Performance Evaluation of WiFiRe using OPNET

Dissertation

submitted in partial fulfillment of the requirements
for the degree of

Master of Technology

by

Venkat Reddy M
(Roll no. 05329038)

under the guidance of

Prof. Sridhar Iyer
and
Prof. Varsha Apte

Department of Computer Science & Engineering
Indian Institute of Technology Bombay

2007

Dissertation Approval Sheet

This is to certify that the dissertation entitled

Performance Evaluation of WiFiRe using OPNET

by

Venkat Reddy M

(Roll no. 05329038)

is approved for the degree of **Master of Technology**.

Prof. Sridhar Iyer
(Supervisor)

Prof. Varsha Apte
(Co-Supervisor)

Prof. Anirudha Sahoo
(Internal Examiner)

Dr. Vijay T. Raisinghani
(External Examiner)

Prof. V. M. Gadre
(Chairperson)

Date: ______________

Place: ______________

INDIAN INSTITUTE OF TECHNOLOGY BOMBAY
CERTIFICATE OF COURSE WORK

This is to certify that **Mr. Venkat Reddy** was admitted to the candidacy of the M.Tech. Degree and has successfully completed all the courses required for the M.Tech. Programme. The details of the course work done are given below.

Sr.No.	Course No.	Course Name	Credits
		Semester 1 (Jul – Nov 2005)	
1.	IT601	Mobile Computing	6
2.	IT605	Computer Networks	6
3.	IT653	Network Security	6
4.	IT619	IT Foundation Laboratory	8
5.	IT623	Foundation course of IT - Part II	6
6.	HS699	Communication and Presentation Skills (P/NP)	4
		Semester 2 (Jan – Apr 2006)	
7.	IT610	Quality of Service in Networks	6
8.	CS640	Natural Computating	6
9.	IT628	Information Technology Project Management	6
10.	IT680	Systems Lab.	6
11	IT694	Seminar	4
		Semester 3 (Jul – Nov 2006)	
12.	CS681	Performance Evaluation of Computer Systems and Networks(Audit)	6
13.	HSS	Classical Indian Thoughts In Modern Times (Institute Elective)	6
		M.Tech. Project	
14.	IT696	M.Tech. Project Stage - I (Jul 2006)	18
15.	IT697	M.Tech. Project Stage - II (Jan 2007)	30
16.	IT698	M.Tech. Project Stage - III (Jul 2007)	42

I.I.T. Bombay Dy. Registrar(Academic)

Dated:

Acknowledgements

I take this opportunity to express my sincere gratitude for **Prof. Sridhar Iyer** and **Prof. Varsha Apte** for their constant support and encouragement. Their excellent guidance has been instrumental in making this project work a success.

I would also like to thank my colleagues **Shravan**, **Janak**, **Ranjith** and **Moni** for helpful discussions in the initial part of my project, and the KReSIT department for providing me world class computing infrastructure.

I would also like to thank my **family** and **friends** especially the entire **MTech Batch**, who have been a source of encouragement and inspiration throughout the duration of the project.

Last but not the least, I would like to thank the entire KReSIT family for making my stay at IIT Bombay a memorable one.

Venkat Reddy M

I. I. T. Bombay

July 11^{th}, 2007

Abstract

The goal of WiFiRe is to provide broadband Internet services in the rural areas. The key idea in WiFiRe is to replace the 802.11b MAC mechanisms, with something more suitable for long-range communication, while using 802.11b PHY. WiFiRe network topology is a star topology, a System (S) with set of Base Stations (BS) with sectorized antennas at the fiber Point of Presence (PoP) and Subscriber Terminals in the surrounding villages with directional antennas. WiFiRe Cell is divided into sectors and each sector will be covered by a BS . WiFiRe has one common medium access (MAC) controller for all the sectors, to co-ordinate the medium access among them. There are multiple STs in each sector. The multiple access mechanism is time division duplexed, multi-sector TDM (TDD-MSTDM) scheduling of slots.

This protocol is modeled in OPNET. In this model we have implemented a Round Robin slot scheduling algorithm and a Connection Admission Control mechanism. We have simulated this protocol in different scenarios with different MAC configurations. We have simulated UGS, rtPS, nrtPS and BE service classes with different configurations, like changing polling time, changing slot length, etc. We have compared throughput and service delays of different service classes in different scenarios. By simulation results, we have determined the best slot length and maximum number of users supported by each BS for different type of service classes.

Contents

Acknowledgements **vii**

Abstract **ix**

List of figures **xiii**

List of tables **xv**

Abbreviations **xvii**

1 Introduction and Motivation **1**

1.1 802.11b MAC 2

1.2 *WiFiRe* protocol 2

1.3 Problem Statement 4

1.4 Thesis Outline 4

2 New Model Design In OPNET **5**

2.1 Simulation in OPNET 5

2.1.1 Modeling Domains In OPNET 5

2.1.2 Modeling Wireless Network 8

3 WiFiRe Protocol **11**

3.1 Introduction to WiFiRe 11

3.2 Protocol Phases 13

3.2.1 Ranging 13

3.2.2 Registration 14

3.2.3 Connection Management 14

3.3 WiFiRe MAC Services . . . 16
3.3.1 Unsolicited Grant Service . . . 17
3.3.2 Real-time Polling Service . . . 17
3.3.3 Non real time Polling Service . . . 17
3.3.4 Best Effort Service . . . 18
3.4 MAC PDU Formats . . . 18
3.5 WiFiRe MAC Operation Details . . . 19
3.6 Modules simulated in WiFiRe OPNET model . . . 20

4 WiFiRe Model Design in OPNET 21
4.1 WiFiRe Model in OPNET . . . 21
4.2 Process models . . . 23
4.2.1 WiFiRe Common MAC Process model . . . 23
4.2.2 ST control child process model . . . 25
4.2.3 BS control child process model . . . 27
4.3 Classifier . . . 27
4.4 Admission Control in WiFiRe . . . 29
4.5 Round Robin Slot Scheduling . . . 30
4.6 Model Features and Limitations . . . 32

5 Simulation Experiments and Results 35
5.1 Experiments . . . 35
5.1.1 VoIP user Scenario . . . 35
5.1.2 Video Conference user Scenario . . . 41
5.1.3 nrtPS Scenario . . . 45

6 Conclusion and Future Work 49
6.1 Conclusion . . . 49
6.2 Future Work . . . 49

Bibliography 51

List of Figures

2.1 Relationship of Hierarchical Levels in OPNET Models 6
2.2 Simple node model with TCP/IP stock in OPNET 7
2.3 ethernet process model in OPNET 8
2.4 Radio Transceiver Pipeline Execution Sequence for One Transmission . . . 9

3.1 WiFiRe Network Configuration 12
3.2 Parallel transmissions in a six sector system 13
3.3 Connection Creation in WiFiRe 15
3.4 Connection Termination in WiFiRe 16
3.5 WiFiRe MAC PDU 18
3.6 WiFiRe Generic MAC Header 18
3.7 WiFiRe Beacon Header 18
3.8 WiFiRe Beacon Message 19

4.1 WiFiRe Model design block diagram 21
4.2 Common MAC process model 23
4.3 WiFiRe BS node model 24
4.4 WiFiRe SS node model 24
4.5 ST control child process model 27
4.6 BS control child process model 27
4.7 Classification Module Block diagram 28
4.8 Classifier Setup 29
4.9 Service flow parameters setup 29
4.10 Admission Control flow chart 30

5.1 Simulation setup with one ST, and one BS 37

5.2 UGS statistics with slot size 32 μs and VoIP application with G 729 codec, 20 ms sampling rate, 8 kbps coding rate. 39
5.3 UGS statistics with slot length 32μs and VoIP application with G729 codec, sampling rate 10 ms and 8 kbps coding rate. 40
5.4 VoIP statistics at application layer with G729 codec with 10 ms sampling rate. 41
5.5 UGS statistics with slot size 40 μs and VoIP application with G729 codec, 10 ms sampling rate and 8 kbps coding rate. 42
5.6 rtPS statistics with slot size 32μs. 45
5.7 rtPS statistics with slot size 45μs. 46
5.8 nrtPS statistics with slot size 32 μs and with 2 seconds polling interval. . . 48
5.9 nrtPS queuing and end to end MAC delay comparison with slot size 32μs for different polling intervals. 48

List of Tables

5.1 Common parameters for all experiments 36
5.2 UGS Setup parameters 36
5.3 UGS service flow parameters 36
5.4 rtPS Setup parameters 43
5.5 rtPS service flow parameters 43
5.6 nrtPS Setup parameters 47
5.7 nrtPS service flow parameters 47

Abbreviations

Abbreviations

CID	: Connection Identifier
BW Req	: Bandwidth Request
PoP	: fiber Point of Presence
S	: WiFiRe System
BS	: Base Station
ST	: Subscriber Station
DSA Req	: Dynamic Service Addition Request
DSD Req	: Dynamic Service Deletion Request
DSC Req	: Dynamic Service Change Request
TDD	: Time Division Duplex
TDM	: Time Division Multiple access
TDD-MSTDM	: Time Division Duplex, multi sector TDM
DL-MAP	: Downlink MAP
UL-MAP	: Uplink MAP
BPSK	: Binary Phase Shift Keying
PDU	: Protocol Data Unit
PHY	: Physical Layer
QoS	: Quality of Service
UGS	: Unsolicited Grant Service
rtPS	: Real Time Polling Service
nrtPS	: non-real Time Polling Service
SAP	: Service Access Point
SDU	: Service Data Unit

Chapter 1

Introduction and Motivation

From the last decade, Internet and cellular Telephony's are growing very fast. But much of this development happened in developed countries and metro cities in developing countries like India. But about 70% of India's population, or 750 million people live in villages. So, Internet revolution has not yet started fully, and it is only limited to developed parts of the country. There is a need for Broad Band Internet access in villages for applications such as distance learning, e-governance, market information access, Voice over IP and email, etc. Broadband wireless access (BWA) can become the best way to meet growing business demand for rapid Internet connection and integrated data, voice and video services. BWA can extend fiber optic networks and provide more capacity than cable networks or digital subscriber lines (DSL). In rural areas, most of the land is farmland, and typically the houses are in one or two clusters with average village size is 250-300 households, and occupies an area of 5 sq. km [1].

