Analysis of Throughput Distribution in IEEE 802.16 Distributed Scheduling

By
Madiha Wahid
2008-NUST-MS PhD-IT-36

Supervisor
Dr. Anjum Naveed
NUST-SEECS

A thesis submitted in partial fulfillment of the requirements for the degree of Masters of Science in Information Technology (MS IT)

In
School of Electrical Engineering and Computer Science,
National University of Sciences and Technology (NUST),
Islamabad, Pakistan.

(March 2012)

Approval

It is certified that the contents and form of the thesis entitled "**Analysis of Throughput Distribution in IEEE 802.16 Distributed Scheduling** " submitted by **Madiha Wahid** have been found satisfactory for the requirement of the degree.

Advisor: **Dr. Anjum Naveed**

Signature: _______________
Date: _______________

Committee Member 1: **Dr. Adnan Khalid Kiani**

Signature: _______________
Date: _______________

Committee Member 2: **Dr. M. Shahzad Younis**

Signature: _______________
Date: _______________

Committee Member 3: **Dr. Zawar Hussain Shah**

Signature: _______________
Date: _______________

Abstract

IEEE 802.16 offers mesh mode of operation to cover outage areas outside the range of Base Station (BS). The standard defines three scheduling mechanisms for mesh mode: coordinated centralized, coordinated distributed and uncoordinated distributed. In coordinated distributed scheduling, all nodes are treated equally and coordinate their transmissions within their two-hop neighbourhood. Since no central authority is involved, throughput distribution among nodes is unfair. Some nodes may undergo starvation and some may enjoy highest achievable throughput. In this thesis, we analyse the distributed scheduling mechanism of WiMAX mesh mode to study the fairness issue. We model the behaviour of a single node under distributed scheduling mechanism using 2-D Markov Chain where a node can be in one of four states of holdoff, election, wait and transmission. Analytical results of the model show that time spent by a node to fulfil its data demand is considerably small (maximum upto 10%). Our analysis shows that for most of the time a node is either in holdoff (at least 24%) or competing for mesh election(on average 40%). Moreover, some allowable high values of holdoff exponent (i.e. 4 to 7) by standard produce very unrealistic scenarios with incredibly scarce chances of data transmission. To control unfairness and impracticality of distributed scheduler, optimal values of holdoff exponent are found to be between 0 and 3. Handshake probability value greater than 0.1 yields more chances for data transmission. The gap from current time till the arrival of requested data slot does not have any significant impact on time duration spent in different states.

Certificate of Originality

I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, nor material which to a substantial extent has been accepted for the award of any degree or diploma at National University of Sciences & Technology (NUST) School of Electrical Engineering & Computer Science (SEECS) or at any other educational institute, except where due acknowledgement has been made in the thesis. Any contribution made to the research by others, with whom I have worked at NUST SEECS or elsewhere, is explicitly acknowledged in the thesis.

I also declare that the intellectual content of this thesis is the product of my own work, except for the assistance from others in the project's design and conception or in style, presentation and linguistics which has been acknowledged.

Author Name: **Madiha Wahid**

Signature: ____________________

Acknowledgement

First of all I would like to express my deepest gratitude to ALLAH The Almighty. Its only the belief in Him that makes it possible for me to complete my thesis. I can never enough acknowledge Him as being an ordinary human being, but all what I have achieved in my life till today is just His blessing. Thanks to Allah.

It is great pleasure to thanks my supervisor, Dr. Anjum Naveed whose guidance and encouragement from the beginning till the end aided me in understanding the topic and complete it well.

Special thanks to my dear friends Umay, Amber and Sobia. You people have always been there for me and and the support that I get from you people is just marvellous.

I wish to express my love and gratitude to my both families, especially to my husband, my brother and my mother, for supporting and encouraging me to complete the thesis. Without my husband's support I would not have been able to complete my degree.

Madiha Wahid

Contents

List of Abbreviations

Abbreviations	**Descriptions**
WMN	Wireless Mesh Networks
LAN	Local Area Network
MAN	Metropolitan Area Network
WiMAX	Worldwide interoperability for Microwave Access
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiplexing Access
SC	Single Carrier
WirelessHUMAN	Wireless High-Speed Unlicensed Metropolitan Area Network
PHY	Physical Layer
MAC	Medium Access Control
PMP	Point-to-Multipoint
BS	Base Station
SS	Subscriber Station
UL	Uplink
DL	Downlink
LOS	Line Of Sight
NLOS	Non Line Of Sight
SDU	Service Data Unit
QoS	Quality of Service
MSH-DSCH	Mesh Distributed Scheduling
TO	Transmission Opportunity
XHE	Xmt Holdoff Exponent
Mx	Next Xmt Mx

List of Figures

List of Tables

Chapter 1

Introduction

With the increased use of mobile computing devices, need for connecting these devices to internet is an urge of today's world. Wireless Mesh Networks (WMNs) provide connectivity to mobile\fixed nodes organized in mesh topology.

1.1 Wireless Mesh Network

WMN is composed of mobile nodes, mesh routers and gateways. Gateways provide access to mesh routers for back haul services and nodes are connected to routers. Intranet communications take place via multi hop wireless paths between transceivers. Figure1.1 shows an example of wireless mesh network. WMN can be seen as a special type of wireless adhoc networks, where no infrastructure is required. Nodes can join the network when they are in communication range of mesh routers or other already connected nodes. Hence WMN are less expensive and they offer advantage of providing access to nodes that are not within the communication range of mesh routers.

Generally WMN nodes carries 802.11a\b\g\n radios which are standards for Wireless LAN (WLAN). WLAN radios carrier sensing range is maximum upto 250 meters approximately, which is not enough to cover larger areas like complete city. To enable mass level deployments of WMN, IEEE issued 802.16d standard for providing broadband wireless access in MAN with the support of mesh mode in 2004. This is commonly known as WiMAX standard.

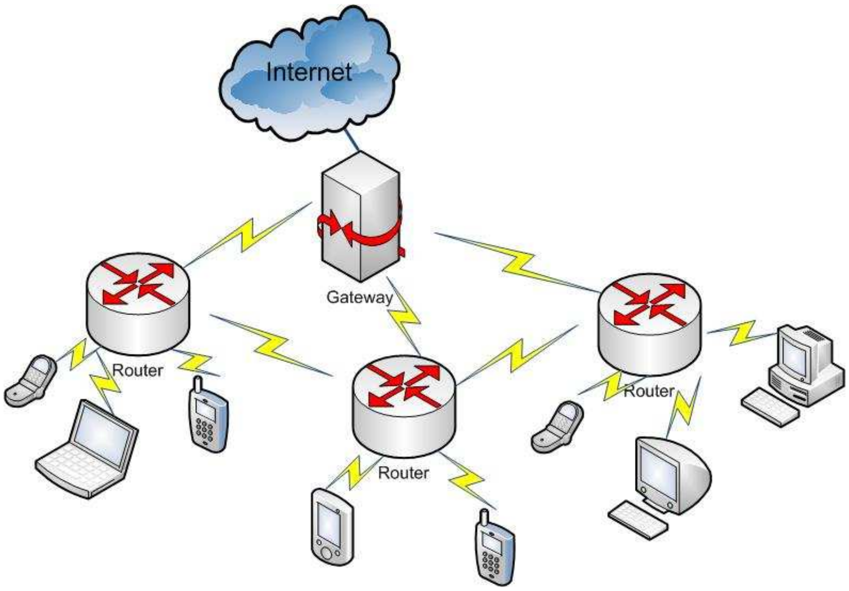


Figure 1.1: Wireless Mesh Network

1.2 WiMAX Networks

IEEE 802.16 standard published in 2004 [1], provides air interface specifications for PHY and MAC layer. PHY layer provides functionalities of modulation and channel coding. Physical layer supports different modulation schemes like 16 QAM, 64 QAM and QPSK and hence offer different data transmission rates.

MAC layer is responsible for scheduling user requests. MAC layer is subdivided into three further layers; Service-Specific Convergence Sublayer (CS), MAC Common Part Sublayer (CPS) and Security Sublayer. CS sublayer is responsible for mapping of external data network packets into MAC SDUs. The CPS sublayer provide functionalities for bandwidth allocation, QoS control, fragmentation, scheduling and retransmission of MAC SDUs. Security sublayer is in charge of authentication, data encryption and secure key exchange.

WiMAX standard specifies various air interfaces as summarized in Table1.1. MAC layer offers two mode of operations, Point-to-Multipoint (PMP) and an optional mesh mode. PMP mode has similar architecture like traditional cellular networks, where Subscriber Stations (SS) have direct wireless connection with Base Station(BS) and there is no one-to-one communication between the SSs. BS is responsible for all types of communications, manage-

Table 1.1: Air Interfaces Specification

Air Interface	**Frequency Band**
WirelessMAN-SC	10-66 GHz
WirelessMAN-SCa	Below 11 GHz(licensed bands)
WirelessMAN-OFDM	Below 11 GHz(licensed bands)
WirelessMAN-OFDM	Below 11 GHz(licensed bands)
WirelessMAN-OFDMA	Below 11 GHz(licensed bands)
WirelessHUMAN	Below 11 GHz(license exempt bands)

ment and operations of network. SSs can not be part of network if they are outside the range of BS. If these SSs have to be provided with connectivity, then there is need of new BS. BS is very expensive unit so in order to avoid high costs and to provide more connectivity, WiMAX supports mesh mode of MAC as well.

1.3 WiMAX Mesh Mode

Like WMN, nodes are organized in mesh topology for WiMAX mesh mode. Nodes do not necessarily have to be within one-hop range of BS. Nodes can forward their data to their neighbours and multi-hop forwarding of data makes it possible for BS to receive data from nodes that are quite far away from the BS.

Figure 1.2 shows an example of WiMAX Mesh network. A node that has connection to back haul services outside a mesh network is called a Mesh BS. All other nodes are termed as Mesh SS. In general, the Mesh SSs are simply called nodes in Mesh mode. There can be direct communication among the neighbouring nodes or data can be routed via multi-hop path to reach mesh BS in case of internet traffic.

Mesh mode is designed to operate in WirelessMAN-OFDM for licensed band and in WirelessHUMAN air interface for unlicensed spectrum. For unlicensed spectrum also, the modulation technique used is OFDM. OFDM is used for NLOS operation in frequency bands below 11 GHz. NLOS operation is possible due to shorter wavelength and multipath propagation in the frequency band below 11 GHz.

