with limited and unlimited number of channels. Figure 4.3 shows the results obtained using just one channel. The results indicate that TS recorded the least length of schedule and outperformed MDFS and NS in all instances. With a small number of nodes, TS produced the same results as the bound. As the number of nodes increases, TS produced results about

95% of the lower bound in the regular graphs and about 92% in the random graphs while MDFS and NS achieved approximately 90% and 87% respectively in the regular graphs. For random graphs, MDFS achieved about 88% while NS achieved 87% of the bound

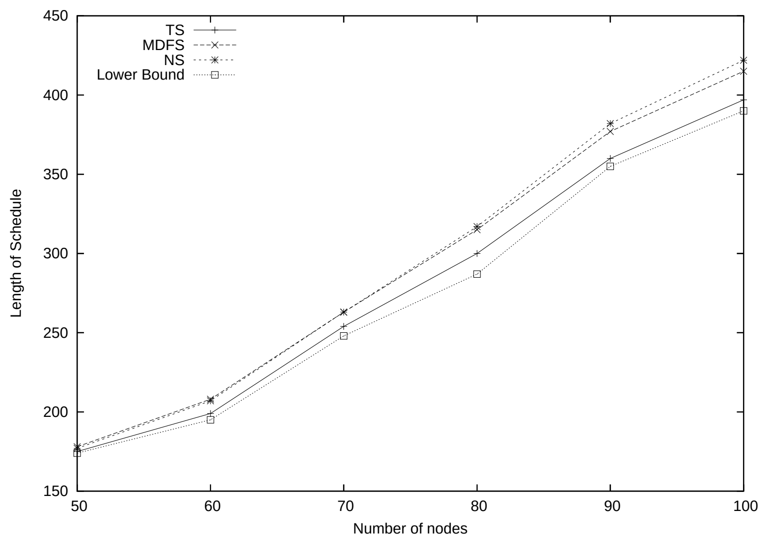
For the case of unlimited number of channels, there was no much difference in the performance recorded by the different algorithms. TS still produced the best results and was about 98% close to the bound.

4.6.2 Channel Utilization Ratio

We record the channel utilization ratio on different graphs for the different algorithms as shown in Figure 4.5. The results follow the same trend. TS outperformed MDFS and NS in all instances. MDFS slightly outperformed NS when the number of nodes was higher.



(a) regular graphs with 1 channel



(b) random graphs with 1 channel

Figure 4.3: Limited number of channels

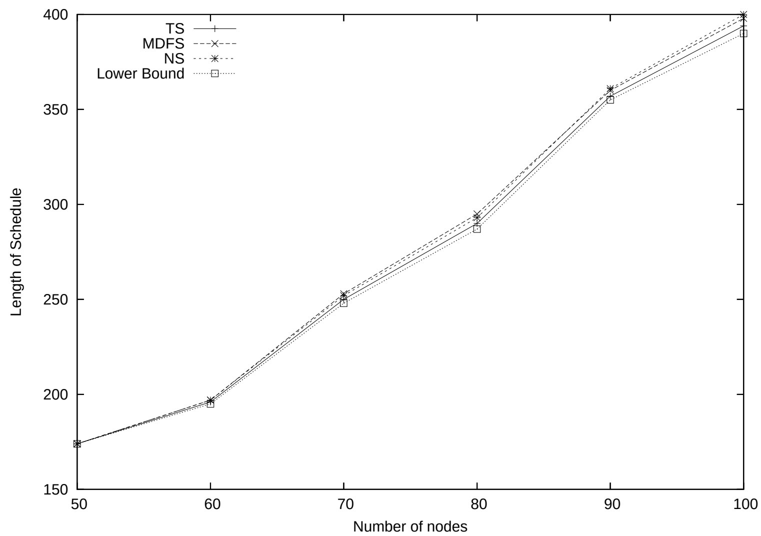


Figure 4.4: Random graphs with Unlimited number of channels

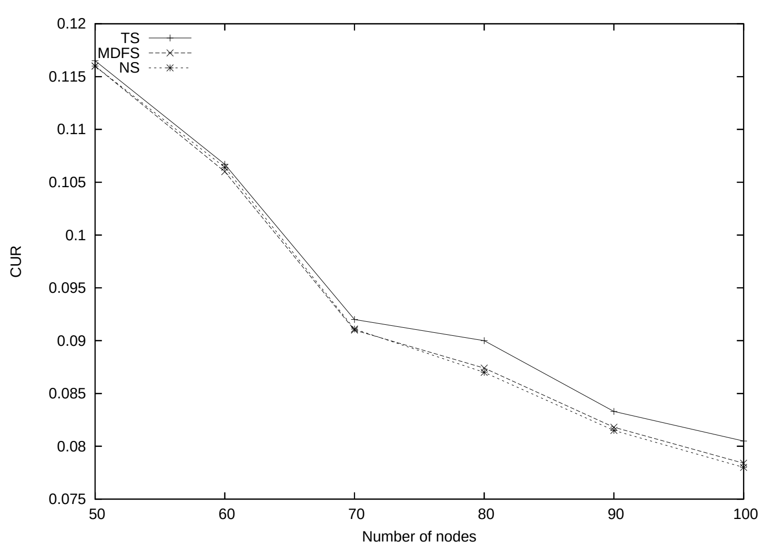


Figure 4.5: Number of nodes vs CUR

Chapter 5
Quality of Service

5.1 Introduction

In this section, we present a model for another important design challenge in WiMAX mesh networks known as QoS provisioning. In recent times, there has been a tremendous increase in the types of applications in internet protocol (IP) networks. Besides the traditional file transfer, email and web browsing, multimedia applications are also becoming increasingly popular. These applications send large amounts of audio and video streams with variable bandwidth and delay requirements. For example, an email application may not need any guarantee except reliable delivery of the message. A VoIP application on the other hand will require low latency (delay) while a video streaming application may tolerate long delay but require relatively high bandwidth. Accommodating all these will require QoS guarantees. QoS provisioning in mesh networks is thus a very important ingredient towards the vision of globally connected heterogeneous networks.

QoS provisioning is a challenging task faced by the research community. Wireless networks are generally less efficient and unpredictable compared to wired networks, which makes QoS provisioning a bigger challenge for wireless communication [4]. For instance, the wireless medium is often characterised by limited bandwidth and high packet error rate which together limit the capacity of the network to provide QoS guarantees.