Alternate technologies to provide Internet services in rural areas are optical networks, wireless broad band, cable TV network, and local distribution via ethernet. Amongst all of these, broad band wireless is the best option to provide Internet services in rural areas. With Wireless Broad Band Access, there is no need to setup large scale infrastructure. Cellular technologies may help in quick deployment of the access network. But, like land-lines, their cost structure too is suited only for the developed world. The services and more importantly the equipments, are very costly, and they are not affordable to rural area people. Lowcost technology is essential in rural deployment since the density of users as well as their paying capacity is low.

Wireless Fidelity - Rural Extension (*WiFiRE*)[2] introduces the concept of wireless communication over WiFi IEEE 802.11b physical layer (PHY) and with a MAC protocol which will support long range communication with low cost. 802.11b PHY has better availability of low cost chip sets which can operate on unlicensed 2.4GHz frequency band. *WiFiRe* uses a star topology network, in which main system (S) which is consisted of set of Base Stations (BS) with sectorized antenna, is established at fiber Point of Presence (PoP). In each village one Subscriber Terminal with directional antenna is established. The communication between ST and System (S) is through WiFiRe protocol.

1.1 802.11b MAC

802.11b [3] has a maximum raw data rate of 11 Mbps and uses the same CSMA/CA media access method defined in the original standard. It operates on 2.4Ghz frequency band. 802.11b is usually used in a point-to-multipoint configuration, wherein an access point communicates via an omni-directional antenna with one or more clients that are located in a coverage area around the access point. Typical indoor range is 30 m (100 ft) at 11 Mbps and 90 m (300 ft) at 1 Mbps. With high-gain external antennas, the protocol can also be used in fixed point-to-point arrangements, typically at ranges up to 8 kilometers (5 miles).

WiFi is well suited for small distance communications like in Airports, Shopping malls, Restaurants etc. The attraction of WiFi technology is the de-licensing of its spectrum in many countries, including India. In rural areas, where the spectrum is hardly used, WiFi is an attractive option, provided its limitations when used over a wide-area are overcomed somehow. But there are some disadvantages for using this in long distance communication. Major problem is its CSMA/CA mechanism. It is designed for short distance wireless communication. DCF function does not provide any QoS guarantees, while PCF is inefficient with increase in the number of nodes [4]. In wide area network setup using off-the-shelf WiFi equipment, Medium Access (MAC) efficiency becomes very poor. One solution for this problem is to replace the MAC protocol with one more suited to wide-area deployment.

1.2 *WiFiRe* protocol

WiFiRe [2] stands for WiFi Rural Extension. It takes advantages of WiFi PHY [3] layer and advantages of WiMax [5] MAC layer. PHY layer is same as 802.11b that uses 2.4

GHz channel. WiFiRe replaces 802.11b MAC mechanism (DCF/PCF) with more suitable one for long range communication, while continuing to use 802.11b PHY. WiFiRe is a star topology where a System (S) with set of base Stations with sectorized antennas at the fiber Point of Presence and Subscriber Terminals in the surrounding villages with directional antenna. Both BS and ST are fixed whereas users within ST (e.g. building, house, small campus etc) can be either fixed or mobile, depending upon the internal network being used. The WiFiRe cell is sectored and each sector will be covered by one BS. In WiFiRe, the multiple access mechanism is time division duplex, multi sector TDM (TDD-MSTDM) scheduling of slots. Time is divided into frames. Each frame further subdivided into UL (uplink) and DL (down link) sub frames which need not be of equal durations. Within each sub frame there are multiple slots of equal duration each. In each down link slot one or zero transmissions can take place in each sector. Multiple BS can transmit in their corresponding sector in non-interference manner. Similarly in each UL slot multiple STs from different sectors may simultaneously transmit packets to the BS in non-interference manner. All BS in the system use same WiFi channel for communication with their respected STs, so transmission by one BS may interfere with adjacent sectors. Sectorization of coverage area while using the same frequency channel for all sectors is a key feature in WiFiRe.

WiFiRe has one medium access (MAC) controller for all sector BSś, to coordinate the medium access among them. Beacons are transmitted at the start of each DL segment. The beacon for each sector contains information for time synchronization of the ST(s), information regarding the DL and UL slot allocations (DL-MAP, UL-MAP) for next frame, and other control information. WiFiRe MAC is connection oriented, So before it begins data transfer it will establish one connection of corresponding type by exchanging DSA Req and DSA Resp messages with the BS. After connection setup, ST is allowed to send data in corresponding slots allocated to it. If no slots allocated to the connection in particular ST, then ST is not permitted to send data. BS will schedule all admitted connection and send scheduling information using Beacon messages. BS will allocate slots to the connection based on the flows type, for example if it is UGS flow then BS will allocate admitted number of slots in each and every frame. Chapter 3 will discuss about the WiFiRe protocol in detail.

1.3 Problem Statement

Problem focuses on *Performance Evaluation* of the *WiFiRe* protocol, which primarily focuses on implementing simulation model and conducting evaluation experiments. Most of the evaluation experiments are focused on deciding the values of various parameters of the protocol, like slot size, frame size and DL-UL frame ratio etc . Problem also focuses on finding what is the maximum number of VoIP calls it can support at the same time. Model Implementation focuses on the following modules.

Admission Control :- Implementation of Admission control which will admit connections on first come first serve basis.

MAP generator :- Implementation of Round Robin slot scheduling algorithm which will generate DL-UL MAP.

Beacon processing and traffic scheduling :- Beacon processing and Traffic scheduling need to implement to send data in corresponding slots.

Implementation of UGS, rtPS, nrtPS and BE service flows

1.4 Thesis Outline

The major contributions of this work are :

- WiFiRe model in OPNET.
- Extensive simulation experiments and results to prove good performance of the protocol in different scenarios with different MAC configurations.

In this thesis, chapter 2 discusses how to design a new model in OPNET. In chapter 3, we discuss WiFiRe protocol. In chapter 4, we discuss the WiFiRe simulation model design in OPNET and Round Robin slot scheduler. We then discuss the simulation setup and the results of simulations in chapter 5. Finally we present conclusion and future work in chapter 6.

Chapter 2

New Model Design In OPNET

2.1 Simulation in OPNET

OPNET(Optimized Network Engineering Tool) [6] provides a comprehensive development environment supporting the modeling of communication networks and distributed systems [7]. Both behavior and performance of modeled systems can be analyzed by performing discrete event simulations. The OPNET environment incorporates tools for all phases of a study, including model design, simulation, data collection, and data analysis. OPNET provides many constructs relating to communications and information processing, providing high leverage for modeling of networks and distributed systems. OPNET provides Graphical specification of model Wherever possible, models are entered via graphical editors. These editors provide an intuitive mapping from the modeled system to the OPNET model specification. OPNET provides four tools called editors to develop a new simulation model. These editors, the Network, Node, Process and Parameter Editors, are organized in a hierarchical fashion, which supports the concept of model level reuse. Models developed at one layer can be used by another model at a higher layer.All OPNET simulations automatically incorporate support for analysis via a sophisticated interactive debugger.

2.1.1 Modeling Domains In OPNET

The Network, Node, and Process modeling environments are sometimes referred to as the modeling domains of OPNET because they span all the hierarchical levels of a model.

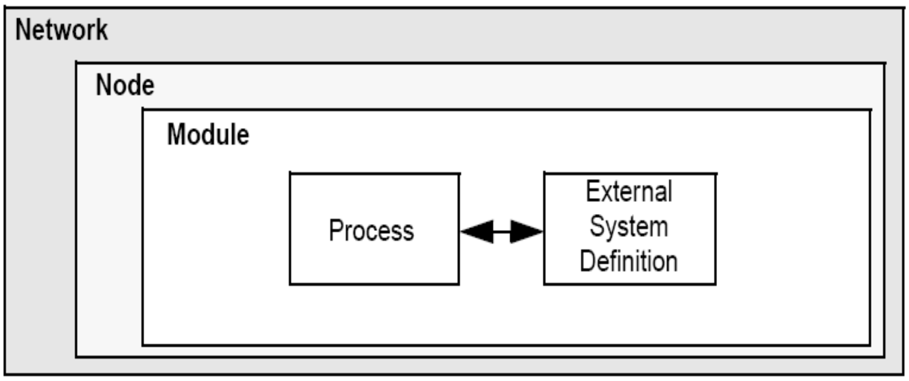


Figure 2.1: Relationship of Hierarchical Levels in OPNET Models

2.1.1.1 Network Domain

The Network Domainś role is to define the topology of a communication network. The communicating entities are called nodes and the specific capabilities of each node are defined by designating their model. Node models are developed using the Node Editor. Within one network model, there may be many nodes that are based on the same node model. The Project Editor can provide a geographic context for network model development. Most nodes require the ability to communicate with other nodes to do their function in a network model. Several different types of communication link architectures are provided to interconnect nodes that communicate with each other. OPNET provides simplex (unidirectional) and duplex (bidirectional) point-to-point links to connect nodes in pairs and a bus link to allow broadcast communication for arbitrarily large sets of fixed nodes.