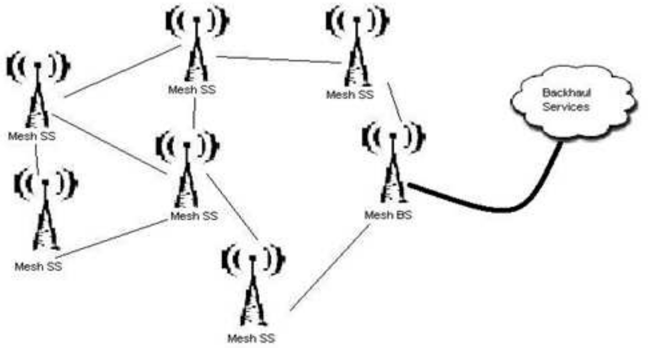


Figure 1.2: WiMAX Mesh Network

1.4 Scheduling in WiMAX Mesh Mode

Two broad types of scheduling supported by WiMAX mesh mode are:

- Centralized Scheduling
- Distributed Scheduling

1.4.1 Centralized Scheduling

In case of centralized scheduling a routing tree is formed from the mesh BS towards other nodes. Nodes forward their data and data of neighbouring nodes below them to nodes one-hop above them such that data from the whole network reaches the mesh BS. It is just like PMP mode where BS is responsible for all the intranet and internet communications. Figure 1.3 shows an example of routing tree formed for centralized scheduling. All the mesh SSs are connected with the mesh BS via this routing tree.

Centralized scheduling is not preferable in WiMAX mesh mode as it is unable to provide all the advantages of mesh mode like resilience and direct communication among mesh SSs. Therefore distributed scheduling is more desirable choice.

1.4.2 Distributed Scheduling

Scheduling is performed in decentralized manner in case of distributed scheduling. There is direct communication between mesh SSs that are within one-hop range of the node. Nodes inform their two-hop neighbours about their schedule to avoid collisions and interference such that no other node within

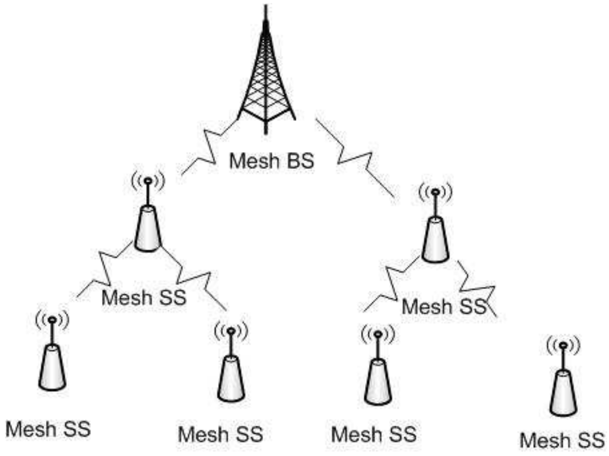


Figure 1.3: Routing Tree for Centralized Scheduling

two-hop neighbourhood transmit at the same time. Mesh BS is not involved in distributed scheduling. Figure 1.4 shows an example of nodes connections in case of distributed scheduling.

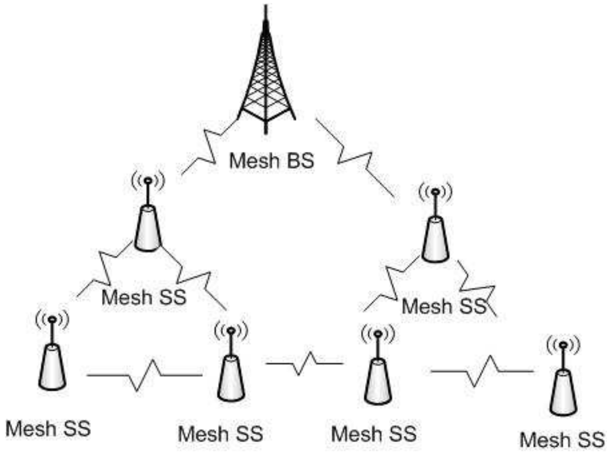


Figure 1.4: Distributed Scheduling

1.5 Motivation

Though distributed scheduling is more desirable for WiMAX mesh networks but literature surveys shows that performance of distributed scheduler is not upto the mark. Aggregate throughput achieved by nodes is quite poor. Also there is debate regarding number of users supported by distributed scheduler. It is observed that if large number of users are there in the system, then some nodes may undergo starvation.

So this thesis aims to analyse the distributed scheduler performance to find out the reasons behind inefficient throughput achieved by nodes.

1.6 Thesis Organization

Rest of the thesis is organized as follows. Chapter 2 studies in detail the distributed scheduling mechanism followed by review on the studies already done on distributed scheduler performance. In chapter 3, we develop an analytical model based on distributed scheduler working. We also build up equation to estimate a node's chance to successfully complete the three-way handshake, which is very important factor in regard of our thesis topic. Chapter 4 evaluates the analytical model. We determine the impact of different parameters on throughput achieved by nodes. Chapter 5 concludes the thesis with our findings on optimal values of all effecting parameters.

Chapter 2

Literature Review

The IEEE standard [1] offers two mode of operations at MAC layer, primarily Point-to-multipoint (PMP) and optional mesh mode. PMP mode is much similar to traditional cellular networks. A frame is divided into uplink subframe and downlink subframe. All SS share portions in same downlink and uplink subframes. Exact time allocated to each SS is defined in UL-Map and DL-Map fields. PMP mode offers QoS support by allowing different traffic types.

2.1 IEEE 802.16 Mesh Mode

Like PMP mode, it is also time slotted based system. Mesh mode adopts the Time Division Multiplexing (TDM) radio access technology. A frame is broadly categorized into two parts; control subframe and data subframe. Control subframe is responsible for sending control signals and messages related to network configuration and data scheduling. Data subframe is responsible for actual data transmission.

Control subframe is further divided into transmission Opportunities (TO). One TO counts to 7 OFDM symbols duration.The exact number of TOs in control subframe is determined by the field of $\mathsf{MSH} - \mathsf{CTRL} - \mathsf{LEN}$ in Network Descriptor [1]. Length of control subframe is fixed and equal to $\mathsf{MSH} - \mathsf{CTRL} - \mathsf{LEN} \times 7$ OFDM symbols. Control subframe is of two types; Network Control Subframe and Schedule Control Subframe. Network Control Subframe occurs periodically as defined by network operator via the Scheduling Frames field of Network Descriptor. Network Control subframe further includes two types of messages; *MSH-NENT* (Mesh Network Entry) and *MSH-NCFG* (Mesh Network Configuration). *MSH-NENT* is used for entry of new nodes in the network and initial synchronization of new nodes. The *MSH-NCFG* message is send by all nodes to keep an uptodate view of the network. This message include information about neighbouring nodes. Figure 2.1 explains the format of mesh mode frame. The schedule control subframe is used for coordinated scheduling of data transfer. *MSH-DSCH-NUM* field in network descriptor specifies the number of TOs reserved for distributed scheduling (*MSH-DSCH*) messages.*MSH-CTRL-LEN* - *MSH-DSCH-NUM* TOs are used for sending centralized scheduling messages. All the messages in control subframe are sent using $QPSK - 1/2$ with necessary coding scheme.

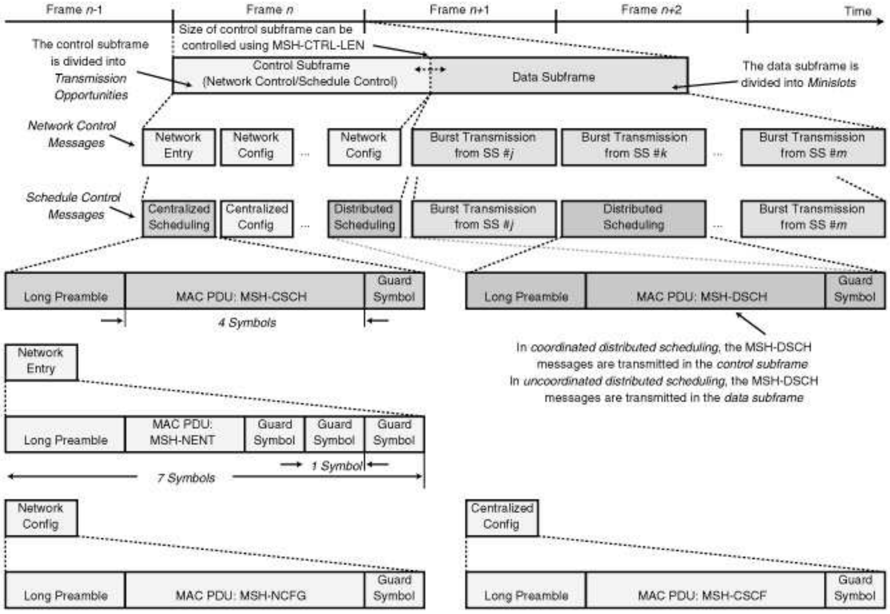


Figure 2.1: Mesh Mode Frame Structure [2]

The data subframe is partitioned into number of data minislots. Minislot is the smallest unit for resource allocation. Minislots in the start of each data subframe are reserved for centralized scheduling. *MSH-CSCH-DATA-FRACTION* field in Network Descriptor IE indicates the maximum percentage of data subframe reserved for centralized scheduling. The rest of data subframe is used by distributed scheduling. The transmission rate r bits per second that a minislot can provide depends on several factors e.g., channelcoding, modulation, and frequency band. The frame structure is explained in Figure 2.1.

In mesh mode bandwidth request and grant mechanisms cannot be like the ones used in PMP mode. The three types of scheduling mechanisms supported by IEEE 802.16 mesh mode are explained below.

2.2 Coordinated Centralized Scheduling

In centralized scheduling, Mesh BS is responsible for allocation of resources to all nodes. It acts like a central authority. A routing tree is formed from Mesh BS to nodes within a certain hop range. Resource requests are gathered from all nodes down the tree. BS then determines flow assignment to each SS request by dividing the frame proportionally among these requests.