5.2 WiMAX QoS Specification

There is no formal definition for QoS although several standards have proposed different definitions. In the field of telephony, the International Telecommunication Union (ITU) defines QoS as a set of quality requirements on the collective behavior of one or more objects. This definition lists six primary components: support, operability, accessibility, retainability, integrity and security [31]. The ITU again defines QoS in the field of data networking as the probability of the telecommunication network meeting a given traffic contract. QoS provisioning encompasses providing Quality of Service to end users in terms of several generic parameters [31]. In WiMAX, such parameters include throughput, average delay, average jitter and packet loss.

1. Throughput is a measure of the data rate (bits per second) generated by the application.

$$TP = \frac{\sum_{i=1}^{n} PacketSize_i}{PA_n - PS_0} \tag{5.1}$$

   Equation 5.1 shows the calculation of throughput TP, where $PacketSize_i$ is the packet size of the ith packet reaching the destination, PS_0 is the time when the first packet left the source and PA_n is the time when the last packet arrived.

2. Average delay or latency is the time taken by packets to travel from source node to destination node. The principal sources of delay are source processing delay, propagation delay and destination processing delay. The calculation of delay is show in equation 5.2 where PA_i is the packet arrival time of the ith packet, PS_i is the packet start time and n is the total number of packets.

$$AverageDelay = \frac{\sum_{i}(PA_i - PS_i)}{n} \tag{5.2}$$

3. Jitter is the variation in the delay introduced by the components along the communication path [31]. It is the variation in time between packets' arrival. Jitter gives a measure of the consistency and stability of a network. Equation 5.3 shows the

calculation of jitter.

$$AverageJitter = \frac{\sum_{i=1}^{n}((PA_{i+1} - PS_{i+1}) - (PA_i - PS_i))}{n-1} \tag{5.3}$$

4. Packet loss affects the perceived quality of the application. Several causes of packet loss or corruption would be bit errors in an erroneous wireless network or insufficient buffers due to network confestion when the channel becomes overloaded [31]. Equation 5.4 shows the calculation of packet loss.

$$PacketLoss = \frac{\sum LostPacketSize_i}{\sum PacketSize_j} \times 100 \tag{5.4}$$

5. Traffic Priority- this parameter specifies the priority assigned to a service flow. Given two service flows identical in all QoS parameters besides priority, the higher priority service flow should be given lower delay and higher buffering preference [2].

Among the numerous proposals brought forward by WiMAX, perhaps the most attractive is its ability to deliver Asynchronous Transfer Mode (ATM) like connection oriented QoS guarantees in broadband wireless networks. This is due to the increase in the demands for multimedia content delivered wirelessly. WiMAX is designed to support a wide range of applications which require different levels of quality of service. To accommodate these applications, the IEEE 802.16 standard [2] defines four different classes of traffic: unsolicited grant service (UGS), real-time polling service (rtPS), non-real-time polling service (nrtPS) and best effort service (BE).

- UGS

  This service class is designed to support real-time applications that generate fixed data packets on periodic basis, such as T1/E1 and VoIP without silence suppression

[2]. The BS allocates fixed number of slots to UGS at periodic intervals regardless of current estimation backlog. UGS requests are granted bandwidth without polling or contention. These requests have the most stringent QoS requirements.

- rtPS

  rtPS class supports real-time applications that generate variable size data packets on a periodic basis, such as moving pictures experts group (MPEG) streaming video. The BS provides periodic dedicated request opportunities for SSs to meet the applications real-time demands [32]. Unlike UGS, rtPS connections have to always notify the BS of their current bandwidth requirements.

- nrtPS

  nrtPS class is designed to support delay tolerant data stream and consists of variable sized data packets which require a minimum data rate. An example of nrtPS application is FTP.

- BE

  This service class is designed to support data streams for which no minimum service level is required and therefore may be handled on a space-available basis, such as HTTP. The SS is allowed to use contention request opportunities as well as unicast request opportunities for BE service requests.

In addition to the definition of the different service classes, the standard also specifies QoS parameters for each service class as illustrated on Table 5.1.

Table 5.1: WiMAX QoS Parameters

Service Class	Qos parameter				
	Minimum Rate	Maximum Rate	Latency	Jitter	Priority
UGS		X	X	X	
rtPS	X	X	X		X
nrtPS	X	X			X
BE		X			X

5.3 Related Work

Several works [16, 17, 14, 18] have been proposed for centralized scheduling and channel assignment in WiMAX networks. While [16, 17, 14] consider single transceiver with multiple channels, [18] consider multi-transceiver and multiple channels. The goal of these algorithms is to minimize the length of schedule. None of these however takes into consideration the different WiMAX service classes. Several generalized schedulers like Fair Scheduling [33], Earliest Deadline First (EDF) [34, 35], Weighted Round Robbin (WRR) [36, 37] and Deficit Round Robbin (DRR) [38, 39] exist in the literature. However, these algorithms cannot be used directly due to the peculiarities of the WiMAX technology. Some authors propose priority scheduling schemes like Deficit Fair Priority Queue (DFPQ) [40], Preemptive Deficit Fair Priority Queue (PDFPQ) [41] and Random Early Detection Deficit Fair Priority Queue (RED-DFPQ) [42] for resource allocation for the different service classes. These algorithms maintain a separate queue for each service class and assign different priority to these queues. The drawback of these schemes is that they do not guarantee QoS requirement of higher priority service classes like UGS and rtPS. In [43], the authors present a load balancing routing scheme. Their scheme considers all alternative paths to determine a feasible route for a request. The authors then propose a call admission control (CAC) (see section 5.4) module for UGS and rtPS service classes. A route is partitioned into two parts. The first part consists of the path from source node to the penultimate node while the second part consists of the path from penultimate node to the destination node. The essential idea is to schedule links in the first part to meet delay while the second part takes care of jitter constraint. A grant per subscriber station (GPSS) scheduling scheme is presented in [44]. In this work, a BS scheduler first grants bandwidth to each service class based on aggregate bandwidth request while an SS scheduler further allocates the resources of each service class among its connections. This increases the complexity of SSs since they require an additional scheduler.

Ghosh *et al.* [45] propose an interference-aware call admission control to guarantee delay constraint. This work does not however consider jitter requirement of UGS service class. In [46, 32, 47], the authors present various call admission schemes but do not provide any implementation details. For example, the authors do not specify details of how slots are assigned to links in an interference aware manner. A distributed hop-by-hop admission control and route discovery is proposed by Cheng *et al.* [46]. In this scheme, when a new connection arrives, each node on the path makes its own admission control decision based on residual bandwidth in its interfering neighborhood. A node computes its residual bandwidth by identifying the maximal clique constraints in its local conflict graph.