2.1.1.2 Node Domain

The Node Domain provides for the modeling of communication devices that can be deployed and interconnected at the network level. In OPNET terms, these devices are called nodes, and in the real world they may correspond to various types of computing and communicating equipment such as routers, bridges, workstations, terminals, mainframe computers, file servers, fast packet switches, satellites, and so on.

Node models are developed in the Node Editor and are expressed in terms of smaller building blocks called modules. Some modules offer capability that is substantially predefined and can only be configured through a set of builtin parameters. These include

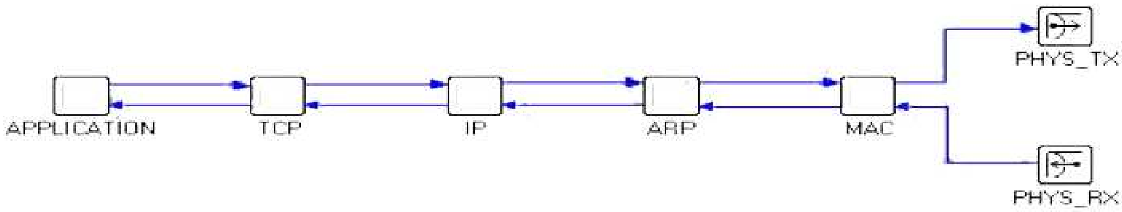


Figure 2.2: Simple node model with TCP/IP stock in OPNET

various transmitters and receivers allowing a node to be attached to communication links in the network domain. Other modules, called processors, queues, and external systems, are highly programmable, their behavior being prescribed by an assigned process model.Three types of connections are provided to support interaction between modules. These are packet streams, statistic wires and logical associations.

2.1.1.3 Process Domain

Processes in OPNET are based on process models that are defined in the Process Editor. Each process that executes in a queue, processor, or esys module is an instance of a particular process model. When a simulation begins, each module has only one process, termed the root process. This process can later create new processes which can in turn create others as well, etc. When a process creates another one, it is termed the new processṕarent; the new process is called the child of the process that created it.

OPNETś Process Editor expresses process models in a language called Proto-C. Proto-C is based on a combination of state transition diagrams (STDs), a library of high-level commands known as Kernel Procedures, and the general facilities of the C or C++ programming language. A process model ś STD defines a set of primary modes or states that the process can enter and, for each state, the conditions that would cause the process to move to another state. The condition needed for a particular change in state to occur and the associated destination state are called a transition. We can also declare state variables to the STDś.

- State Variables: Processes maintain private state variables with named variables of arbitrary data types, including OPNETspecific, general C/C++ language, and user-defined types. This capability allows a process to flexibly maintain counters, routing tables, statistics related to its performance, or messages requiring retransmission.

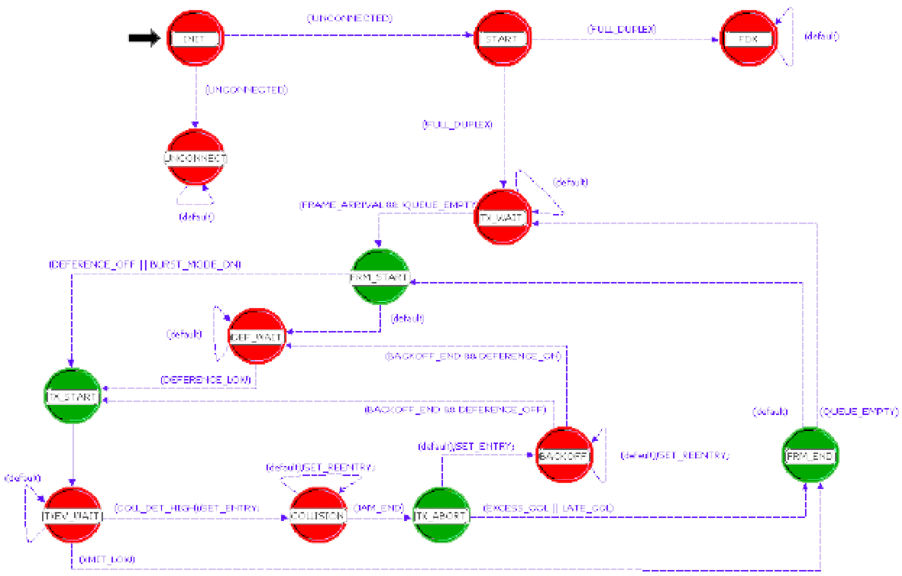


Figure 2.3: ethernet process model in OPNET

- State Executives: Each state of a process can specify arbitrarily complex actions associated with the process entering or leaving that state. These actions, called state executives, are expressed with the full flexibility of the C/C++ language.

- Transition Conditions: Transition condition statements, which determine whether a transition should be traversed, may be expressed as general C/C++ language booleans that make reference to properties of a new interrupt as well as to combinations of state variables.

- Transition Executives: Transitions may specify general actions, called executives, which are implemented each time that they are traversed.

2.1.2 Modeling Wireless Network

2.1.2.1 Antenna Object

Antennas are only used for modeling radio transmission, and affect the simulation only through their association with radio transmitters and receivers. Packet streams are used to couple antennas to receivers and transmitters. A packet stream can lead from a radio transmitter to an antenna, or from an antenna to a radio receiver. No other connections to or from antennas are permitted. The primary purpose of the antenna object is to allow modeling of directional gain at radio receivers and transmitters. Each antenna is assigned a gain pattern which specifies the gain relative to isotropic radiated power as a function of horizontal and vertical plane deflection angles, phi and theta. Thus antenna gain patterns are three dimensional models generated as a function of two variables.

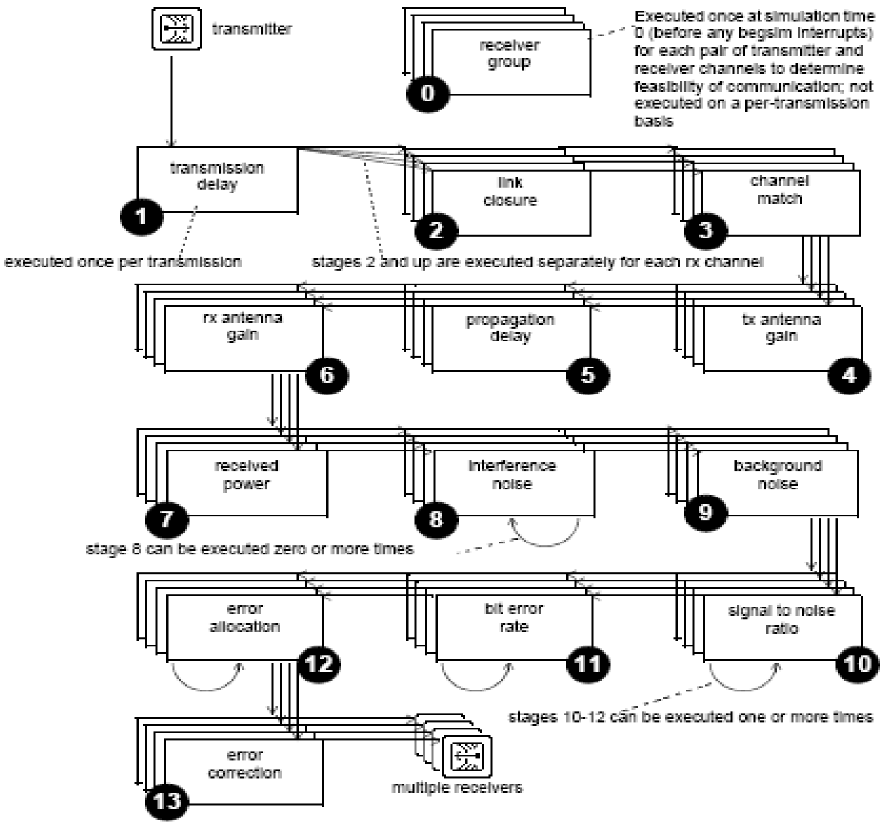


Figure 2.4: Radio Transceiver Pipeline Execution Sequence for One Transmission

2.1.2.2 Radio Receiver Object

Radio receiver objects act as the entry points in a node for packets received on radio communication links. They can contain multiple channel objects, each of which can be accessed by appropriate radio transmitter channels in remote nodes via a radio link.Receivers can have multiple output streams, each one corresponding to a separate radio channel, and each connected to another module in the node.Radio receivers are considered as data sources from the perspective of other modules within the node. The receiver is connected to an antenna module via an input stream.

2.1.2.3 Radio Transmitter Object

Radio transmitter objects act as the exit points of a node for packets transmitted on radio links. They can contain multiple channel objects, each of which can attempt to access appropriate radio receiver channels in remote nodes via a radio link. The processes of modulation and transmission are modeled in this object.Transmitters can have multiple input streams, each one corresponding to a separate radio channel, and each connected to a separate packet stream originating at other modules in the node. Radio transmitter objects are the interface between a node and the radio transmission medium.

2.1.2.4 Radio Transceiver Pipeline to model Radio link

Because radio links provide a broadcast medium, each transmission can potentially affect multiple receivers throughout the network model. The Radio Transceiver Pipeline consists of fourteen stages, most of which must be executed on a per-receiver basis whenever a transmission occurs. However, stage 0 (receiver group) is invoked only once for each pair of transmitter and receiver channels in the network, to establish a static binding between each transmitter channel and the set of receiver channels that it is allowed to communicate with.