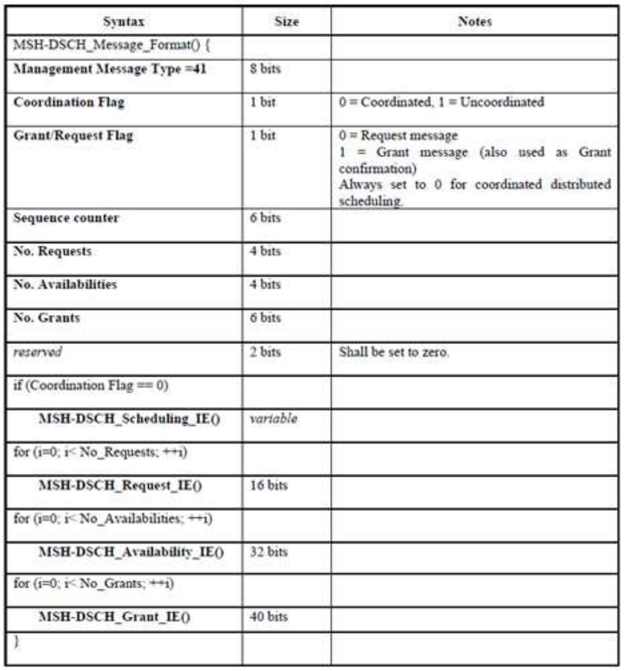
Centralized scheduling is accomplished by means of two control messages: *MSH-CSCH* (Mesh Centralized Scheduling) and *MSH-CSCF* (Mesh Centralized Scheduling Configuration). *MSH-CSCF* message is broadcast by BS to all attached nodes and further forwarded by each node to nodes down the routing tree. *MSH-CSCF* is used to update information of nodes in the routing tree. *MSH-CSCH* message is used to carry request and grant information. A node send *MSH-CSCH* message to indicate its resource request and requests of its child nodes in a sub-tree.

The Mesh BS acts just like a BS in PMP mode, except that SS should not must have direct links to BS. Mesh BS ensures collision free transmissions over the links of routing tree. Hence, BS is responsible for overall traffic control and network related activities.

2.3 Coordinated Distributed Scheduling

In distributed scheduling, nodes that have direct link with each other forms the neighbourhood. There is no central BS involved in scheduling. Nodes coordinate their transmission in extended two-hop neighbourhood by regularly exchanging their schedules in control subframe via *MSH-DSCH* message. Format of *MSH-DSCH* message is explained in Figure 2.2. Coordinated Distributed Scheduling is achieved by means of three-way handshake mechanism provided in Standard [1]. It is a three step procedure necessary to be completed before actual successful transmission of data in data subframe part. It is explained in Figure 2.3.

- **Request:** A node sends a *MSH-DSCH* Request IE indicating the Link ID on which node wants to reserve bandwidth, along with fields of *Demand Level* and *Demand Persistence*. *Demand Level* indicates the bandwidth demand in terms of minislots and *Demand Persistence* reflects the number of frames wherein demand exists. A requester also



Syntax	Size	Notes
MSH-DSCH_Message_Format() {		
Management Message Type =41	8 bits	
Coordination Flag	1 bit	0 = Coordinated, 1 = Uncoordinated
Grant/Request Flag	1 bit	0 = Request message 1 = Grant message (also used as Grant confirmation) Always set to 0 for coordinated distributed scheduling.
Sequence counter	6 bits	
No. Requests	4 bits	
No. Availabilities	4 bits	
No. Grants	6 bits	
reserved	2 bits	Shall be set to zero.
if (Coordination Flag == 0)		
MSH-DSCH_Scheduling_IE()	*variable*	
for (i=0; i< No_Requests; ++i)		
MSH-DSCH_Request_IE()	16 bits	
for (i=0; i< No_Availabilities; ++i)		
MSH-DSCH_Availability_IE()	32 bits	
for (i=0; i< No_Grants; ++i)		
MSH-DSCH_Grant_IE()	40 bits	
}		

Figure 2.2: MSH-DSCH Message Format [1]

indicates upto 16 sets of minislots available for data transmission by means of *MSH-DSCH_Availability_IE*.

- **Grant:** A node attached to other side of link on which request was made will be called granter from here on. Granter matches the requested minislots to its available minislots . If they fit in its available minislots, then it will reply back with *MSH-DSCH_Grant_IE*
- **Grant-Confirm:** To confirm the schedule, requester sends back a grant confirm message to the granter.

All the stations in a network, use same channel for transmitting scheduling information in terms of requests and grants. So there are quite many chances of collisions to occur. To avoid collisions in control subframe each node broadcasts its transmission timing by means of *MSH-DSCH_Scheduling_IE* in *MSH-DSCH* message. Each node reports two parameters of its own and its 1-hop neighbours. This information helps in calculating Next Transmission Time (NextXmtTime) and holdoff time (XmtHoldoffTime)of a node.These two important parameters contained in Scheduling IE are *Next Xmt Mx* (Mx)and *Xmt Holdoff Exponent* (XHE). The XmtHoldoff Time is the number of *MSH-DSCH* TOs after NextXmtTime, that this node is ineligible to transmit *MSH-DSCH* packets. This time is computed as:

$$XmtHoldoffTime = 2^{XHE+4} \tag{2.1}$$

The NextXmtTime, which is the next *MSH-DSCH* eligibility interval for this node, is computed as the range:

$$2^{XHE} \cdot Mx < NextXmtTime \leq 2^{XHE} \cdot (Mx + 1) \tag{2.2}$$

If a node discovers that its NextXmtTime is overlapped with some other stations' NextXmtTime. The scenario is considered "collision". To prevent that, a mesh election algorithm is given in the standard. The mesh election algorithm is similar to a hashing function, only one station shall win the mesh election. Detail of mesh election algorithm is provided in the standard[1]. Mesh election is held among the local node and set of eligible competing nodes using *TempXmtTime* and list of Node IDs of all competing nodes as an input. *TempXmtTime* is set as first TO in the eligibility interval of node. A neighbouring SS is considered to be a competing node if

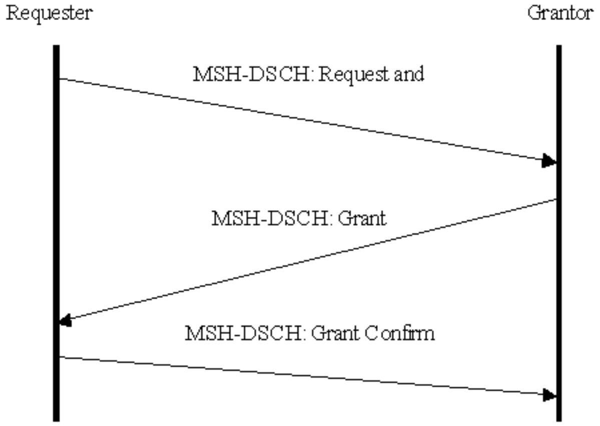


Figure 2.3: Three-Way Handshaking

- Its NextXmtTime interval includes the temporary transmission opportunity i.e. *TempXmtTime*
- Its Earliest Subsequent Xmt Time is less or equal to the transmission opportunity, where Earliest Subsequent Time = NextXmtTime + XmtHoldoffTime.

- Its NextXmtTime is not known.

Node that wins the mesh election sets the NextXmtTime equal to TempXmtTime. And other nodes that lose the election compete for Next *MSH-DSCH* opportunity.

2.4 Uncoordinated Distributed Scheduling

The scheduling mechanism for both coordinated and uncoordinated distributed schedulers is same. The only difference lies in the method of transmitting control scheduling messages. For uncoordinated distributed scheduler, the scheduling messages i.e. *Request*, *Grant* and *Grant − Confirm* are sent in data subframe instead of control subframe. Since no coordination occurs among the interfering nodes for adjusting the transmission timing of these messages, collisions are quite likely to occur. Nodes access the medium randomly in pure aloha MAC protocol fashion.

2.5 Related Work

N. Abu Ali et.al. [3] and M. Kas et.al [4] had surveyed the issues in IEEE 802.16 mesh schedulers. They surveyed the existing research done for all the three types of scheduling; centralized, coordinated distributed and uncoordinated distributed. They highlighted key challenges that had not been addressed so far in the literature at that time. Proposals on coordinated scheduling do not address all the issues highlighted by authors, and secondly in most of the proposals there is an inherent deviation from the standardś capabilities and limitations. Proposals on coordinated distributed scheduling do not analyse the control and data schedulers together. Also it is not clear whether the proposals are accommodated within the standard. No proposals have been made for the uncoordinated distributed scheduler. This could be due to the schedulerś definition in the standard, where it is mandated that uncoordinated contentions must not collide with schedules set by coordinated schedulers.

The research about distributed scheduling for IEEE 802.16 can be mainly grouped into two. The first group focuses on the performance evaluation of the distributed schedulers, by analysing the election based transmission timing of control messages. They derive or extend a mathematical model to analyse and model its behaviour or propose techniques to improve the performance of distributed scheduler mechanism. The studies in the second group mostly propose algorithms to fulfil the data scheduling step left open in the standard. Our focus will be on first group of studies.

The first ever theoretical investigation of mesh mode distributed scheduler is done by authors of [6] and [5]. They developed a stochastic model with assumption that transmit time sequences of all nodes form statistically independent renewal processes. Authors investigated time between two consecutive successful transmissions to be the sum of holdoff time and expected number of slots lost before winning the mesh election. They also derive the equation for delay encountered in connection setup. XHE value is chosen from range between 0-4, as values greater than 4 causes intolerable latency in connection setup. Other than total node number, network topology and XHE value, the traffic generation pattern of nodes is also very important parameter that is missing in this research.

V. Loscri [7] have suggested to adjust the XMT Holdoff Exponent (XHE) value based on the queue size. A node that has more data packets pending in its queue should be given more chances. Therefore its XHE is set to smaller value and vice versa. This dynamic adjustment leads to better results in terms of throughput and delay but average latency is increased.

Cesar et.al. proposed gradual dynamic adjustment of XHE [8]. If a node either has pending request or grants to send its XHE is decremented once. Even if buffer is not empty, XHE is decremented and if none of the above stated condition is satisfied, XHE is incremented gradually till it reach the value of 4. This scheme leads to better throughput with variable traffic demands but do not show improvement with CBR traffic.

IEEE 802.16 distributed scheduling intends to provide collision free transmissions by relying on assumption of quasi-interference RF model. This model assumes that all RF signals are completely absorbed by one-hop neighbor and therefore it is possible to perform concurrent transmissions in extended neighborhood. However in [9], this assumption is proved wrong in realistic environments by extensive simulations in QualNet. Authors provide new scheme called Collision Free- Coordinated Distributed Scheduling (CF-CDS) to guarantee collision free control scheduling and on demand data scheduling that outperforms traditional three-way handshaking approach in terms of throughput and delay.

R. Krenz [10] has determined the capacity of WiMAX mesh networks by applying the concept of collision domains in chain topology. The bandwidth on links near the gateway greatly affects the performance of 802.16 based WMN but this result is not validated by any simulations.

N.Bayer et.al. [11] suggested some extensions to Election Based Transmission Timing (EBTT) mechanism provided in IEEE 802.16 standard[1]. It is highlighted that the consistent numbering of transmission opportunities at all nodes is important, as it is the seed value for the pseudo-random component of EBTT mechanism. The reference point calculation for two hop neighbors is unstandardized as well. Methods are formulated for consistent numbering of TOs at all nodes and to calculate reference point value.