5.4 Call Admission Control (CAC)

Call or Connection Admission Control (CAC) plays a crucial role in QoS provisioning. It refers to a network's QoS mechanism that determines whether a new connection with given QoS requirement can be established. WiMAX is connection oriented, which means an SS must register a connection with the BS before it can transmit. When a new request arrives at the SS, the SS has to set up an end to end connection with the BS. Depending on the current bandwidth utilization and the QoS requirement of the new request, the CAC module will decide whether to accept or reject the connection, and how much bandwidth should be set aside for the connection.

There are two ways in which BS grants bandwidth: grant per connection (GPC) and grant per subscriber station (GPSS). In the former, the BS grants the requested bandwidth explicitly to the connection while in the latter, the BS grants a whole bandwidth to the SS and an additional scheduler at the SS must distribute the resources among its service flows to maintain QoS. GPSS is more efficient and scalable since less information is sent to the BS. The benefit of using GPC is that it reduces the complexity of SSs since SSs require no additional schedulers. Our scheme is independent of whether the BS allocates resources per subscriber or per connection.

5.5 Our proposed Scheme

The IEEE 802.16 standard provides specifications for QoS guarantees in the MAC layer, however, no implementation details are specified and the task is left to vendors to design various schemes. This has become an active area of research. Providing QoS guarantees in WMNs poses several challenges due to interference inherent in multihop transmission. The scheduling algorithm in addition to maximizing the network's throughput has to ensure that QoS requirements of all accepted connections are met. Furthermore, the scheduling algorithm only has a small amount of time to produce an optimal schedule since the length of a frame is typically about 10ms. For example, one of the configurations in the standard specifies 400 frames per second [2], this means the scheduling algorithm has to produce an optimal schedule 400 times in a frame. In this work, we propose a simple and efficient heuristic link scheduling which is also used for CAC for new requests. We first present two routing schemes that find paths for requests and compare our schemes to shortest path routing and the scheme proposed in [23]. Our routing schemes find less congested routes with less interference. This load sharing mechanism ensures that requests meet their deadlines.

5.5.1 Routing

In this section, we extend our routing schemes in Chapter 3 to support QoS provisioning. We investigate the effects of routing on QoS provisioning.

Interference and Load Aware Routing (ILR)

In this scheme, we construct routes with less interference while ensuring load sharing as illustrated in section 3.4.1. ILR is a tree base routing that is computed at the beginning of every frame to adapt to the traffic demands of the network. The route of a request is then obtained from the routing tree. Given a set of admitted requests, we compute the demands of all nodes in the network. The demand of a node is computed as the sum of the minimum bandwidth of all accepted requests for which it is the source node. Since BE requests do not have minimum requirements, we do not consider them in computing the demand of nodes. The new routing tree stays in effect for one frame after which a new tree is constructed in the next frame. The routes of requests belonging to UGS, rtPS and nrtPS thus change from frame to frame. However, when a BE request is admitted, its route is fixed throughout the life time of the request. This is because, in our CAC algorithm, BE packets may take several frames to reach the BS so changing the route may result in packets reaching the BS out of sequence.

Interference and Load Aware Multipath Routing (ILMR)

The drawback to tree base routing is that some links are never used for communication in the network. To effectively ensure load sharing, we employ multipath routing that explores alternative routes for requests. This scheme is similar to the one outlined in section 3.4.2.

The difference here is that we are computing paths for connections and not just the nodes. In this scheme, we define load degree (*Load*) of a node to be the amount of traffic that passes through the node and blocking metric of a path as the sum of blocking values as well as load degrees of all nodes on the path. When a request is considered, we choose the route with the least blocking metric. The route is fixed throughout the life time of the requests. After a request is accepted, we increase the load degrees of all nodes on the route by $R/Rmax \times \varepsilon_2$ where R is the minimum bandwidth requirement of the request and $Rmax$ is the highest of the minimum bandwidth of all accepted requests. For example, in Figure 5.1, if the selected path is $c - a - BS$, then we update the load degrees of nodes c and a by adding the quantity $R/Rmax \times \varepsilon_2$ and this will affect the load degree of b which will also need to be updated by the same quantity. This update will change the blocking metrics of c, a and b. When a request exits, we also decrease the load degree of nodes on its route to reflect the decreased traffic.

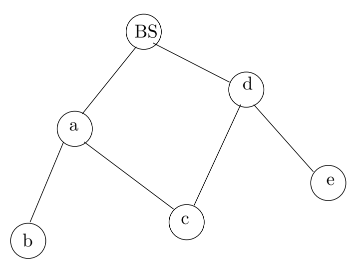


Figure 5.1: Updating blocking metrics

Updating the load degree of nodes is a crucial part of this algorithm since we are combining two metrics- interference and traffic load. Updating with a bigger quantity leads to longer routes which will result in more requests being rejected. We tried different quantities and we obtained the best results by setting ε_2 to 0.02. This quantity is however specific to our experimental set up but can be easily modified to handle any configuration with different parameters like number of subchannels and bandwidth requirements of requests.

Algorithm 5 ILMR Algorithm

$S \leftarrow \{1\}$ set of selected nodes (starts with base station)
$N_s \leftarrow \{2, 3, ..., N\}$
$W \leftarrow Neighbour(1)$
while $N_s \neq \oslash$ **do**
  $(i, j) \longleftarrow \underset{j \in S,\, i \in Neighbour(j) : i \in W}{\text{argmin}}\, B(i) + Load(i) + Bk(j)$
  Add i to S
  $Parent(i) = j$
  $Bk(i) \longleftarrow Bk(j) + B(i) + Load(i)$
  $W \leftarrow W \cup Neighbour(i)$
  $N_s \leftarrow N_s - \{i\}$
end while
If the request is accepted, update *Load* of nodes on its route
When request exists, update *Load* of nodes on its route.

5.5.2 Link Scheduling and Channel Allocation with QoS constraints

In this section we present our slot allocation algorithm which will find a feasible slot assignment for requests to satisfy their QoS constraints. This module is first used as a CAC for new requests and for assigning slots for requests at the beginning of each frame. When a new requests arrives, the CAC module first computes its path based on the routing schemes presented in the previous section. The new request together with all existing requests go through a slot assignment scheme. Requests are processed in the order of priority i.e. UGS, rtPS and then nrtPS. Our link scheduling and channel allocation procedure performs CAC while finding a feasible slot assignment for the new and existing requests to satisfy minimum bandwidth requirement. If the QoS requirements of the new request as well as existing requests are met, the request is accepted, otherwise it is rejected. UGS, rtPS and nrtPS are guaranteed minimum end to end bandwidth allocation in each frame. After all new requests have been considered, the slot assignment module allocates extra slots to rtPS and nrtPS requests until there are no more slots for ent-to-end allocation.