Chapter 3

WiFiRe Protocol

3.1 Introduction to WiFiRe

WiFiRe [2] [8] stands for WiFi - Rural extension. It takes advantage from the license free nature of the WiFi spectrum (IEEE 802.11b, 2.4 GHz Band) in order to provide long-range communications (15-20 Kms) for rural areas. The key idea in WiFiRe is to replace the 802.11b MAC mechanisms (DCF/PCF), with something more suitable for long-range communication, because DCF/PCF will not work in long range communications. WiFiRe network topology is a star topology, a Base Station (BS) at the fiber Point of Presence (PoP) and Subscriber Terminals (ST) in the surrounding villages with sectorized antennas at the BS and a directional antenna at each ST. Cell is divided into sectors, for each sector there is one base station (BSs), with a sectorized antenna, mounted on a transmission tower at a height of 40 m for enabling line-of-sight communication, as shown in Figure 3.1. WiFiRe system S consists of a set of base stations (BSs).

WiFiRe [2] has one common medium access (MAC) controller for all the sectors, to co-ordinate the medium access among them. There are multiple STs in each sector. The multiple access mechanism is time division duplexed, multi-sector TDM (TDD-MSTDM) scheduling of slots. Time is divided into frames. Each frame is further partitioned into a downlink (DL) and an uplink (UL) segments which need not be of equal durations. Within each segment there are multiple slots of equal duration each. Beacons are transmitted at the start of each DL segment. The beacon for each sector contains information regarding the DL and UL slot allocations (DL-MAP, UL-MAP) for that frame. All DL transmissions follow the DL-MAP in the Beacon. In each DL slot, one or zero transmissions can take place for each sector. The DL-MAP may allow multiple non-interfering BS to simultaneously transmit a packet to the ST(s) in their respective sectors. In each UL

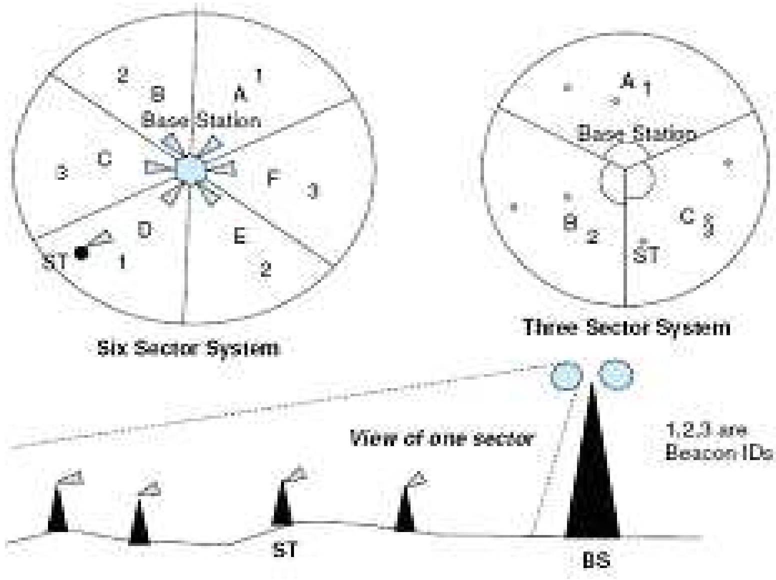


Figure 3.1: WiFiRe Network Configuration

slot, one or zero transmissions can take place for each sector, as governed by the UL-MAP. The UL-MAP is constructed in such a way that multiple ST(s) from different sectors, may transmit in the same UL slot, provided these transmissions are non-interfering at the BS.

In WiFiRe MAC layer there are three sub layers, Service Specific Sub-Layer (SSS), Link control sub layer and security sub layer.

The Service Specific Sub-layer (SSS) resides on top of the MAC Link Control Sub-layer (LCS). It provides the following services to upper layer. 1) Connection management and 2) Packet classification Link Control Sub-Layer (LCS) The MAC-LCS services are used by Service Specific Sub-layer (SSS) to access the connection-oriented wireless link for data packet transport. The LCS layer provides the following categories of services

- Connection provision services, including creation, termination and change.
- Data delivery services, from/to the higher layer SSS to/from the peer LCS entity

The association between a ST and the System S is static. This is determined by configuration at the ST during deployment. The association of a ST with a BS in a system S is dynamic and can change during each 'power-on' scenario of the ST. Ranging and registration are required for the synchronization between ST and BS and to ensure that a ST communicates with one and only one BS of system S. This association is fixed as long as the ST remains in 'power-on' mode.

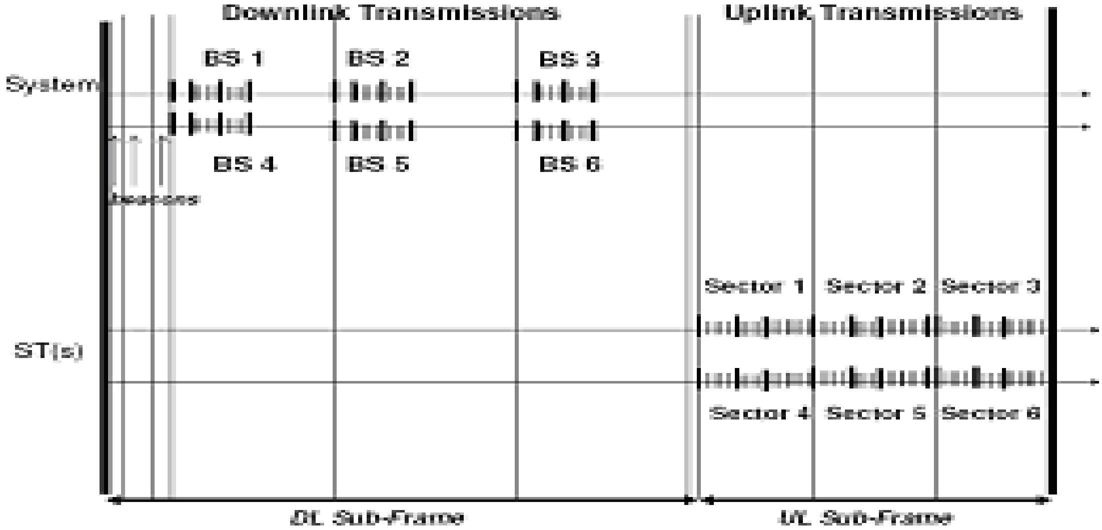


Figure 3.2: Parallel transmissions in a six sector system

3.2 Protocol Phases

The WiFiRe MAC protocol can be divided into two major phases [8]:

- Network Entry and Initialization (with two sub phases: Ranging and Registration)
- Connection Management and Data Transport.

3.2.1 Ranging

During power-on initialization, a ST attached to a BS of the system S, depending on the beacons it is able to hear from the system S. On powering up a ST listens for one or more beacons from the Operator and System ID it is configured for. There are some time slots reserved in the uplink segment for ranging. These are called ranging blocks and ranging request packets are transmitted in them. Ranging request has the following information: <System ID, ST ID, BS IDs that are audible to the ST, Signal strengths of beacons from the various BS>. Based on this, the system S associates the ST with one of the BS. Then S informs the ST about the timing synchronization and BS ID that will service the ST. This is done through a ranging response packet. Upon receipt of a ranging response from a BS, the ST is ready to receive from and transmit data to that BS.

3.2.2 Registration

The registration process is required prior for any data connection formation. The process involves a registration request from the ST, followed by a registration response from S. During this process, ST and S exchanges operational parameters and capabilities. In registration response S will send IP configuration information.

3.2.3 Connection Management

After registration, the ST can request for any number of connections further. In response to connection request S sends connection response to the ST, which is requested the connection. The MAC is connection-oriented and data flow between BS and ST occurs as per the service flow type associated with that particular connection. For example, a real-time VoIP data flow may be associated with one type of service flow while a best-effort TCP data flow may be associated with another type of service flow. The various types of service flows supported are described in section 3.3. An active service flow is identified uniquely by a connection identifier (CID). There are two ways to create and change service flow with intended QoS parameters:

- create the connection with the desired QoS, using a Dynamic Service Addition message
- create a generic connection (by specifying only the type of Bandwidth request service) and then use the CID to send Dynamic Service Change message to add specified QoS parameters to the connection.

Thus, Connection Management consists of Service Addition, Change and Deletion sub-procedures.

3.2.3.1 Service Addition

This is called connection creation process. In this BS or ST wishing to create a data connection exchanges management messages. If ST is wishing to create a data connection, it sends a Dynamic Service Addition Request (DSA-Req) control packet to BS. DSA-Req includes primary CID and requested QoS parameter set which can be used to represent QoS parameters for the requested connection.

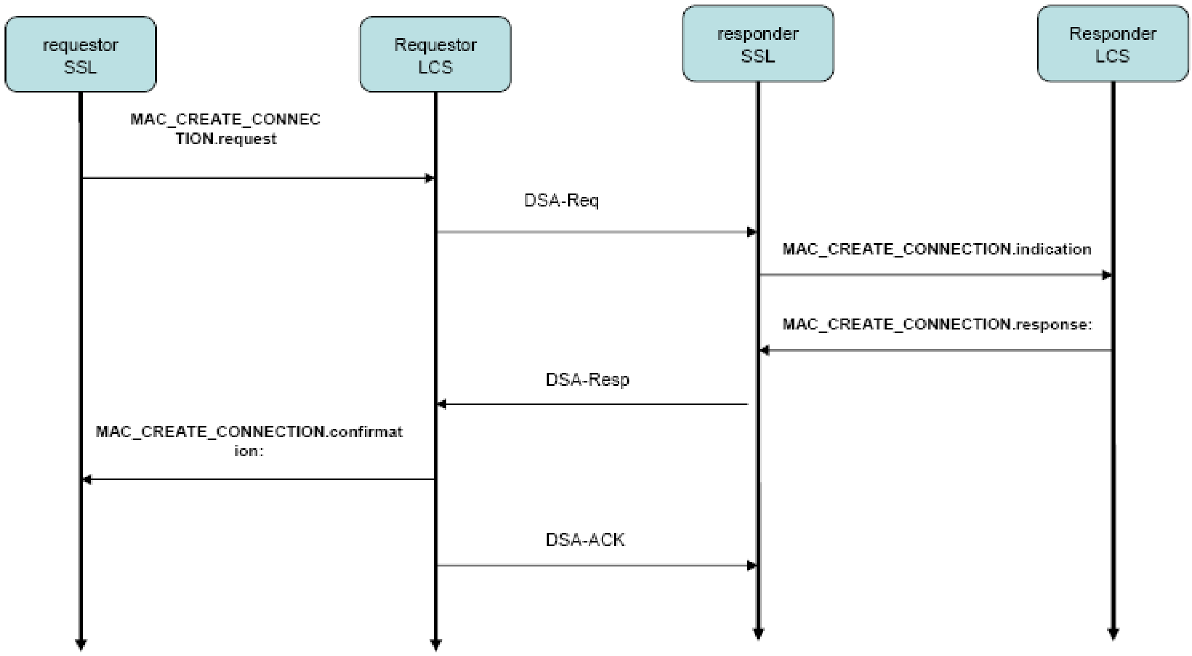


Figure 3.3: Connection Creation in WiFiRe

The BS processes the incoming DSA-Req, assigns the data CID if it is accepted and responds with DSA-Resp. DSA-Resp includes data CID and accepted QoS parameter set.