In another research by N.Bayer et.el. [12], focus of research is on scalability problem caused in dense configurations of IEEE 802.16 mesh networks. To improve the performance, dynamic adaptation of XHE is proposed based on the status of node to reduce contention in network. Moreover, authors have suggested putting a limit on the maximum requestable bandwidth in order to provide fairness to some extent.

To improve the performance of distributed scheduler, Wang et.al. [13] suggested that the holdoff time should be set to an appropriate value that is large enough to avoid congestion but small enough to avoid large transmission delays. So they proposed static and dynamic two-phase approach to set holdoff time. In first phase, nodes set their holdoff time statically according to number of nodes in two-hop neighborhood. If node has data to send its holdoff time is adjusted dynamically to shorten the transmission cycle. MAC layer performance is quantified against few performance metrics and this scheme outperforms the fixed-value schemes.

The performance of mesh election procedure is investigated via extensive simulations in ns-2 [14] by varying system parameters like frame duration, number of control slots per frame, XHE and network topology. It is found that access delay is directly proportional to the frame duration and value of XHE. However, access interval is reduced by increasing the number of control slot per frame. Network topology greatly affects the performance of network. The ideal value of XHE is to be set as zero for all scenarios considered in the study.

In study by M.Zhang et.al. [15], dynamic adjustment of XHE is proposed to guarantee QoS for multimedia services. On network entry, XHE value of node is initialized as 0. Afterwards, each node adjusts its XHE value according to its node type and the contention extent. This scheme performs better than static approach in terms of throughput and average delay.

The coordinated distributed scheduling is considered to be collision free in [1]. Study in [16] examines the conditions under which coordinated distributed scheduling can actually be collision free. Therefore appropriate selection of parameters is required to guarantee a correct collision free functionality.

Based on the study of [6] and [5], the holdoff time of node k is 2^{XHE+4} where 4 is base value. This base value of 4 imposes a restriction that any node can transmit at most once in a control subframe. S. Chakraborty et.al. [17] suggested the base value to be set according to the size of two-hop neighbourhood. This induces fairness w.r.t topology of network. This technique enhances the scheduler performance for different types of traffic.

Our work focuses on analytical modelling of the distributed scheduler to get a deep insight on the parameters effecting the scheduler's performance. Previous studies have shown the efficacy of holdoff exponent value on the scheduler performance. Our study consent with these findings but we also show that other than holdoff exponent, there are some other parameters also that can effect the scheduler performance like the number of contending neighbours and the bandwidth demanded by nodes. In chapter three we discuss the analytical model and the contributing parameters in detail.

Chapter 3

Analytical Model for Distributed Resource Scheduling

In order to model the distributed scheduler performance, we have build the analytical model to represent the behaviour of a single node in a resource allocation process. Since in WiMAX mesh mode, all nodes are treated equally. So we are confident in our claim that all the mesh nodes behave in a same manner for resource allocation.

3.1 Model Formulation

Resource allocation process for a single node is expressed as a markov chain where station undergoes transitions from one state to another. Figure 3.1 shows all the possible states and transitions between them. Table 3.1 summarizes all the notations used in modelling.

3.2 Model Explanation

Any node undergoes four phases for transmitting data on allocated resources i.e. holdoff, mesh election, wait for allocated resources and the actual data transmission. To represent these stages and transitions between them, a 2-D markov model in three variables is used where each state is represented as a combination of three parameters i.e. (i, j, k). i represents the value of holdoff counter, j represents the number of transmission opportunities and k represents the number of frames in one resource allocation cycle.

Table 3.1: Used Parameters

i	Value of holdoff counter
j	Number of TO
k	Number of frame
exp	Holdoff exponent Value
imax	2^{exp+4}
X	Number of frame in which first requested minislot resides
P_{hand}	Probability of successful handshake

According to IEEE standard [1], each node has to spend some time in holdoff phase before making a bandwidth request . Duration of holdoff time is dictated by the value of exp. Holdoff counter assumes maximum value of $imax - 1$ in the beginning and decrements by one in every time slot. Once the holdoff counter expires by reaching the value of zero, node enters into mesh election phase. In this stage, value of i will remain zero. Hence the allowable range for parameter i is from $imax - 1$ to zero. We have assumed that each control subframe is composed of 4 TOs. So after each frame, value of j is incremented by 4. The node remains in mesh election phase until three-way handshake is successfully achieved or frame X is reached. X is the number of first frame indicated as available in *MSH-DSCH_Availability_IE* or in other words it is a frame in which requester wants to transmit data. X can assume any positive value based upon node's own schedule. Maximum allowable time for a node to complete three-way handshake is thus upto arrival of frame X. Allowable range for parameter j is thus between 4 to $4.X$. After each frame, node checks whether handshake has been successful or not. We estimate the chance of successful handshake as probability of completing the three-way handshake and represent it as P_{hand}. Node competes for mesh election in further one more frame with probability of $1 - P_{hand}$ after every unsuccessful attempt. If node is unable to complete three-way handshake before the arrival of frame X, then node will jump back to holdoff phase with probability $\frac{1-P_{hand}}{imax}$ before making new bandwidth request.

Frame numbers are represented by variable k and its value varies between 1 to X. Upon successful completion of handshake in any frame between 1 to X-1, node stop competing in mesh election and enters wait phase with probability P_{hand} . In this state node waits till the frame X arrives and actual data transmission takes place. If handshake is completed in X frame, then node doesn't wait any further and start data transmission on allocated

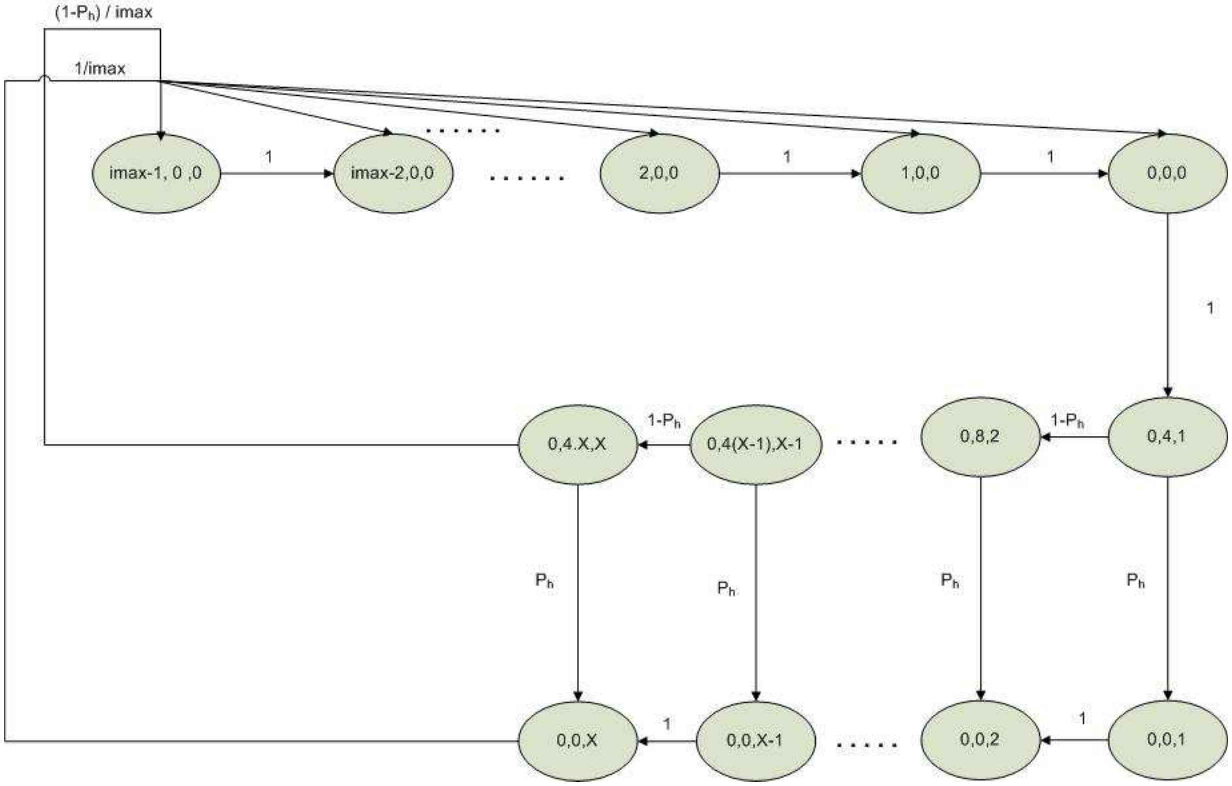


Figure 3.1: 2-D Markov Model for Data Transmission

minislots in the data subframe of the same frame. After data transmission, node again enter back into any of the holdoff states with probability $\frac{1}{imax}$.

Considering research problem, our goal is to find the time spent by node in each of these states, so we can figure out the reason behind scheduler's inadequate performance. To estimate the time, we find out long term probabilities of the node being in any of the possible states. We accomplish this by computing steady state probabilities of the markov model.

3.3 Steady State Equations

Let $\pi_{(i,j,k)}$ be the steady state probability of being in state (i,j,k).

Steady state probability of being in state (imax-1,0,0): This is the first stage in holdoff phase. Transition to this state is possible from states (0,4.X,X) and (0,0,X) with probabilities $\frac{1-P_{hand}}{imax}$ and $\frac{1}{imax}$ respectively. Node transits to next state (imax-2,0,0) with probability 1.

$$\left(\frac{1}{imax}+\frac{1-P_{handshake}}{imax}\right)\cdot\pi\left(imax-1,0,0\right)=1\cdot\pi\left(imax-2,0,0\right); \tag{3.1}$$

Steady state probability of being in states (imax-2,0,0) to (1,0,0): These are the intermediate states in holdoff phase, where node keeps waiting in holdoff phase. However the holdoff counter decrements with probability of 1 to the next value. Node can make this transition from states (0,4.X,X),(0,0,X) and (0,0,i).

$$\left(\frac{1}{imax}+\frac{1-P_{handshake}}{imax}+1\right)\cdot\pi\left(i,0,0\right)=1\cdot\pi\left(i-1,0,0\right); \tag{3.2}$$

$$imax-2\leq i\leq 1$$

Steady state probability of being in state (0,0,0): It is a last stage in holdoff phase where counter value has reached zero. Entry into this state is possible with probability $\frac{1-P_{hand}}{imax}$ from last mesh election state, probability of $\frac{1}{imax}$ from data transmission state or with probability 1 from state (1,0,0). From holdoff phase, node make transition to election phase i.e. state (0,4,1) with 1 probability.

$$\left(\frac{1}{imax}+\frac{1-P_{handshake}}{imax}+1\right)\cdot\pi\left(0,0,0\right)=1\cdot\pi\left(0,4,1\right); \tag{3.3}$$