First we translate the bandwidth requirements of requests into slots in a way similar to the approach in [47]. Let B_i denote the bandwidth requirement of the i^{th} request and S_i the number of bytes request i can send in one slot. For ease of exposition and without loss of generality, we assume that SSs use the same coding and modulation scheme and that channel condition remains the same on all channels. The number of slots required in each frame is obtained from the expression:

$$N_i = \left\lceil \frac{B_i}{S_i F} \right\rceil \tag{5.5}$$

where F stands for the number of frames per second. Equation (5.5) essentially translates the bandwidth requirement of a request into the appropriate number of slots needed for each frame. Given a path from source node to the BS, our CAC algorithm assigns slots to the links in the reverse direction i.e. from BS to the source node. The central idea here is to assign slots to the last link on the path to satisfy jitter and/or latency requirements and then back track to the source [45]. We introduce the concept of feasible interval as the interval of timeslots within which a link should be scheduled to meet the QoS requirement of a request. For UGS requests, we denote by d_{min} the earliest time the last link could be scheduled to satisfy jitter requirement. The difference in slot assignment for UGS, rtPS and nrtPS is the way we compute the feasible slot interval for the last link. The order of requests is UGS, followed by rtPS and then nrtPS. For requests belonging to the same class, the order is arbitrary. The CAC module consists of three parts.

1. First we perform end to end slot allocation to requests belonging to UGS, rtPS and nrtPS to satisfy minimum bandwidth requirement. End to end slot allocation means we allocate slots to all links on the path of a request to ensure that all packets reach the BS before the end of the frame.

2. Next, we perform end to end slot allocation to requests belonging to rtPS and nrtPS to satisfy extra demands in a round robbin approach. We process these requests in order of priority ie rtPS followed by nrtPS. UGS requests are admitted at their maximum bandwidth so we do not assign any extra slots to these requests. In each round, each request is assigned one end to end slot. This continues until there are no more end to end slots available.

3. Since BE requests do not have minimum requirements, We do not allocate end to end slots to these requests. Instead, we take advantage of the slots that remain after end to end slot allocation and assign these to BE requests.

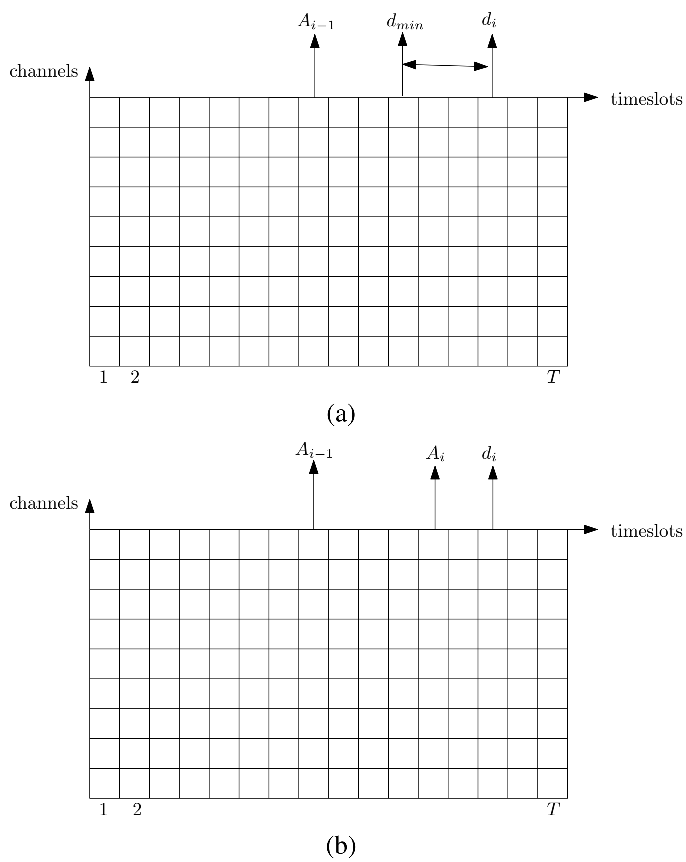


Figure 5.2: Slot Assignment

We denote by B_i^{min} and B_i^{max} the minimum and maximum bandwidth requirements of the i^{th} request respectively. Let N_i^{min} denote the minimum number of slots for the i^{th} connection, N_i^{max} the maximum number of slots and R_i the requested bandwidth, then we can compute slots for different requests as follows:

- UGS requests do not participate in polling and contention. Once admitted, they are always allocated their maximum bandwidth requirement since they do not have minimum bandwidth requirement. UGS slots are calcuated as shown in equation (5.6)

$$N_i^{min} = N_i^{max} = \left\lceil \frac{B_i}{S_i F} \right\rceil, \forall \text{ UGS requests} \tag{5.6}$$

  UGS requests have the most stringent QoS requirements so they are assigned slots first. For each packet of a UGS request, we compute the deadline timeslot by which the packet should reach the BS. Consider the frame shown in Figure 5.2. Assume the deadline of the i^{th} packet of a UGS request is d_i and A_{i-1} is the timeslot the $i-1^{th}$ packet of the request reached the BS as illustrated in Figure 5.2(a). Assume the path of the connection is $i-j-BS$. First we find a slot assignment for link $j-BS$. To do this, we compute the feasible slot interval for this link as shown on Figure 5.2(a), d_{min} is determined by jitter and it is computed as $d_{min} = jitter + delay_{i-1} + s_i$ where s_i is the generation timeslot of the i_{th} packet and $delay_i = A_i - s_i$. We greedily choose the latest available time slot in this interval so that there is enough time to schedule the remaining links. After we choose a time slot, channel assignment is done in the same way as illustrated in section 4.5.2. Assume A_i is the chosen timeslot as shown in 5.2(b), next, we consider link $i-j$ and we search the feasible slot interval $1-A_i-1$ for an available timeslot. This continues until slots have been assigned for all links and for all packets of the request. If we reach the beginning of the frame and cannot find any available timeslot, then the request cannot be scheduled.

- rtPS requests specify a minimum and maximum bandwidth. They are only guaranteed minimum bandwidth but when they request for higher bandwidth, they could be given extra slots depending on the availability of resources. For rtPS requests, the requested bandwidth R_i is between B_i^{min} and B_i^{max}. The minimum and maximum number of slots of requests belonging to this class are calculated as follows:

$$N_i^{min} = \left\lceil \frac{B_i^{min}}{S_i F} \right\rceil \tag{5.7}$$

$$N_i^{max} = min\left\{ \left\lceil \frac{B_i^{max}}{S_i F} \right\rceil, \left\lceil \frac{R_i}{S_i} \right\rceil \right\}, \tag{5.8}$$

$$\forall \text{ rtPS requests}$$

Slot assignment for rtPS is similar to UGS. The only difference is that rtPS requests do not have jitter requirement, so to assign a slot to the last link, we only satisfy the delay constraint of the packet. Feasible slot interval for the last link is computed as $[A_{i-1}, d_i]$. rtPS requests are admitted at minimum bandwidth requirement. When packets arrive at a rate higher than the granted rate, the packets are queued at the source node. When a packet does not reach the BS by its deadline, it is dropped. Queueing may influence the behavior of the scheme as packets will be dropped when the queue is full, but in this study we isolate the effects of routing and link scheduling alone only so we assume infinite queue. We can deal with finite queues by simply imposing a limit on the queue length.