If BS is wishing to create data connection then it generates data CID and sends DSA-Req message to ST which includes data CID and QoS parameter set. ST responds with DSA-Resp with same CID and acceptance of QoS parameter set.

3.2.3.2 Service Change

This also called QoS management phase. It is used to change QoS parameters of existing connection or to set QoS parameters of new connection having CID but not having any allocated bandwidth resources. DSC messages are same except CID here is data CID for active connection every thing remaining is same.

3.2.3.3 Service Deletion

This is called connection termination phase. In this BS or ST wishing to terminate connection will exchange management messages to inform peer entity. If ST wishing to terminate a data connection sends a Dynamic Service Deletion Request (DSD-Req) message to BS. The DSD-Req contains a Data CID and QoS parameter set.

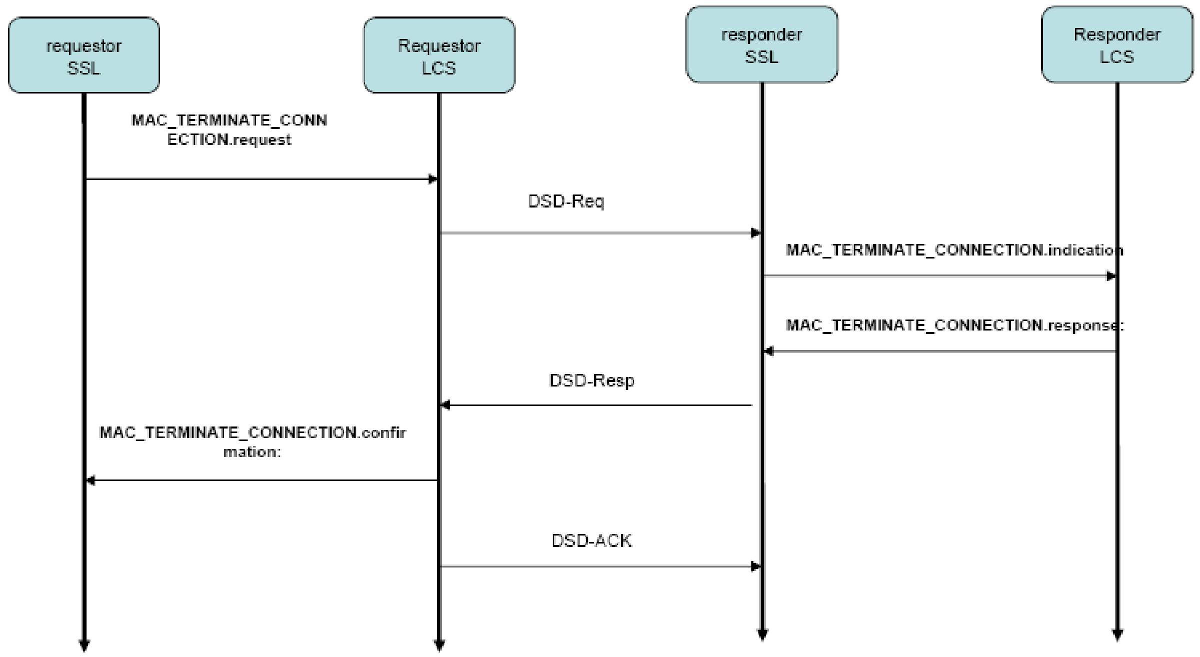


Figure 3.4: Connection Termination in WiFiRe

BS processes the DSD-Req, releases the resources assigned to that connection and sends DS-Resp. DSD-Resp contains data CID for connection being terminated and deletion status. if BS wishing to terminate a connection simply sends DSD-Req to ST. ST responds with DSD-Resp containing deletion status.

3.3 WiFiRe MAC Services

WiFiRe MAC is connection oriented like 802.16. A connection defines the mapping between peer data link processes that utilize both MAC and service flow category. The service flow category defines the QoS parameter for PDU(s) that are exchanged on the connection. Each connection has unique identifier to identify connections. A service flow is a MAC transport service that provides unidirectional transport of packets either to uplink packets transmitted by the SS or to downlink packets transmitted by the BS. A service flow is characterized by a set of QoS parameters such as latency, jitter etc. Each service flow is associated with some service classes and each service class is associated with one of four scheduling types. S may grant bandwidth to an ST in one or more of the following ways. :Unsolicited bandwidth grants, polling (real-time and non-real time) and contention based. WiFiRe MAC supports four types of service classes, Unsolicited Grant

Service (UGS), Real-time Polling Service (rtPS), non Real-time Polling Service (nrtPS) and Best Effort.

3.3.1 Unsolicited Grant Service

UGS is designed to support real-time flows that generate fixed size data packets on a periodic basis, such as T1/E1 and Voice over IP Without silence suppression. When a data CID is associated with UGS service flow type, the ST does not have to send periodic bandwidth request to the BS for that connection (data CID). The UGS service offers fixed size grants on a real-time periodic basis, which eliminate the overhead and latency of ST requests and assure that grants are available to meet the flow's real-time needs. The BS shall provide fixed size data grant slots at periodic intervals to the service flow.

3.3.2 Real-time Polling Service

The Real-Time Polling Service (rtPS) is designed to support real-time flows that generate variable size data packets on a periodic basis, such as MPEG video. The service offers real-time, periodic, unicast request opportunities, which meet the flow's real-time needs and allow the ST to specify the size of the desired grant. This service requires more request overhead than UGS, but supports variable grant sizes for optimum data transport efficiency. The BS shall provide periodic unicast request opportunities, by assigning appropriate polling slots in the uplink.

3.3.3 Non real time Polling Service

The Non-Real-Time Polling Service (nrtPS) is designed to support non real-time flows those require variable size data grant slots on a regular basis, such as high bandwidth FTP. The service offers unicast polls on a regular basis, which assures that the flow receives request opportunities even during network congestion. The BS typically polls nrtPS connections on an interval (periodic or non-periodic). The BS shall provide timely unicast request opportunities by assigning appropriate polling slots in the uplink frame.

3.3.4 Best Effort Service

The intent of the Best Effort (BE) service is to provide efficient service to best effort traffic . These flows served in contention slots.

3.4 MAC PDU Formats

Each PDU shall begin with fixed length Generic MAC header. Header may be followed by payload. Payload shall consists of zero of more sub headers and zero or more MAC SDU(s). Figure 3.5 shows the WiFiRe MAC PDU structure.

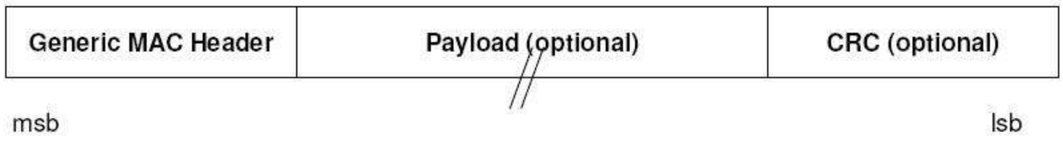


Figure 3.5: WiFiRe MAC PDU

There are two MAC header formats, Generic MAC header and Beacon MAC header. Generic MAC header is used for management messages and Data PDU(s). Beacon header is used to transmit beacons.

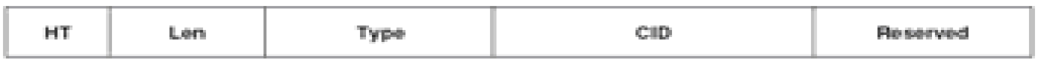


Figure 3.6: WiFiRe Generic MAC Header

Figure 3.6 shows the generic MAC header. In MAC header, HT field distinguishes the Generic and Beacon header formats and Len is 7 bits to indicate length of MAC PDU. Type is 1 byte to indicate MAC management message type, i.e DSA Request, DSA Response etc. CID is 2 bytes.

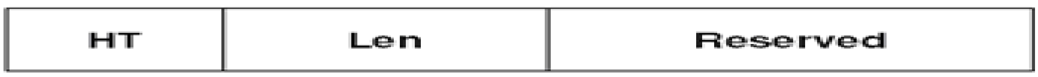


Figure 3.7: WiFiRe Beacon Header

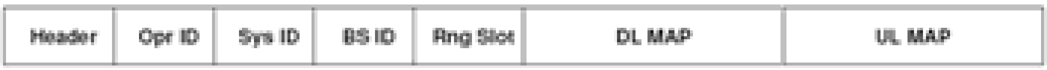


Figure 3.8: WiFiRe Beacon Message

Figure 3.8 shows the Beacon message. This Beacon message contains beacon header and Scheduling information. Header is 2 bytes; Rng Slots is 1 bit indicating if there are any ranging slots allocated in ULMAP. DL-MAP is 50 bytes. It is 50 element vector of $< CID1, CID2, .. >$. ULMAP is also 50 element vector of $< CID1, CID2, .. >$.