Steady state probability of being in state (0,4,1): Node moves to mesh election phase after expiry of holdoff counter with probability of 1. If handshake is achieved successfully, it enters the wait state with $P_{handshake}$, otherwise contends once more with chance of $1-P_{handshake}$.

$$1\cdot\pi(0,4,1)=(1-P_{handshake})\cdot\pi(0,8,2)+P_{handshake}\cdot\pi(0,0,1); \tag{3.4}$$

Steady state probability of being in states (0,8,2) to (0,4.(X-1),X-1): These states are reached with the probability of unsuccessful handshake i.e. $1-P_{handshake}$. The two possible transitions from these states are same like previous case.

$$(1-P_{handshake})\cdot\pi(0,j,k)=(1-P_{handshake})\cdot\pi(0,j,k+1)+P_{handshake}\cdot\pi(0,0,k); \tag{3.5}$$

$$j=4.k, 1<k<X$$

Steady state probability of being in state (0,4.X,X): This is the last possible state upto which node can wait for successful handshake. If three-way handshake is accomplished, node start sending data by entering state (0,0,X). On the contrary, it will chose any random holdoff exponent value between 0 to imax-1, and make transition to that state.

$$(1-P_{handshake})\cdot\pi(0,4\cdot X,X) = P_{handshake}\cdot\pi(0,0,X)+\sum_{i=0}^{imax-1}\left(\frac{1-P_{handshake}}{imax}\cdot\pi(i,0,0)\right); \tag{3.6}$$

Steady state probability of being in state (0,0,1): Upon successful completion of handshake in first frame, node make transition to this state and waits till the X frame arrives for data transmission.

$$P_{handshake}\cdot\pi(0,0,1) = 1\cdot\pi(0,0,2); \tag{3.7}$$

Steady state probability of being in states (0,0,2) to (0,0,X-1): These states represent increase in number of frames in wait state. Following data transmission, node chooses random holdoff counter value between 0 to imax-1 and moves to holdoff phase.

$$(1+P_{handshake})\cdot\pi(0,0,X) = \sum_{i=0}^{imax-1}\left(\frac{1}{imax}\cdot\pi(i,0,0)\right); \tag{3.8}$$

3.4 Model Parameters

The developed model relies on three important parameters, the value of holdoff exponent *'exp'*, the number of requested frame *'X'* and the probability of handshake $P_{handshake}$. Parameters *'exp'* and *'X'* are decided by the node during operation, whereas $P_{handshake}$ is an unknown parameter. In the following section we will model the scheduler performance to find out the probability of successful handshake.

3.5 Probability of Handshake

Consider the scenario shown in Figure 3.2. When a node has data to send at the current time, it first observes its schedule and the schedule of neighbouring nodes to find data minislots that are not yet reserved for any data communication. Suppose it finds *Nth* minislot as the first available slot in the X frame. Both requester and granter have to complete the three-way handshake before this frame arrives, to make data transmission possible for this *Nth* minislot. Handshake messages of both "request" and "grant" are sent in MSH-DSCH message. Actual transmission time of MSH-DSCH message

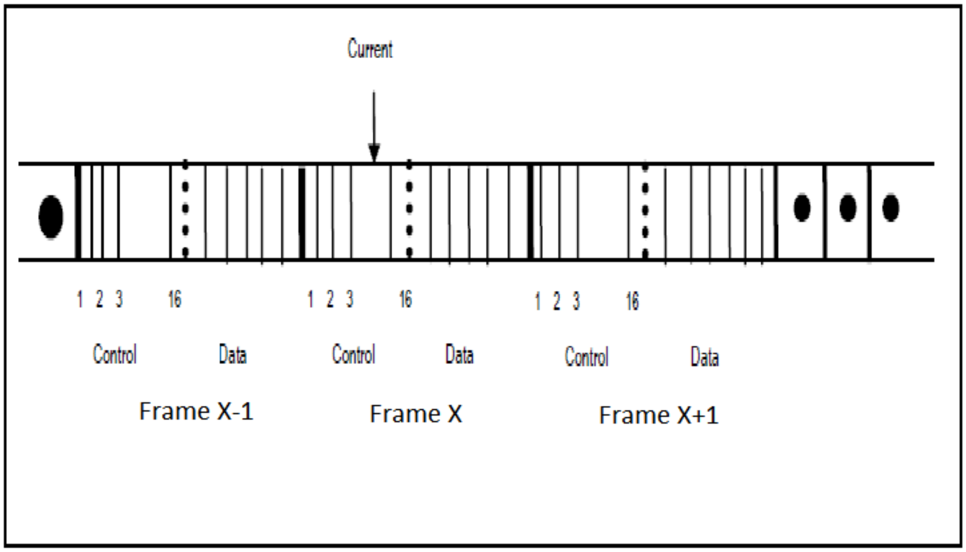


Figure 3.2: Scenario

is decided by *Mesh Election* algorithm. *Mesh Election* is a pseudo-random election algorithm, where decision is taken based upon competing Node IDs and their TempXmt Time. It is a fair election procedure, where no node is preferred over other. Nodes keep running the *Mesh Election* until they get a transmission opportunity to send MSH-DSCH message.Time spent by a node until it wins a mesh election is undetermined. A node could either win mesh election for the very first time or it may keep competing for several slots before it can actually win.

A frame is composed of both control and data slots. So, meanwhile three-way handshake is completed, many data slots must have been passed during that time. Hence, there is a possibility that requester and granter will spend too much time in completion of three-way handshake, so that the requested data minislot passes away without requisition.

Our task is to compute the probability of successfully completing three-way handshake both by requester and granter. Simply we find the probability of transmitting request, grant and grant-confirm before the arrival of data minislot in time domain.We will also compute the probability that out of the 16 data minislot requests, at least 1 data slots range is available at granter for transmission to take place.

3.5.1 Two-Hop Neighbourhood

For modelling the system performance, total number of nodes in two hop neighbourhood is a crucial parameter. Number of two hop neighbours is believed to have a very strong impact on the system performance.To find this, we analyse the network under typical modelling assumptions. It is supposed that mesh nodes topology follow a random uniform distribution. Let total number of nodes in the network to be M and system area in which these M nodes are placed is represented as A. C. Bettstetter[18] had investigated the expected number of neighbours of a node in a multi hop distributed network. Each node is assumed to have uniform radio range r_o. Node density i.e. number of nodes per unit area is found to be

$$NodeDensity = \rho = \frac{M}{A}$$

Area covered by each node is a function of its radio range and calculated as

$$A_o = \pi r_o^2$$

Assume degree of a node i.e number of 1-hop neighbours is represented as n. Expected degree is found to be number of nodes within the radio range of node i.e. product of node density and area under node radio's range, which is

$$n = E\left(degree\right) = \rho \cdot \pi r_o^2 \tag{3.9}$$

Two-hop neighbourhood of a node is represented by a set N' and number of two-hop neighbours is calculated as:

$$\mid N' \mid = n \times n \tag{3.10}$$

In distributed scheduling,nodes' two-hop neighbourhood has prime importance as all the scheduling has to take place between only neighbours that are two-hop away. Therefore from now on, we will use following notations:

- b_1 : Number of 2-hop neighbours that have been allocated next transmission time.
- b_2 : Number of 2-hop neighbours that are in holdoff state.
- k : Number of 2-hop neighbours that are contending with a reference node in Mesh Election algorithm.

3.5.2 Number of nodes in each state

First we will find the number of nodes in each state at the beginning of system followed by the effect of state transitions on these values as system proceeds further.

3.5.2.1 Nodes Assigned Next Transmission Time

When the system starts, no node will have next transmission time allocated yet. So value of b_1 is initialized as zero in the beginning of system. In every transmission opportunity of control subframe, a mesh election is held and 1 node out of competing nodes is allocated transmission time. Hence, b_1 is incremented by 1 in each control slot. i.e.

3.5.2.2 Nodes in Holdoff State

All the nodes will be in holdoff state, as the system initialize. Therefore b_2 is equal to $\mid N' \mid$

3.5.2.3 Nodes in Contention

Number of contending neighbours can be found out by subtracting neighbouring nodes that are in holdoff state or have been allocated transmission time from total number of neighbours, i.e.

$$k_1 = \mid N' \mid - b_1 - b_2 \tag{3.11}$$

Now we will figure out the effects of transition on the above mentioned parameters. As we have stated before, a node moves to holdoff state when its data demand is satisfied. We use notation y to refer the number of such nodes whose data demand is fulfilled. Value of y is dependent upon two parameters i.e. nodes demand and data rate of frame. Data rate of a frame is computed as

$$FrameDataRate = Numberofdataslotsinframe \times DatarateofferedbyMCSscheme \tag{3.12}$$

So number of frames, f required to satisfy a single user demand is:

$$Numberofframes = f = \frac{UserDemand}{FrameDataRate} \tag{3.13}$$

As a single user need f frame to complete its data demand. Therefore we can conclude that $\frac{1}{y}$ users will be served in each frame. After every frame, nodes whose demand is satisfied should be subtracted from b_1 and they must be added to b_2). Therefore b_1 and b_2 will attain following values after every frame.

$$b_1 = b_1 - \lfloor y \rfloor \tag{3.14}$$

$$b_2 = b_2 + \lfloor y \rfloor \tag{3.15}$$

Value of y is mapped to smaller following integer, since number of nodes cannot attain fractional values.

Number of contending neighbours will be also changing over time due to transition of nodes from one state to other. Some of the neighbour nodes might complete their holdoff time and start contending. Such nodes must be added in the system. Similarly, the node that won the last election should be subtracted, or in other words value of b_1 is incremented by 1 in each control slot. Therefore, the number of contending neighbours would be:

$$k = [|N'| - b_1 - b_2] - 1 + \alpha \times b_2 \tag{3.16}$$

Whereα represents the mean number of nodes completing holdoff time in each control slot.

By putting back the value of α, b_1 and b_2 in above equation, some values cancel out. We get to know that number of contending neighbours will be same for any time. So the number of contending neighbours at any time will be

$$k = (n \times n) - b_1 - b_2 \tag{3.17}$$

3.5.3 Probability of Winning Mesh Election

Mesh election algorithm is a pseudo random algorithm, where each participating node has equally likely chance to win. So probability of success, p_s to win mesh election algorithm follows a uniform distribution, means that each contending node has equally likely chance to win. Therefore probability of winning a mesh election would be

$$p_s = \frac{1}{k+1} \tag{3.18}$$

which means that one node out of total competing nodes will min the election. Competing neighbours are represented by k and 1 is added to k to show that reference node is also competing.