- nrtPS requests unlike rtPS are delay tolerant. The resource allocated to these requests might be less than the minimum bandwidth requirement [44]. The minimum and maximum number of slots are calculated as follows:

$$N_i^{min} = min\left\{ \left\lceil \frac{B_i^{min}}{S_i F} \right\rceil, \left\lceil \frac{R_i}{S_i} \right\rceil \right\} \tag{5.9}$$

$$N_i^{max} = min\left\{\left\lceil\frac{B_i^{max}}{S_iF}\right\rceil, \left\lceil\frac{R_i}{S_i}\right\rceil\right\}, \tag{5.10}$$

$\forall$ nrtPS requests

Since nrtPS requests do not have latency and jitter requirements, to find an available slot for the last link, we start searching from the last time slot of the frame. The feasible slot interval for the last link is computed as $i - T$ where T is the last timeslot of the frame. The goal is to schedule all packets to reach the BS in the current frame.

- BE requests do not have any minimum requirements so we do not reserve any slots for these requests. The maximum number of slots for BE requests is calculated as:

$$N_i^{min} = 0 \tag{5.11}$$

$$N_i^{max} = \left\lceil\frac{R_i}{S_i}\right\rceil \tag{5.12}$$

$\forall$ BE requests

BE requests have not minimum requirements so they are always accepted without going through CAC. Each link keeps a queue for BE packets and the remaining slots are used to forward BE packets. Because BE packets may take several frames to reach the BS, their routes are fixed in all our routing schemes.

Details of our CAC algorithm is illustrated in Algorithm 6. The CAC module uses the procedure *Slot_Assignment* shown in Algorithm 7 to allocate slots to a request. If there is a feasible slot assignment for the new request as well as all existing requests, the request is accepted, otherwise it is rejected. Slot assignment is done in an interference free manner and ensures that requests meet their QoS requirements. Two links can be assigned the same slots if they do not interfere with each other. A slot here refers to a slot in time and frequency domain while timeslot refers to a slot in time domain

Algorithm 6 CAC

Input: $G = (V, E)$, existing requests,new request.
Output: schedule

```
Compute path of new request
arrange all requests in order of priority (existing requests and the new request)
for each request i do
    if Slot_Assignment(i) is false then
        reject new request
        Exit
    else
        update temporal schedule.
    end if
end for
accept new request
save schedule
```

Algorithm 7 *Slot_Assignment*

Input: request
Output: channel assignment

```
path ← path of request (links arranged in reverse order)
packets ← minimum bandwidth of request
while packets > 0 do
    temp ← path
    for each l in temp do
        compute feasible slot interval
        Find an available slot from feasible slot interval
        if no available slot then
            return false
        end if
    end for
end while
return true
```

5.6 Experiments

5.6.1 Introduction

In this section, we perform extensive simulation to evaluate the algorithms proposed in this chapter. We compare our routing schemes to the routing scheme proposed in [23] which we label as IR. We compare the algorithms using several metrics like throughput, acceptance rate and packet drop rate.

5.6.2 Simulation Setup

We used a custom simulator written in C++ to evaluate our proposed scheme. We performed the experiments using a 25 node regular grid graph. The arrival of requests follows a Poisson distribution with mean arrival rate of λ. For each request, the source node and service class are uniformly distributed. The life time of requests is exponentially distributed with mean life time of 1000 frames. The bandwidth request size of rtPS and nrtPS is uniformly distributed between the minimum and maximum bandwidth and its duration follows an exponential distribution with mean of 20 frames. The simulation was performed for 5000 frames and we assume the length of each frame to be $10ms$. We use 50 subchannels and 200 slots in the data subframe. The QoS parameters of the different service classes are shown in Table 5.2

Table 5.2: QoS Parameters of different service classes

Service Class	Minimum Bandwidth (in slots)	Maximum Bandwidth (in slots)	Delay (in slots)	Jitter (in slots)
UGS	20 – 30	20 – 30	200	10
rtPS	15 – 20	35 – 45	250	
nrtPS	15 – 20	35 – 45		
BE		100		

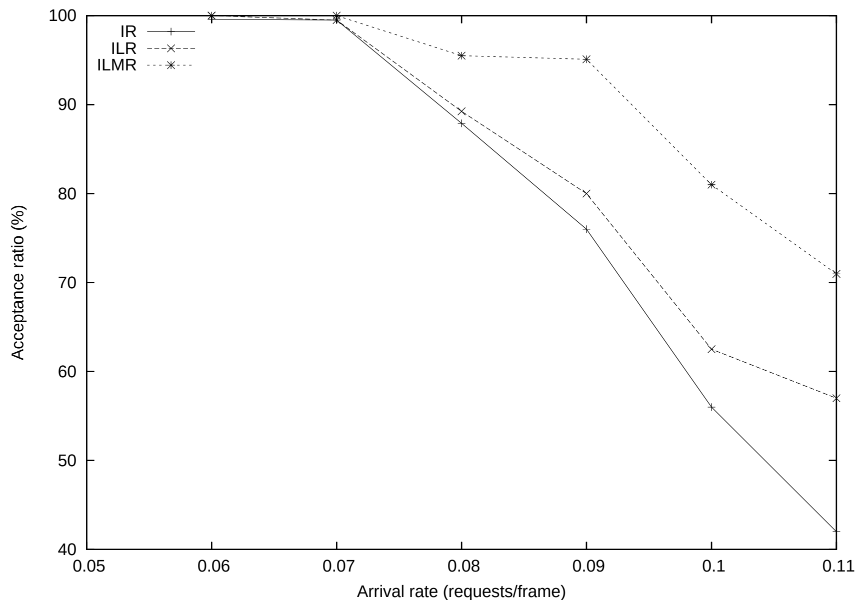


Figure 5.3: Acceptance ratio vs Arrival rate of requests

5.6.3 Acceptance Ratio

We investigate the acceptance ratio of the different routing schemes. For an incoming request, we compute the route based on the three routing schemes before it goes through call admission. We define the acceptance ratio as the ratio of the number of accepted requests to the total number of requests. Since BE requests do not go through call admission, we define the acceptance ratio only in terms of UGS, rtPS and nrtPS requests. Figure 5.6.3 illustrates the results obtained. The results indicate that ILMR produces the best acceptance ratio followed by ILR. IR does not ensure load sharing leading to congestion on some links in the network. As a result, more requests are rejected and this accounts for its poor performance. As expected, the results indicate that load sharing metrics are very useful in QoS provisioning when the network is congested. By using session based routing, the acceptance ratio was increased by 15%.