3.5 WiFiRe MAC Operation Details

Network Entry and Initialization: On ST Power On it determines operator ID and system ID from the configuration file and starts listening for beacons. The ST then constructs a Ranging Request and sends it in the ranging block of the UL sub frame for each BS that it hears. Ranging requests includes signal strength. After that ST waits for a Ranging Response. If no response is received within a timeout period, ST sends ranging request again after random back-off period. S receives the ranging request message and selects an appropriate BS, then S constructs the ranging response and puts it in the transmit queue of the corresponding BS and it invokes the scheduler to construct DL-MAP and UL-MAP. S transmits DL-MAP and UL-MAP in next beacon. Ranging response gets scheduled in next frame or some other subsequent frames. After successful Ranging, ST will register with its BS by sending Registration Request. Connection Management and Data Transport: When higher layer at ST has data to send, then it sends Dynamic Service Addition (DSA) message to BS in polling slots or contention slots. If no response is received within time out period it transmits again after waiting random back-off time. This DSA message is processed by admission control module in S. If resources are available then S assigns data CID to this connection and sends response through DSA-Response. QoS parameters associated with connection are known at both end. If it is UGS service flow then Scheduler at S will assign periodic grants in each frame. If it is rtPS or nrtPS then ST will request bandwidth whenever needed by sending Bandwidth Request in respected UL slot, then S will grant requested slots in next frame. To terminate connection ST sends Dynamic Service Deletion message to BS.

3.6 Modules simulated in WiFiRe OPNET model

All important parts of WiFiRe has been simulated in WifiRe model except PHY layer. Polling and Bandwidth Request mechanism are implemented in WiFiRe model to simulate rtPS and nrtPS service flows. Simple admission control has been implemented, this will admit flows on first come first serve basis. Round Robin Slot schedular is implemented to schedule accepted flows. Beacon processing and traffic scheduling has been implemented. This will works at both ST and BS. Packet segmentation and reassembly has been implemented to simulate slotting behavior.

Chapter 4

WiFiRe Model Design in OPNET

4.1 WiFiRe Model in OPNET

We used OPNET to develop a simulation model [9] for the WiFiRe protocol for evaluating performance. We are using WiMAX 03-Oct-2005 release Patch to build WiFiRe OPNET model. Main components in WiFiRe OPNET model are Base Station node model, ST node model, common MAC process model, BS child control process model and ST Child control process model. Base Station node consists of 4 Ethernet interfaces and 6 WiFiRe interfaces. Each WiFiRe interface represents a sector shown in figure 4.3.WiFiRe ST node model is fixed node model with workstation functionality shown in figure 4.4.

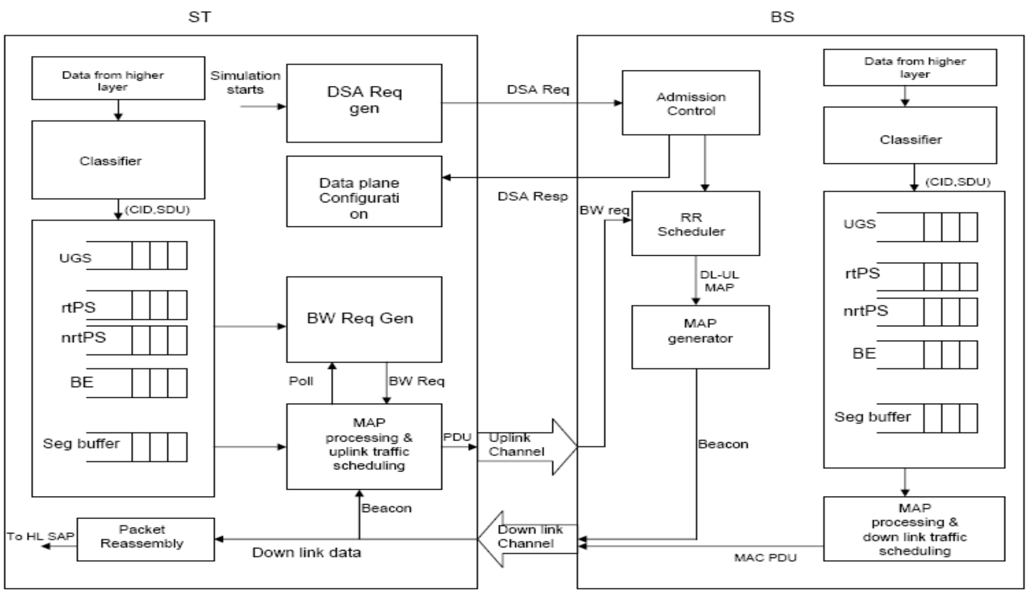


Figure 4.1: WiFiRe Model design block diagram

Figure 4.1 shows the Design block diagram of WiFiRe model in OPNET. Some modules shown in figure are common to both BS node and ST node, for example, Classifier is needed at both ends. Beacon processing, traffic scheduling and lower layer data processing modules are also common for both ST and BS. Admission control module runs only on Base Station node. Admission control, slot scheduler and MAP generator are BS specific tasks. DSA Req, BW Req generator tasks are specific to ST. The figure 4.1 shows how all these modules work together to build the WiFiRe simulator.

When the simulation starts, STs will send the DSA Req to BS, for each service flow (uplink or downlink) defined on it. Both uplink and downlink service flows are defined at each ST node. DSA Req contains the service flow parameters for requesting connection and source MAC address. When BS receives DSA Req message, it calls the Admission control module to process it. Admission control module will process it and send DSA Resp to requested ST. DSA Resp contains acceptance status and accepted service flow parameters. Admission control process is different for different types of services. Section 4.4 and algorithm 2 discuss the admission control process for different types of flows. If ST receives DSA Resp with an accepted state of connection, then it will configure the data plane data structures, containing Shaper, Classifier, data queue and segmentation buffer for the accepted connection.

Periodically, The scheduler processes the list of all accepted flows and allocates bandwidth for the active flows. It allocates polling slots to rtPS and nrtPS flows if the last polled time exceeds the polling interval. Section 4.5 and algorithm 3 discuss the MAC generation process using Round Robin scheduler.

When MAC receives a packet from higher layer, it gives this packet to the classifier module. The classifier classifies this packet into the connection it belongs to and gives connection ID (CID). Section 4.3 discuss how classifier classifies higher layer packets into connection IDs. After classification, the higher layer SDU is enqueued into the queue corresponding to that CID. The BS sends beacons which contain slot scheduling information for downlink and uplink. When an ST receives a beacon, it extracts the Uplink MAP and schedules its traffic in corresponding slots. Algorithm 1 discusses the beacon processing and traffic scheduling in both uplink and downlink. The ST sends a BW request with request size equal to queue size, if it is polled.

For each connection, there is one segmentation buffer shown in figure 4.1. While scheduling packets from packet buffer, if the next packet size in the queue is greater than allocated size of that connection, then scheduler calls the segmentation module which segments the packet and inserts resultant segments into corresponding segmentation buffer for that connection. After segmentation is completed, scheduler takes one segment from segmentation buffer and schedules it for transmission. Scheduler checks for pending segments of previous packet in segmentation buffer before scheduling new packets from packet buffer. If segmentation buffer is empty, then it takes one packet from packet buffer and segmentation is applied if packet size is more than allocated size. The scheduler schedules packets in FCFS order. At the receiving end, reassembly module reconstructs the segmented packets and sends them to the upper layer SAP.

4.2 Process models

4.2.1 WiFiRe Common MAC Process model

The model shown in Figure 4.2 is the main process model. This process runs at both BS and ST nodes. Packet classification, constructing MAC frames, segmentation and reassembly, and beacon processing etc. are functionalities of this process. It is called whenever packets or control come to the WiFiRe MAC layer. It calls child processes depending upon the packet type.