Mesh election satisfies the definition of Bernoulli trial. Winner is chosen randomly for each election and possible outcome is either success or failure for any node. Node will keep running the mesh election for a series of control slots until it wins a slot to send MSH-DSCH control message. This pattern follows geometric distribution. Therefore probability that a node wins the mesh election in ith transmission opportunity, after losing in $i-1$ slots is:

$$Pr\left\{X = i\right\} = \left(1 - p_s\right)^{i-1} \cdot p_s \tag{3.19}$$

Now referring back to our main problem, we have to compute the probability of successful three-way handshake before the data minislot arrives. This probability is found to be based on three further parts. As described earlier, both requester and granter have to send MSH-DSCH message to successfully complete the handshake. We will compute probabilities that both requester and granter send MSH-DSCH message before data minislot arrives. Moreover, handshake is successful only when granter has those requested data slots available for data communication.

Assume total time available to complete three-way handshake to be N time units away. Both requester and granter have to send MSH-DSCH messages within these N slots. Further, a requester also has to reply back with grant-confirm message. Therefore, requester and granter have $N-1$ slots to complete handshake. Requester has to send MSH-DSCH before granter or in other words request has to be made before the grant. Hence we make two partitions of these $N-1$ slots. First partition comprises from current time slot to arbitrary jth slot. Second partition includes slots from $j+1$ to $N-1$. Now for ideal case, a requester must send request within the first slots partition and granter should reply back in remaining slots. But since this is a probability based system, so we evaluate the probabilities associated with these ideal cases.

3.5.4 Probability of Request

First of all we are going to find the probability of sending MSH-DSCH by a requester. For ease, we call current slot as slot number 1, such that requester successfully transmit MSH-DSCH message at slot j when it starts competing from slot 1. Value of j is not fixed. It can assume any value between 2 and N-2. We compute the probability by assigning different values to j. Hence probability of sending MSH-DSCH message by a requester at jth slot is

$$Pr_{request} = Pr\{X \leq j\} = \sum_{t=1}^{j} \left(1 - p_s\right)^{t-1} \cdot p_s \tag{3.20}$$

3.5.5 Probability of Grant

Likewise requester, ideal case is that granter must respond back with a grant message before N-1 slots. It should start competing for control slot when it receives bandwidth request message. So requester will start contending from $j+1$ slot and the last slot will be N-1 to ensure timely response. Therefore probability of sending MSH-DSCH message by a granter between *j+1* and *N-1* time units is:

$$Pr_{grant} = Pr\{j+1 < X \leq N-1\} = \sum_{t=j+1}^{N-1} (1-p_s)^{t-1} \cdot p_s \tag{3.21}$$

3.5.6 Probability of Requested Data Slots Availability

According to standard, A requester can show availability upto 16 data slot ranges. Granter checks its schedule and schedule of its neighbouring nodes to verify that data transmission is possible or not. If neither of 16 data slot ranges is available free at granter, then data transmission is impossible and requester has to request again for soome different data slots. But even if 1 data slot range is available such that no transmissions are scheduled on this range neither by granter itself nor by any of two hop neighbours of granter, then data transmission is possible and three-way handshake could be completed successfully.

If we say that 1st requested data slot range is unavailable at granter, it means that it must have been occupied by one of its those two hop neighbours, who have been allocated transmission time i.e. neighbour nodes in $b-1$ state. We mention this probability as p'_1.

$$p'_1 = \frac{1}{b_1} \tag{3.22}$$

Since requester can show availabilities upto 16 slots. By assuming independence we can say probability that status of any one of 16 requested slots is unavailable would be equal. Therefore we can say,

$$p'_1 = p'_2 = p'_3 = p'_4 = p'_5 = p'_6 = p'_7 = p'_8 = p'_9 = p'_{10} = p'_{11} = p'_{12} = p'_{13} = p'_{14} = p'_{15} = p'_{16} \tag{3.23}$$

Subscripts from 1 to 16 represents 16 requested data slot ranges. Granter will not reply back with Grant MSH-DSCH if all of the requested 16 slots are unavailable. We use symbol P' to represent probability that all the 16 requested ranges are unavailable.

$$P' = p'_1 \cdot p'_2 \cdot p'_3 \cdots p'_{16} \tag{3.24}$$

Putting back values of p'_1 to p'_{16} in above equation yields

$$P' = \frac{1}{b_1} \cdot \frac{1}{b_1} \cdot \frac{1}{b_1} \cdots \frac{1}{b_1} \tag{3.25}$$

this implies

$$P' = \left(\frac{1}{b_1}\right)^{16} \tag{3.26}$$

For successful completion of handshake, at least one of the 16 requested data slots must be available at granter, hence

$$Pr_{available} = 1 - P' \tag{3.27}$$

So probability of successfully completing three-way handshake before data slot arrives would be the product of probability of sending request in time with probability of sending grant in time and probability that one of requested data slot is available.

$$Pr_{handshake} = Pr_{request} \cdot Pr_{grant} \cdot Pr_{available} \tag{3.28}$$

Eventually we get,

$$Pr_{handshake} = \left[\sum_{t=1}^{j} (1 - p_s)^{t-1} \cdot p_s\right] \cdot \left[\sum_{s=j+1}^{N-1} (1 - p_s)^{s-1} \cdot p_s\right] \cdot [1 - P'] \tag{3.29}$$

Chapter 4

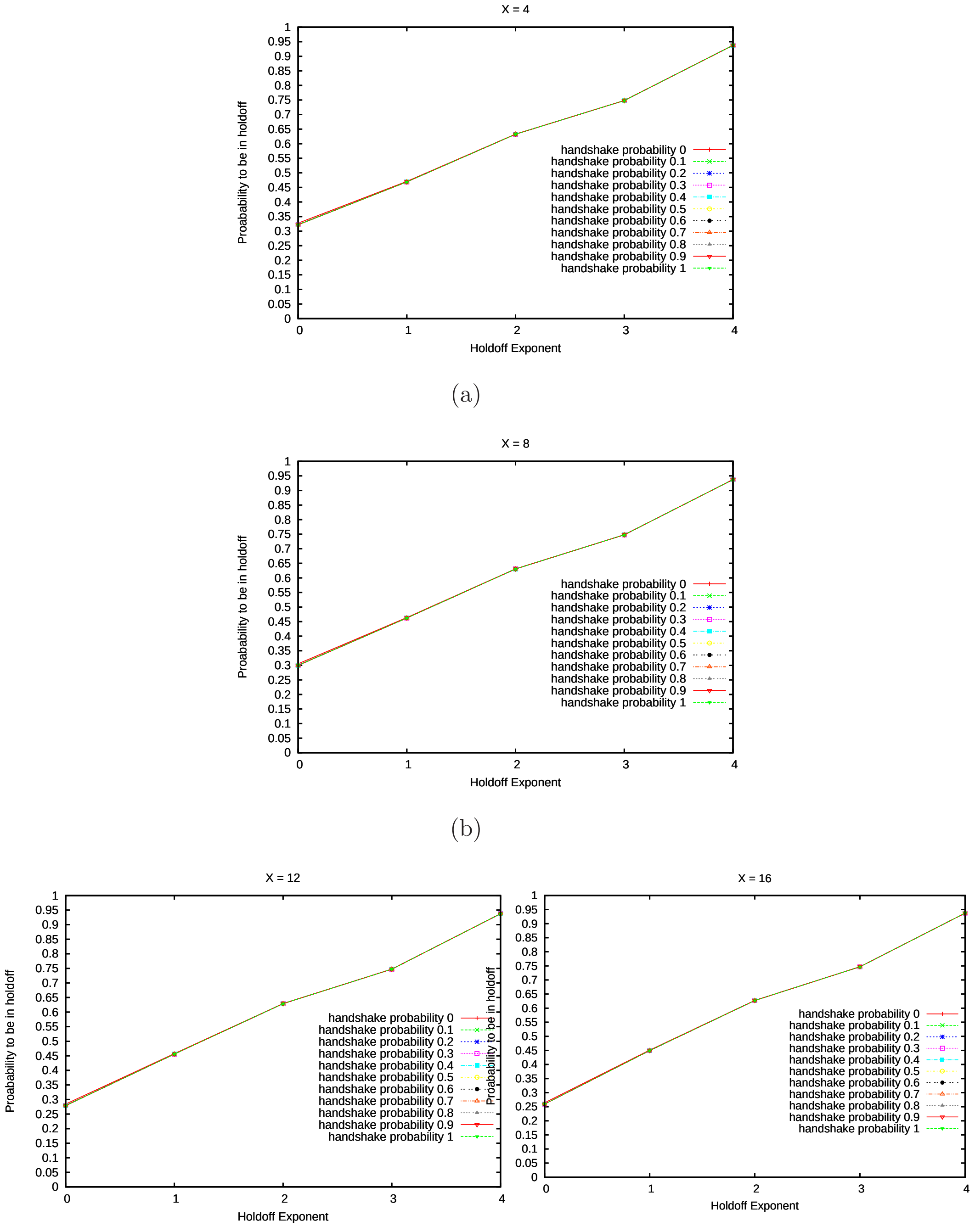
Analytical Evaluation

In this chapter we are going to evaluate the distributed scheduler performance in light of our proposed analytical model. We find out the probability of a single node being in different states by varying the parameters i.e. holdoff exponent "*exp*", values of frame in which requested data slot resides "X" and probability of handshake "$P_{handshake}$". Main goal of this thesis is to find the parameters that have impact on scheduler's performance. By analytical evaluation we will be able to find optimum values of these affecting parameters.

4.1 Probability of being in holdoff

A node's probability of being in holdoff is computed for values of X as 4, 8, 12 and 16 respectively, as shown in figures 4.1(a), 4.1(b), 4.1(c) and 4.1(d). All the values of $P_{handshake}$ yields same result for different values of *exp*. Holdoff probability increases drastically with increase in values of *exp*.

For *exp* values 1 to 4, it assumes values 0.44, 0.62, 0.75 and 0.94 respectively. Only for *exp* value 0, the probability of being in holdoff decreases with increase in value of X (from 0.32 to 0.25) otherwise X doesn't have much impact on holdoff probability. So we can say that probability of being in holdoff is mainly determined by the value of *exp*. Other parameters i.e. $P_{handshake}$ and X don't have much impact on this probability.