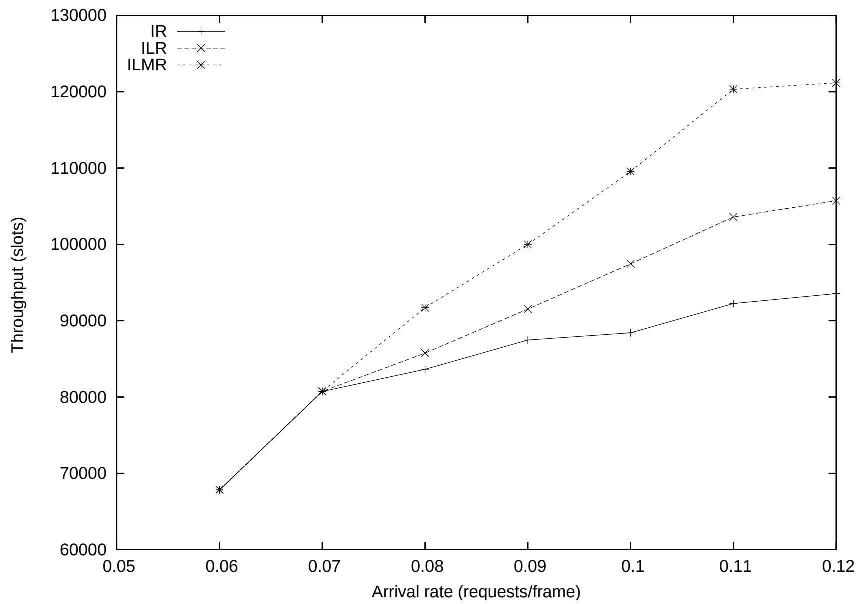


Figure 5.4: Throughput vs Arrival rate of requests

5.6.4 Throughput

We define throughput as the total number of packets received at the BS during the simulation time. Since ILMR accepts more requests, it produces the highest throughput followed by ILR. IR produced the least throughput as indicated on Figure 5.4. The results also indicate that load sharing has the potential to increase the efficiency of the use of network resources. ILMR increased the network throughput by approximately 25% while ILR achieved a 15% increase.

5.6.5 Packet Drop Ratio

In this section, we investigate the packet drop rate of the different routing schemes. We define the packet drop rate as the ratio of the number of rtPS packets that are dropped to the total number of rtPS packets. An rtPS packet is dropped if it fails to reach the BS by its deadline. Figure 5.5 illustrates the results for varying arrival rates. As the arrival rate

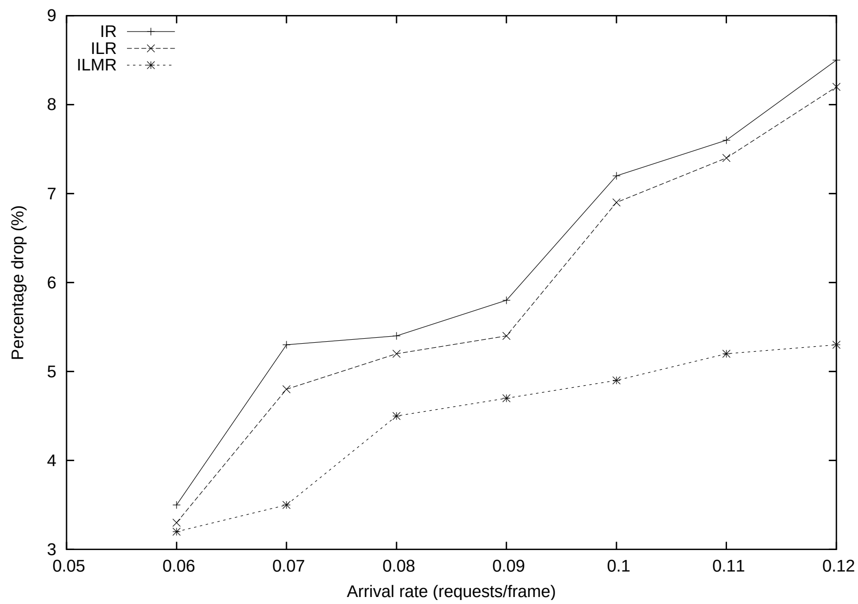


Figure 5.5: Packet drop vs Arrival rate of requests

increases, the packet drop increases for all the routing schemes. The results indicate that ILMR and ILR outperformed IR for the different arrival rates with ILMR producing the least packet drop. ILR and ILMR are able to allocate more extra slots to rtPS requests resulting in fewer packets missing their deadlines. By using session based routing, the percentage of packets dropped was approximately halved when the network was congested.

Chapter 6

Conclusion And Future Work

In this thesis, we study the resource allocation problem in WiMAX mesh networks. We propose a joint routing, centralized scheduling and channel assignment scheme for WiMAX mesh.

We present routing schemes that use a metric combining interference and load sharing. We depart from classical tree based routing by constructing session based routes and we quantify the gains when QoS guarantees are considered. We also propose a fast and effective heuristic algorithm for link scheduling. Our scheduling aims to find a shortest length schedule for all data to reach the BS by keeping the BS busy in each timeslot. We present a simple channel assignment algorithm inspired by a constraint programming heuristic which proves effective in maximizing the number of concurrent transmissions. We compare our scheme to a simple combinatorial bound through extensive simulations. Results from our experiments indicate that our scheme improves the network performance. Our session based routing and link scheduling produce results close to 90% of the trivial lower bound while our tree based routing achieved about 85% of the bound.

We also investigate the impact of routing, link scheduling, channel allocation and CAC on QoS provisioning in WiMax mesh networks. While several works consider these problems in isolation, we provide a comprehensive framework that considers all. For routing, we consider two schemes that incorporate interference and load sharing. One scheme constructs routing tree and the other assigns a path for each request without constraining the union of all routes to form a tree. We propose a link scheduling that is also used as CAC for new requests and considers all classes of service. We provide simulations results which indicate that load sharing metrics are indeed useful in QoS provisioning when the network is congested. Our session based routing scheme provided significant improvement in network performance. By using the session based routing, the acceptance ratio increased by approximately 15% while the percentage of packets dropped was almost halved.