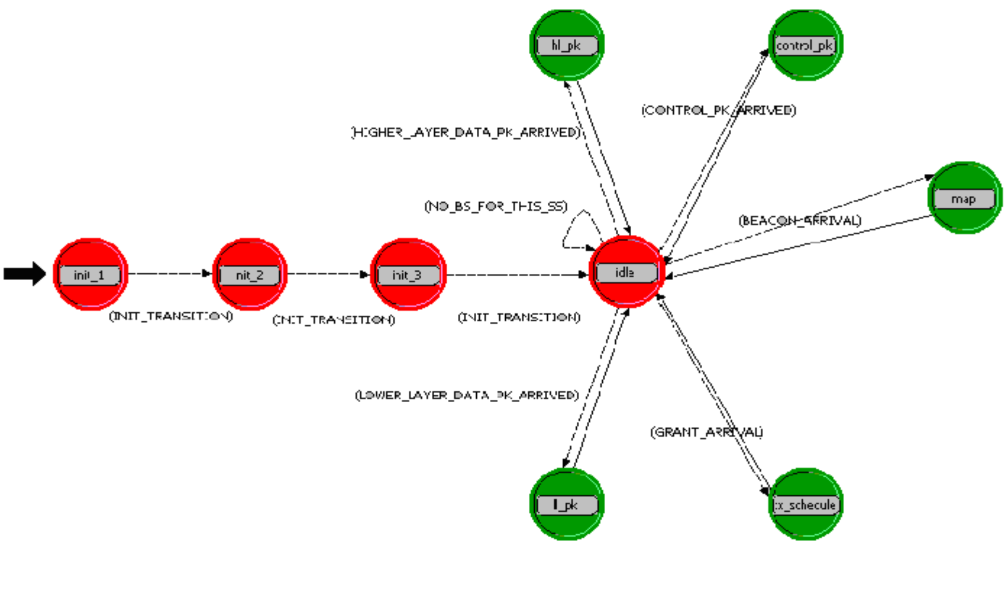


Figure 4.2: Common MAC process model

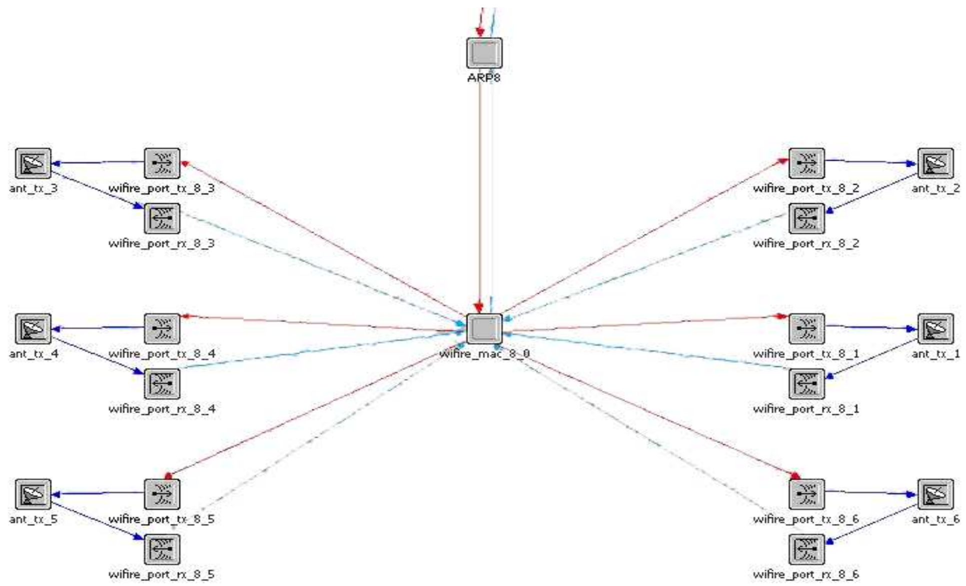


Figure 4.3: WiFiRe BS node model

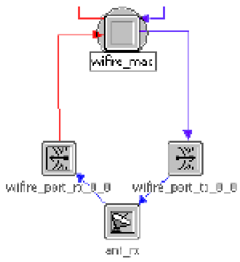


Figure 4.4: WiFiRe SS node model

Init1 state:

In this state, MAC parameters will be parsed, and data plane structures: multiplexer, demultiplexer, and classifier set are created and initialized. Classifiers defined at both ST and BS are parsed and inserted into the data plane.

Idle state:

This state will check whether the incoming packet is higher layer data packet, a lower layer data packet or a control packet. If the packet is coming from higher layer then it goes to `'hl_pk'`state. If it is coming from lower layer then it goes to `'ll_pk'`state if it is data packet and it goes to `control_pk` state if it is control packet.

hl_pk state:

The incoming packet is first classified, and associated with a CID. After classification, it gets queued into corresponding queue for that CID. Section 4.3, discusses the classification process in detail.

ll_pk_state:

It extracts the CID from the incoming MAC PDU and checks whether the packet is destined to this ST or not by checking its CID. If it is destined to this ST, then it simply extracts the MAC SDU from the incoming packet and checks whether SDU is segmented or is a complete packet. If it is segmented SDU, then it is inserted into packet reassembly buffer. If any packet reassembling completes then it is sent to the upper layer. It also updates throughput and traffic received statistics.

Beacon process state:

All STs process beacon independently and schedule their traffic according to slot numbers in MAP. Downlink transmission from BS is followed by uplink transmission by STs . Sending packet is implemented using direct delivery method. So communication between BS and ST is MAC to MAC communication. Algorithm 1 discusses the procedure of beacon processing and traffic scheduling in detail.

***control_pk_state*:**

Root MAC process spawns a control child process to process the control messages. If the MAC role is BS and incoming control packet is DSA request or bandwidth request then BS control child process is invoked. The ST control child process is invoked if MAC role is ST and incoming control packet is a DSA response.

4.2.2 ST control child process model

This is invoked when parent process receives a control packet or wants to send a control packet i.e. DSA Request, DSA Response. Initially when simulation starts, this process is invoked by parent process to deliver DSA request for each service flow defined on this ST.

Algorithm 1 Beacon processing

```
// get DL-MAP and UL-MAP
if mac_role=BS then
    // Downlink Transmission
    for i = 0 to 49 do
        cid = DL_MAP[i];
        no_slots=get number of allocated slots;
        if is_my_data_cid(cid)=true then
            // get packet with less than or equal to no_slots*slot_size
            pkt=get_packet(cid,no_slots);
            tx_time=pk_len/data_rate;
            delay=frame_duration*dl_ul_ratio+starting_slot*slot_duration+tx_time;
            schedule_packet(pkt,delay);
        end if
    end for
else if mac_role=ST then
    // Uplink Transmissions from STs
    for i = 0 to 49 do
        cid = UL_MAP[i];
        no_slots=get number of allocated slots;
        if is_my_data_cid(cid)=true then
            // get packet with less than or equal to no_slots*slot_size
            pkt=get_packet(cid,no_slots);
            tx_time=pk_len/data_rate;
            delay=frame_duration*dl_ul_ratio+starting_slot*slot_duration+tx_time;
            schedule_packet(pkt,delay);
        else if is_my_primary_cid(cid)=true then
            send bandwidth request
            bw_pkt=get_bw_req_packet(cid);
            tx_time=bw_pkt_len/data_rate;
            delay=frame_duration*dl_ul_ratio+starting_slot*slot_duration+tx_time;
            schedule_packet(bw_pkt,delay);
        end if
    end for
end if
```

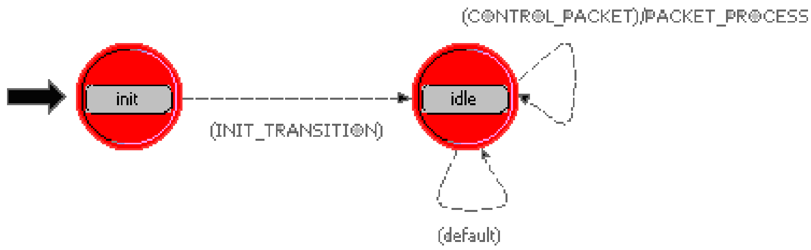


Figure 4.5: ST control child process model

4.2.3 BS control child process model

Functionalities of this process are admission control, admitted flows scheduling, MAP generation, bandwidth grant scheduler and activation of admitted service flows in downlink. This process is invoked when parent process receives a control packet or when the parent process wants to send a bandwidth request. It also processes DSA request messages coming from STs. These DSA request messages are processed by admission control module. MAP generation function will be called periodically to generate DL-MAP and UL-MAP. Sections 4.4 and 4.5, discuss simple FCFS based admission control algorithm, and Round Robin slot scheduling algorithm in detail.

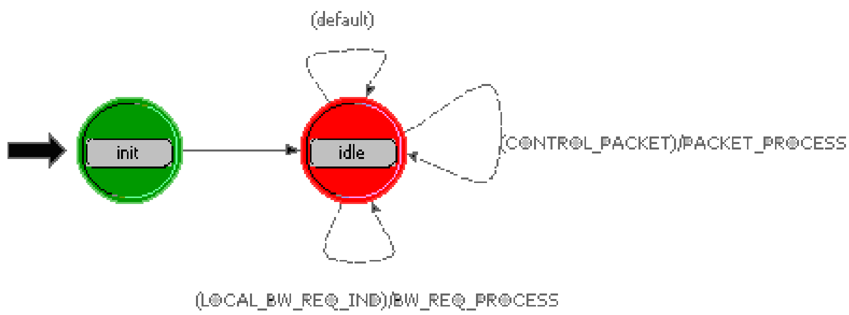


Figure 4.6: BS control child process model

4.3 Classifier

Classification is the process by which a MAC SDU is mapped to a particular connection for transmission between MAC peers. Basic classification module block diagram is shown in figure 4.7. A classifier has a set of matching criteria applied to each packet entering into WiFiRe network. It consists of some protocol specific packet matching criteria (destination IP address, source IP address etc.), and reference to a CID. If packet matches specified packet matching conditions, it is given the service class name of that packet. It concatenates service class name with destination MAC address to form a key, which can be used to extract the CID for this packet. The key is unique at both ST and BS. This restricts the maximum number of uplink and downlink service flows at ST to maximum number of service classes. So we are not allowed to define more than one uplink/downlink service flow for the same service class in any ST.

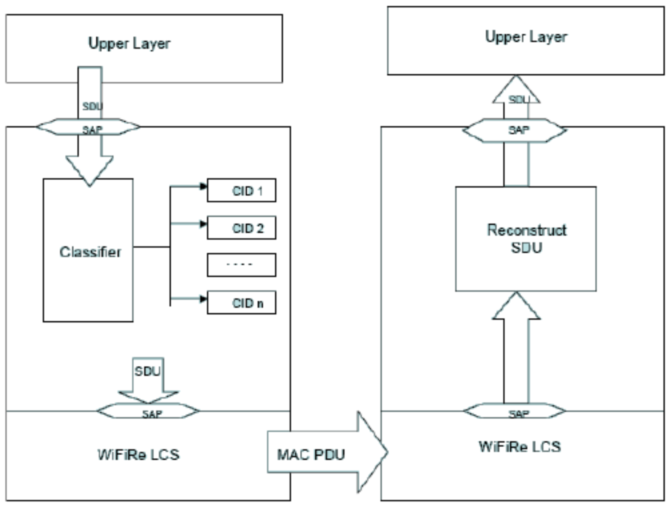


Figure 4.7: Classification Module Block diagram

After completion of mapping, packet will be enqueued into corresponding CID queue for delivery on the connection defined by CID. Each CID is associated with a set of service flow characteristics. These service flow characteristics of connection provide the QoS for packet. Downlink classifiers are applied by BS to packets it is transmitting in downlink and uplink classifiers are applied at the ST.