(a)

(b)

(c)

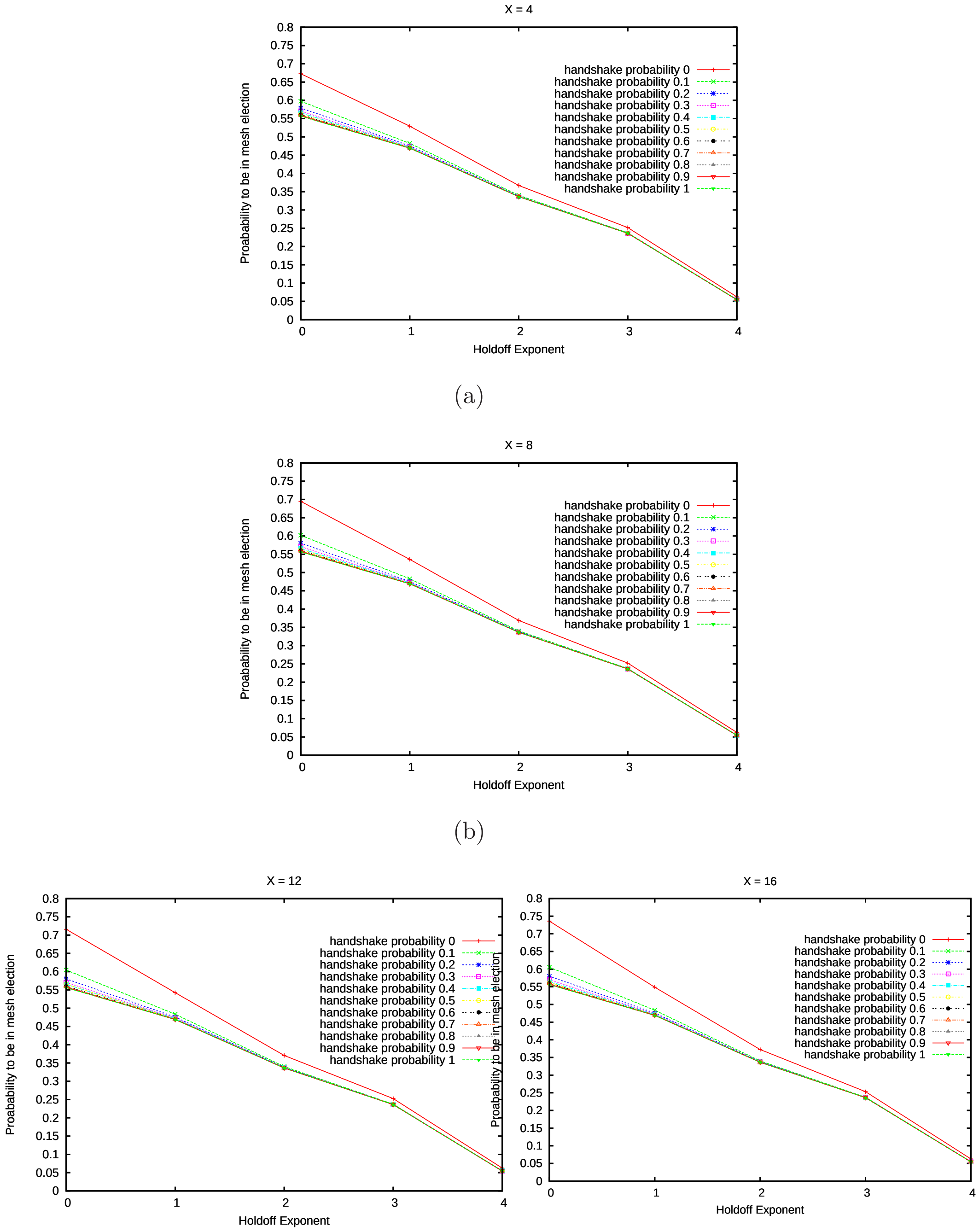
(d)

Figure 4.1: Probability of a node being in holdoff

4.2 Probability of being in election

We observe the node's probability of being in election for different values of X as depicted in Figure 4.2. Probability of being in election is computed for all possible combinations of holdoff exponent and $P_{handshake}$. It is easily observable that by increasing the value of holdoff exponent i.e *exp*, time spent in election phase decreases. More time is spent in holdoff phase rather than being in contention for TO (see figure 4.1).

We can notice in figures 4.2(a) , 4.2(b) , 4.2(c) and 4.2(d), all the values of handshake probability produce almost same results against different values of X except for 0 handshake probability. Moreover when *exp* is set to zero, only then $P_{handshake}$ seems significant. With lower handshake probabilities, chances of being in election state are higher. Hence it is found that value of holdoff exponent has the most important effect on the node's probability of being in election comparable to other two parameters.



(a)

(b)

(c)

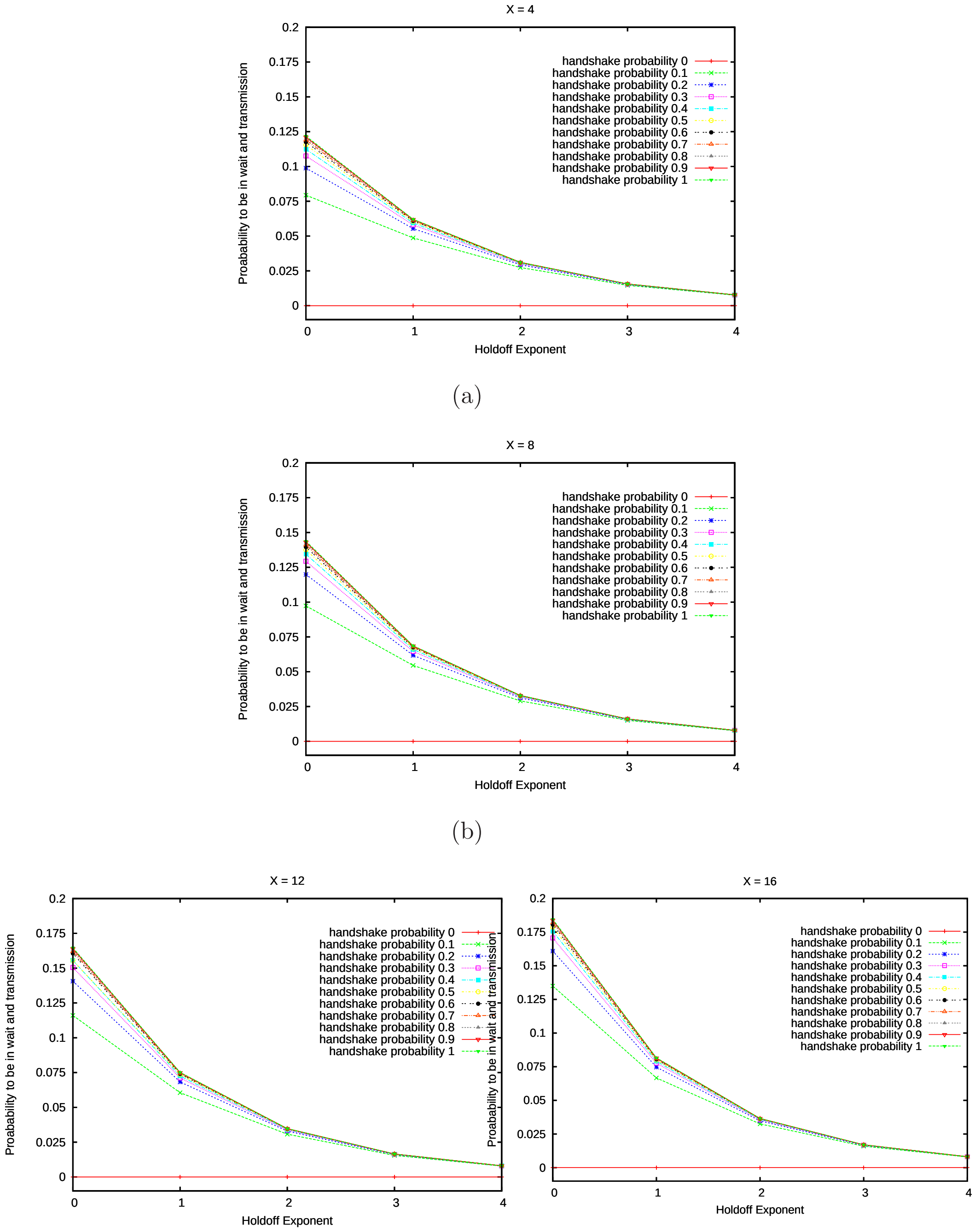
(d)

Figure 4.2: Probability of a node being in election

4.3 Probability of being in wait and transmission state

Amount of time spent by a node in wait and data transmission phase is estimated in figure 4.3. The most interesting observation is for zero value of $P_{handshake}$, a node doesn't enter the wait stage, and more the chances of handshake, higher is the probability to be in this state.

For *exp* values 1 to 4, $P_{handshake}$ is not having much effect but when *exp* is zero, the $P_{handshake}$ produce an impact on node's probability to enter wait state. The probability to be in wait state is directly proportional to the value of X and it increases from 0.1 to 0.2 with change in X from 4 to 16 (see figures 4.3(a) to 4.3(d)).



(a)

(b)

(c)