Future work on WiMAX resource allocation can be driven in several directions. This thesis performs routing and scheduling separately which are sub optimal. In our future work, we will like to consider a scheme that performs both routing and scheduling at the same time. We will also like to make some extension to our QoS provisioning. As we considered unlimited buffer sizes in this work, we will study QoS and buffer management in our future work. We will also consider a CAC scheme that performs back tracking and changes slot assignment of accepted requests in order to find a feasible assignment for a new request.

Bibliography

[1] I. Akyildiz and X. Wang, "A survey on wireless mesh networks," *Communications Magazine, IEEE*, vol. 43, no. 9, pp. S23–S30, 2005.

[2] I. . W. Group *et al.*, "Ieee standard for local and metropolitan area networks, part 16: Air interface for fixed broadband wireless access systems," *IEEE Standard*, pp. 802–16, 2004.

[3] G. Narlikar, G. Wilfong, and L. Zhang, "Designing multihop wireless backhaul networks with delay guarantees," *Wireless Networks*, vol. 16, no. 1, pp. 237–254, 2010.

[4] Y. Zhang, *Wimax network planning and optimization*. Auerbach Publications, 2009.

[5] I. Akyildiz, X. Wang, and W. Wang, "Wireless mesh networks: a survey," *Computer networks*, vol. 47, no. 4, pp. 445–487, 2005.

[6] X. Li, *Wireless Ad Hoc and Sensor Networks: Theory and Applications*. Cambridge University Press, 2008.

[7] R. Bruno, M. Conti, and E. Gregori, "Mesh networks: commodity multihop ad hoc networks," *Communications Magazine, IEEE*, vol. 43, no. 3, pp. 123–131, 2005.

[8] D. Ghosh, A. Gupta, and P. Mohapatra, "Scheduling in multihop wimax networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 12, no. 2, pp. 1–11, 2008.

[9] B. Kumar and P. Gupta, "Scheduling algorithms in a wimax network," in *Advanced Computing & Communication Technologies (ACCT), 2012 Second International Conference on*, pp. 457–462, IEEE, 2012.

[10] P. Djukic and S. Valaee, "Centralized scheduling algorithms for 802.16 mesh networks," *Wimax/MobileFi: Advanced research and technology*, 2007.

[11] J. He, K. Yang, K. Guild, and H. Chen, "Application of ieee 802.16 mesh networks as the backhaul of multihop cellular networks," *Communications Magazine, IEEE*, vol. 45, no. 9, pp. 82–90, 2007.

[12] W. Stallings, *Wireless communications and networking*. Prentice Hall, 2002.

[13] S. Deb, V. Mhatre, and V. Ramaiyan, "Wimax relay networks: opportunistic scheduling to exploit multiuser diversity and frequency selectivity," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, pp. 163–174, ACM, 2008.

[14] P. Djukic and S. Valaee, "Delay aware link scheduling for multi-hop TDMA wireless networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 17, no. 3, pp. 870–883, 2009.

[15] Y. Qassem, A. Al-Hemyari, C. Ng, N. Noordin, and M. Rasid, "Review of network routing in ieee 802.16 wimax mesh networks," *Australian Journal of Basic and Applied Sciences*, vol. 3, no. 4, pp. 3980–3996, 2009.

[16] Y. Tang, Y. Yao, and X. Lin, "A joint centralized scheduling and channel assignment scheme in wimax mesh networks," in *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, pp. 552–556, ACM, 2009.

[17] P. Du, W. Jia, L. Huang, and W. Lu, "Centralized scheduling and channel assignment in multi-channel single-transceiver wimax mesh network," in *Wireless Communications and Networking Conference, 2007. WCNC 2007. IEEE*, pp. 1734–1739, IEEE, 2007.

[18] J. Xiao, N. Xiong, and A. Vasilakos, "Centralized scheduling and channel assignment scheme in third generation router based wimax mesh network," in *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, pp. 721–725, ACM, 2009.

[19] L. Chen, Y. Tseng, D. Wang, and J. Wu, "Exploiting spectral reuse in resource allocation, scheduling, and routing for ieee 802.16 mesh networks," in *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pp. 1608–1612, IEEE, 2007.

[20] Y. Cao, Z. Liu, and Y. Yang, "A centralized scheduling algorithm based on multipath routing in wimax mesh network," in *Wireless Communications, Networking and Mobile Computing, 2006. WiCOM 2006. International Conference on*, pp. 1–4, IEEE, 2006.

[21] J. Tao, F. Liu, Z. Zeng, and Z. Lin, "Throughput enhancement in wimax mesh networks using concurrent transmission," in *Wireless Communications, Networking and Mobile Computing, 2005. Proceedings. 2005 International Conference on*, vol. 2, pp. 871–874, IEEE, 2005.

[22] F. Jin, A. Arora, J. Hwang, and H. Choi, "Routing and packet scheduling in wimax mesh networks," in *Broadband Communications, Networks and Systems, 2007. BROADNETS 2007. Fourth International Conference on*, pp. 574–582, IEEE, 2007.

[23] H. Wei, S. Ganguly, R. Izmailov, and Z. Haas, "Interference-aware ieee 802.16 wimax mesh networks," in *Vehicular Technology Conference, 2005. VTC 2005-Spring. 2005 IEEE 61st*, vol. 5, pp. 3102–3106, Ieee, 2005.

[24] A. Raniwala, K. Gopalan, and T. Chiueh, "Centralized channel assignment and routing algorithms for multi-channel wireless mesh networks," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 2, pp. 50–65, 2004.

[25] L. Fu and Z. Cao, "Joint optimization of routing and scheduling for higher throughput in ieee 802.16 mesh networks," in *Wireless, Mobile and Multimedia Networks, 2006 IET International Conference on*, pp. 1–4, IET, 2006.

[26] T. Chung, H. Fu, and H. Lee, "Dcls: A fair and efficient scheduling with bandwidth guarantee in wimax mesh networks," *Journal of Information Science and Engineering*, vol. 26, no. 4, pp. 1429–1442, 2010.

[27] S. Nazari and H. Beigy, "A new distributed uplink packet scheduling algorithm in wimax newtorks," in *Future Computer and Communication (ICFCC), 2010 2nd International Conference on*, vol. 2, pp. V2–232, IEEE, 2010.

[28] P. Kong, H. Wang, Y. Ge, C. Ang, J. Pathmasuntharam, W. Su, M. Zhou, and H. Harada, "Distributed adaptive time slot allocation for wimax based maritime wireless mesh networks," in *Wireless Communications and Networking Conference, 2009. WCNC 2009. IEEE*, pp. 1–6, IEEE, 2009.

[29] B. Makarevitch, "Distributed scheduling for wimax mesh network," in *Personal, Indoor and Mobile Radio Communications, 2006 IEEE 17th International Symposium on*, pp. 1–5, IEEE, 2006.

[30] A. Morell, G. Seco-Granados, and J. Vicario, "Distributed algorithm for uplink scheduling in wimax networks," in *Broadband Communications, Networks and Systems, 2008. BROADNETS 2008. 5th International Conference on*, pp. 257–264, IEEE, 2008.

[31] R. Talwalkar and M. Ilyas, "Analysis of quality of service (qos) in wimax networks," in *Networks, 2008. ICON 2008. 16th IEEE International Conference on*, pp. 1–8, IEEE, 2008.

[32] H. Wang, W. Li, and D. Agrawal, "Dynamic admission control and qos for 802.16 wireless man," in *Wireless Telecommunications Symposium,2005*, pp. 60–66, IEEE, 2005.

[33] S. Lu, V. Bharghavan, and R. Srikant, "Fair scheduling in wireless packet networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 7, no. 4, pp. 473–489, 1999.

[34] K. Wongthavarawat and A. Ganz, "Packet scheduling for qos support in ieee 802.16 broadband wireless access systems," *International Journal of Communication Systems*, vol. 16, no. 1, pp. 81–96, 2003.

[35] J. Chen, W. Jiao, and Q. Guo, "Providing integrated qos control for ieee 802.16 broadband wireless access systems," in *IEEE Vehicular Technology Conference*, vol. 62, p. 1254, IEEE; 1999, 2005.

[36] G. Chu, D. Wang, and S. Mei, "A qos architecture for the mac protocol of ieee 802.16 bwa system," in *Communications, Circuits and Systems and West Sino Expositions, IEEE 2002 International Conference on*, vol. 1, pp. 435–439, IEEE, 2002.

[37] C. Cicconetti, A. Erta, L. Lenzini, and E. Mingozzi, "Performance evaluation of the ieee 802.16 mac for qos support," *Mobile Computing, IEEE Transactions on*, vol. 6, no. 1, pp. 26–38, 2007.

[38] D. Cho, J. Song, M. Kim, and K. Han, "Performance analysis of the ieee 802.16 wireless metropolitan area network," in *Distributed Frameworks for Multimedia Applications, 2005. DFMA'05. First International Conference on*, pp. 130–136, IEEE, 2005.

[39] C. Cicconetti, L. Lenzini, E. Mingozzi, and C. Eklund, "Quality of service support in ieee 802.16 networks," *Network, IEEE*, vol. 20, no. 2, pp. 50–55, 2006.

[40] J. Chen, W. Jiao, and H. Wang, "A service flow management strategy for ieee 802.16 broadband wireless access systems in tdd mode," in *Communications, 2005. ICC 2005. 2005 IEEE International Conference on*, vol. 5, pp. 3422–3426, Ieee, 2005.

[41] H. Safa, H. Artail, M. Karam, R. Soudah, and S. Khayat, "New scheduling architecture for ieee 802.16 wireless metropolitan area network," in *Computer Systems and Applications, 2007. AICCSA'07. IEEE/ACS International Conference on*, pp. 203–210, IEEE, 2007.

[42] P. Ting, C. Yu, N. Chilamkurti, W. Tung-Hsien, and C. Shieh, "A proposed red-based scheduling scheme for qos in wimax networks," in *Wireless Pervasive Computing, 2009. ISWPC 2009. 4th International Symposium on*, pp. 1–5, IEEE, 2009.

[43] P. Goyal and A. Sahoo, "A scheduling and call admission control algorithm for wimax mesh network with strict qos guarantee," in *Communication Systems and Networks (COMSNETS), 2010 Second International Conference on*, pp. 1–10, IEEE, 2010.

[44] L. You, M. Song, and J. Song, "Multi-service scheduling mechanism with qos guarantee for centralized scheduling of wimax mesh mode," in *Global Mobile Congress 2009*, pp. 1–6, IEEE, 2009.

[45] D. Ghosh, A. Gupta, and P. Mohapatra, "Admission control and interference-aware scheduling in multi-hop wimax networks," in *Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on*, pp. 1–9, IEEE, 2007.

[46] X. Cheng, P. Mohapatra, S. Lee, and S. Banerjee, "Maria: Interference-aware admission control and qos routing in wireless mesh networks," in *Communications, 2008. ICC'08. IEEE International Conference on*, pp. 2865–2870, IEEE, 2008.

[47] A. Sayenko, O. Alanen, J. Karhula, and T. Hämäläinen, "Ensuring the qos requirements in 802.16 scheduling," in *Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, pp. 108–117, ACM, 2006.

Appendix A

Glossary

ATM: Asynchronous Transfer Mode
BE: Best Effort
BFS: Breadth First Search
BS: Base Station
BPSK: Binary Phase Shift Keying
BWA: Broadband Wireless Access
CAC: CAll Admission Control
CUR: Channel Utilization Ratio
DSL: Digital subscriber Line
FDM: Frequency Division Multiplexing
FTP: File Transfer Protocol
GPC: Grant Per Connection
GPSS: Grant Per Subscriber Station
HTTP: Hyper Text Transfer Protocol
IEEE: Institute of Electrical and Electronic Engineering
IP: Internet Protocol
ITU: International Telecommunication Union
LOS: Line of Sight
MAC: Medium Access Control
MPEG: Moving Pictures Experts Group
MSH-CSCH: Mesh Centralized Scheduling
MSH-DSCH: Mesh Decentralized Scheduling
MSH-NCFG: Mesh Network Configuration
MSH-NENT: Mesh Network Entry
nrtPS: Non-real Time Polling Service
OFDM: Orthogonal Frequency Division Multiplexing
OFDMA: Orthogonal Frequency Division Multiple Access
PDU: Protocol Data Unit
PHY: Physical (Layer)
QAM: Quadrature Amplitude Modulation
QPSK: Quadrature Phase Shift Keying
RS: Relay Station
rtPS: Real Time Polling Service
SDU: Service Data Unit
SINR: Signal to Interference plus Noise Ratio
SOFDM: Scalable Orthogonal Frequency Division Multiplexing
SS: Subscriber Station
TDMA: Time Division Multiple Access

UDG: Unit Disk Graph
UGS: Uninterrupted Grant Service
VoIP: Voice Over Internet Protocol
WMN: Wireless Mesh Network
WMAN: Wireless Metropolitan Area Network
WiMAX: World Interoperatability for Microwave Access