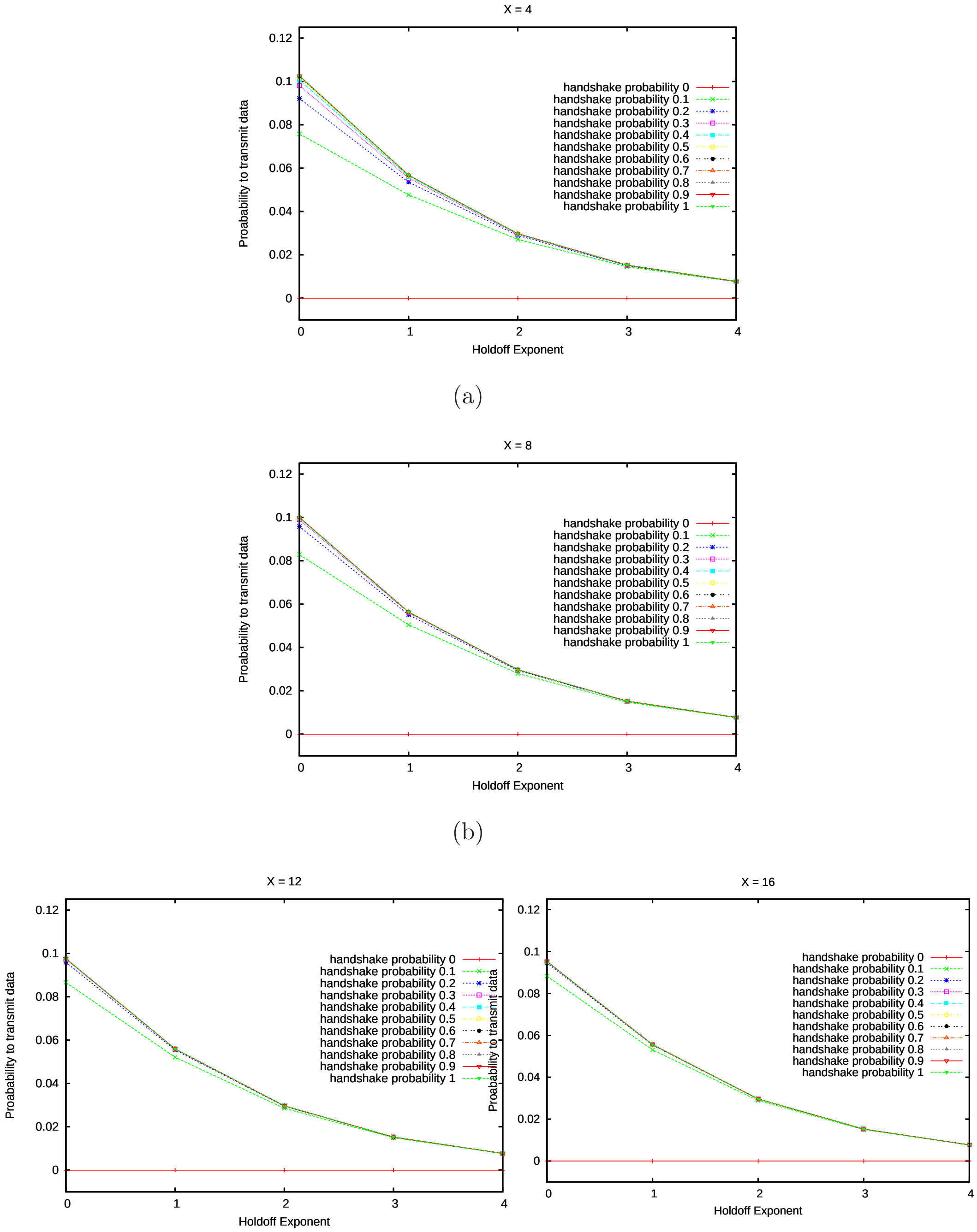
(d)

Figure 4.3: Probability of a node being in wait and data transmission phase

4.4 Probability to transmit data

This is the most important phase with regards to our thesis goal. In figure 4.4 we are estimating a node's chances to actually transmit the data. With increase in *exp*, the probability to transmit data decreases exponentially. The probability is highest around 0.1 for *exp = 0* and lowest(0.01) at *exp = 4*.

Moreover, higher the value of $P_{handshake}$, more are the chances to transmit data. By increasing the value of X, node's chances to transmit data are further reduced (from 0.1 to 0.09) as more time is then spent in waiting for frame *"X"*. So it is easily observable that all the three factors are playing part in determining node's chances to transmit data where *exp* value is being the most crucial of all.



(a)

(b)

(c)

(d)

Figure 4.4: Probability of a node to transmit data

4.5 Conclusion

By analytical evaluation we come across to find that the holdoff exponent value is the most critical parameter in finding node chances of being in any of the possible four states. The other two parameters $P_{handshake}$ and X effect the probabilities only when holdoff exponent value is zero.

So we conclude the optimal value of holdoff exponent to be set as zero for distributed scheduler. In addition, higher the chances to transmit data, more is the effective throughput of a node. So in order to increase node's chances to transmit data, the value of X should be chosen as smaller as possible. The $P_{handshake}$ can not be directly fed into system , it is computed by knowing the number of contending two-hop neighbours of the node, so ideal value is found to be 2, which causes $P_{handshake}$ to assume value of 0.5.

Chapter 5

Conclusion

5.1 Conclusion

Wireless mesh networks are easy to deploy and cost effective solution for meeting the escalating bandwidth demand of today world. Realizing the benefits of wireless mesh networks, WiMAX also supports mesh mode of operation. Mesh mode uses OFDM modulation with Time Division Multiplexing (TDM). Two types of scheduling are supported in mesh mode: centralized and distributed. Centralized scheduling is quite similar to PMP mode, where BS is responsible for all types of network operations. Hence, it is unable to exploit all the advantages a mesh network can provide. Therefore distributed scheduling is preferred over centralized scheduling.

In distributed scheduling, nodes coordinate their transmissions in decentralized manner. Within two-hop neighbourhood, only one node can transmit at a time to avoid collisions and interference. Transmission timings of a node are decided by pseudo-random algorithm called "Mesh Election". Node has to complete the three-way handshake before data transmission.

Despite the benefits promised by WiMAX mesh mode, it is not much popular in operators due to lower throughput achieved by nodes and lower number of users supported by system. We investigate reasons behind the scheduler's inefficient performance by developing a 2-D Markov Chain which models all the possible states a node can attain. These states are found to be holdoff, contending in mesh election, wait for the arrival of requested data slot and data transmission.

By analytical evaluation we estimate the time spent by node in all states by finding node's probability to be in each state. It is found that node is in data transmission state for not more than 10% of time, which is quite less as comparable to time spent in other phases. Most of the time is spent in holdoff from 24% to 94%. Time spent in election phase varies from 75% to 6% depending on the values of effecting parameters. Lesser the time spent in data transmission, lesser is the throughput, this is the reason behind insufficient throughputs level.

Literature survey reveals that higher values of holdoff exponent is the only parameter responsible for scheduler's inefficient performance. Our study confirms this finding along with reliance of scheduler performance on two other parameters also. These parameters are handshake probability and how far is the requested frame from current time. These two parameters also have some impact, though its quite insignificant as comparable to the effect of holdoff exponent on throughput achievement.

We find the optimal value of holdoff exponent to be zero. To increase the node chances to transmit data, request for earliest data slot should be made. Ideal data slot should not be more than 4 frames away from current time slot. Nearer value of data slot reduces the time spent in wait state, and hence there are more chances for data transmission. Value of handshake probability can be computed by knowing the number of contending two-hop neighbours. This factor is uncontrollable, but handshake probability values greater than 0.1 yields improved chances for data transmission.

5.2 Future Work

In future, simulations can be done to further illustrate the results. The value of holdoff timer which is computed as $2^{(XHE+4)}$ in standard, is the main reason behind the scheduler's inefficient performance. In future, this model can be extended to find the optimal value of holdoff timer.

Bibliography

[1] "IEEE standard for local and metropolitan area networks part 16: air interface for fixed broadband wireless access systems", *IEEE Std 802.16-2004*, 2004

[2] "Frame Structure and Bandwidth Management in the MESH Mode".WIMAX MADE SIMPLE. September 13, 2010. http://wimax-made-simple.blogspot.com/2010/09/frame-structure-and-bandwidth.html .

[3] N. Abu Ali, A. Taha, H. Hassanein, H. Mouftah. "IEEE 802.16 Mesh Schedulers: Issues and Design Challenges". *IEEE Network Magazine*, pp. 58-65. Jan-Feb, 2008.

[4] M. Kas, B. Yargicoglu, I. Korpeoglu, E. Karasan. "A Survey on Scheduling in IEEE 802.16 Mesh Mode", *IEEE Communications Surveys Tutorials*, Vol. 12, No. 2, Second Quarter 2010.

[5] M. Cao, W. Ma, Q. Zhang, X. Wang. "Analysis of IEEE 802.16 Mesh Mode Scheduler Performance," *IEEE Transactions on Wireless Communications*, vol.6, no.4, pp.1455-1464, April 2007

[6] M Cao, W. C. Ma, Q. Zhang, X. D. Wang, W. W. Zhu. "Modelling and performance analysis of the distributed scheduler in IEEE 802.16 mesh mode". *Proceedings of the 6th ACM international symposium on mobile ad hoc networking and computing*, pp. 78-89, May 2005

[7] V. Loscri, "A queue based dynamic approach for the Coordinated distributed scheduler of the IEEE 802.16," *IEEE Symposium on Computers and Communications, 2008. ISCC 2008.* , pp.423-428, 6-9 July 2008

[8] Ceclia de A. Castro Cesar et al. "Adjusting Holdoff Time of the IEEE 802.16 in Mesh Mode," *3rd International Conference New Technologies, Mobility and Security (NTMS), 2009* pp.1-5 , 2009

[9] H. Zhu, Y. Tang, I. Chlamtac, "Unified collision-free coordinated distributed scheduling (CF-CDS) in IEEE 802.16 mesh networks," *IEEE Transactions on Wireless Communications*, vol.7, no.10, pp.3889-3903, October 2008

[10] R.Krenz, "IEEE 802.16 Wireless Mesh Networks Capacity Estimation Using Collision Domains ", *2009 Second International Conference on Advances in Mesh Networks*

[11] N. Bayer, D. Sivchenko, B. Xu, V. Rakocevic, and J. Habermann, "Transmission Timing of Signaling Messages in IEEE 802.16 Based Mesh Networks," *Proc. European Wireless,* pp.2-5, 2006.

[12] N. Bayer, B. Xu, V. Rakocevic, and J. Habermann, "Improving the Performance of the Distributed Scheduler in IEEE 802.16 Mesh Networks," *IEEE 65th Vehicular Technology Conference, 2007. VTC2007-Spring.,* pp. 1193-1197, 2007

[13] S. Wang, C. Lin, H. Chu, T. Hsu, K. Fang , "Improving the Performances of Distributed Coordinated Scheduling in IEEE 802.16 Mesh Networks," *IEEE Transactions on Vehicular Technology,*, vol.57, no.4, pp.2531-2547, July 2008

[14] C. Cicconetti, A. Erta, L. Lenzini, and E. Mingozzi, "Performance Evaluation of the Mesh Election Procedure of IEEE 802.16/WiMAX," *Proceedings of the 10th ACM Symposium on Modeling, Analysis, and Simulation of Wireless and Mobile Systems,* pp. 323-327, 2007

[15] M. Zhang, S. Wang, T. He, "Study on Coordinated Distributed Scheduling in WiMAX Mesh Network," *5th International Conference on Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09.*, pp.1-4, 24-26 Sept. 2009

[16] Y.Zhang, X. Gao, X. You , "The IEEE 802.16 mesh mode coordinated distributed scheduling can be collision free," *IEEE Transactions on Wireless Communications*, vol.7, no.12, pp.5161-5165, December 2008

[17] S. Chakraborty, D.K. Sanyal, A. Chakraborty, A. Ghosh, S. Chattopadhyay, M. Chattopadhyay, "Tuning Holdoff Exponents for Performance Optimization in IEEE 802.16 Mesh Distributed Coordinated Scheduler", *IEEE 2nd International Conference on Computer and Automation Engineering (ICCAE), 2010*

[18] C. Bettstetter,"On the Minimum Node Degree and Connectivity of a Wireless Multihop Network", *In Proceedings of ACM MobiHoc*, pages 80-91, Lausanne Switzerland, 2002.