Figure 4.8 shows how classifier can be defined for a particular type of traffic. The classifier shown in that figure maps voice packets to Gold service class, then it is mapped to corresponding CID by using the destination address. Gold service class associated with UGS scheduling type and some other service flow parameters are shown in figure 4.9.

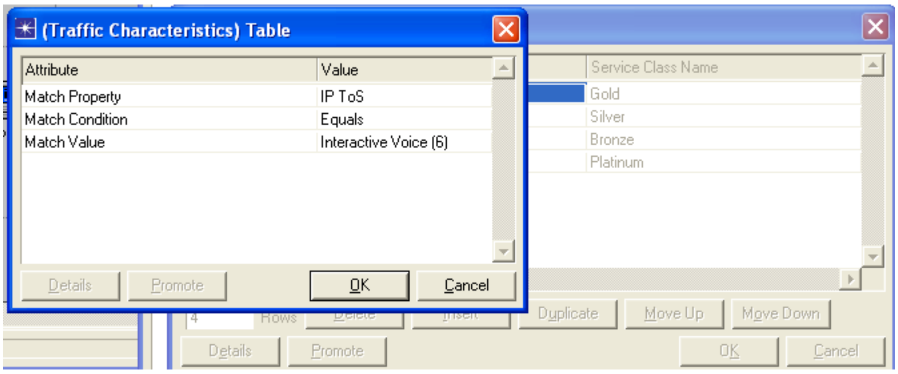


Figure 4.8: Classifier Setup

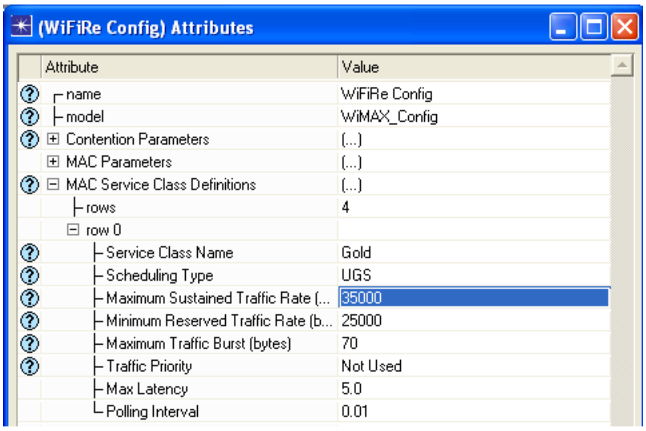


Figure 4.9: Service flow parameters setup

4.4 Admission Control in WiFiRe

DSA Request from STs triggers the Admission control function at the Base Station. Admission control module extracts the service flow parameters from DSA request packet and processes the flow parameters. If sufficient resources are available then BS admits the flow and generates CID for that service flow. After that it activates that service flow and sends DSA response which includes the acceptance service flow parameters and CID. If scheduling type of the service flow is rtPS or nrtPS then it also generates primary CID and includes it in DSA response. Primary CID is used by STs for requesting bandwidth.

If service flow is rejected due to insufficient resources then BS constructs DSA response which includes admission decision and sends it to the requested ST.

Admission control decision is done differently depending on the scheduling type of

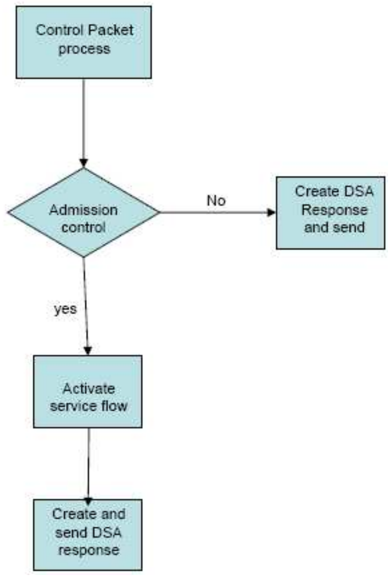


Figure 4.10: Admission Control flow chart

service flow. If scheduling type is UGS then BS will admit it with maximum sustained traffic rate. If scheduling type is rtPS or nrtPS BS will admit it with minimum reserved traffic rate. Best effort flows are accepted by BS.

The requested rate from STs is free of MAC and PHY overhead. So BS adds the MAC and PHY overhead to the requested rate. Now if the available bandwidth is greater than requested rate, then it admits the flow. BS admits UGS flows with maximum sustained traffic rate, rtPS and nrtPS flows with minimum reserved traffic rate. For uplink rtPS and nrtPS flows there is additional polling overhead. After adding overhead, if the remaining bandwidth is greater than or equal to requested bandwidth then BS will admit the flow otherwise it is simply reject them. Algorithm 2 shows the admission control algorithm.

4.5 Round Robin Slot Scheduling

Round Robin scheduler is a simple slot scheduling algorithm. The frame is divided into downlink and uplink sub-frames. Downlink and Uplink sub-frames are further dived into three contiguous parts if opposite sectors transmit in parallel. For example, in six sector system, sectors 1 and 4 serve in first part, sectors 2 and 5 serve in second part and sectors 3 and 6 serve in the third part. If alternate base stations transmit in parallel, then Downlink and Uplink further divide them into two contiguous parts only. Sectors 1,3,5 are served

Algorithm 2 Admission Control

```
// check available rate with requested rate
scheduling_type = service_flow → parameters → scheduling_type ;
// If it is UGS flow.
if scheduling_type = UGS then
    rate = service_flow → parameters → max_sustained_traffic_rate;
    rate = rate*(average_sdu_size + mac_header_size)/average_sdu_size);
end if
// If it is rtPS or nrtPS flow.
if scheduling_type = rtPS or scheduling_type = rtPS then
    rate = service_flow → parameters → max_sustained_traffic_rate;
    rate = rate*(average_sdu_size + mac_header_size)/average_sdu_size);
    if is_uplinkflow=true then
        // add polling overhead. Assume one bandwidth request for each packet
        rate_pps=service_flow → parameters → max_sustained_traffic_rate/(8.0*service_flow -
        avg_sdu_size_bytes));
        rate += rate_pps*MAC_HEADER_SIZE_BYTES*8;
    end if
end if
// If it is Best Effort flow.
if scheduling_type = BE then
    is_admitted=true;
end if
rate=PHY_overhead
if rate ≤ available_bandwidth then
    // admit this flow
    is_admitted=true
    available_bandwidth-=available_bandwidth
else
    // reject this flow
    is_admitted=false
end if
```

in first part and sectors 2,4,6 served in second part. Maximum number of simultaneous transmissions, $n_0 = 3$ are allowed in each slot in six sector system. But with greedy slot scheduling algorithm we can have a maximum of 4 simultaneous transmissions in one slot.

Round Robin scheduler first allocates polling slots to rtPS and nrtPS flows if last polled time exceeds polling interval. Then it scans the list of all admitted flows and allocates admitted number of slots for UGS flows. It allocates requested number of slots for rtPS and nrtPS flows, if the requested number of slots are less than the admitted number of slots. Otherwise, it allocates admitted number of slots, it will allocate remaining slots in next frames. Scheduler will allocate remaining slots to Best Effort flows. If there are any slots free after allocating BE flows, scheduler will mark all those slots as contention slots.

Algorithm 3 is same for scheduling both uplink and downlink flows, except in downlink there is no need to allocate polling slots for rtPS and nrtPS.

4.6 Model Features and Limitations

The *WiFiRe* OPNET model is a discrete event simulation model that lets you analyze performance of *WiFiRe* MAC layer protocol. The *WiFiRe* model includes all potential features of the protocol. The model features and limitations are given below.

- Model Features:
  1. MAC messages: DSA Request, DSA Response, BW Request, Beacon and WiFiRe MAC PDU.
  2. Polling and Bandwidth allocation.
  3. Round Robin Scheduler and Connection Admission Control.
  4. Traffic scheduling in both uplink and downlink.
  5. IP Classifier.
- Model Limitations:
  1. Physical layer and ranging functionalities are not modeled.
  2. Dynamic connection creation and release.
  3. Contention based Best Effort service is not modeled.

Algorithm 3 Round Robin Slot Scheduling

```
// 1. Allocate Bandwidth request opportunities(polling slots)
while admission_control_elem ≠ NULL do
  service_flow=admission_control_elem → service_flow
  if service_flow → direction = UL then
    if service_flow → type=rtPS and elapsed_time ≥service_flow → polling_interval then
      // poll This connection
      ul_map[i++]=service_flow → primary_cid;
    else if service_flow → type=nrtPS and elapsed_time ≥service_flow → polling_interval*2 then
      ul_map[i++]=service_flow → primary_cid;
    end if
  end ifadmission_control_elem=admission_control_elem → next;
end while
2.1 Allocate slots to admitted service flows
while admission_control_elem ≠ NULL do
  service_flow=admission_control_elem → service_flow
  if service_flow → type=UGS then
    // always Allocate admitted number of slots to UGS flows.
    ul_map[i++]=service_flow → cid;
  else if service_flow → type=rtPS or service_flow → type=nrtPS then
    // get the bandwidth request from corresponding bandwidth queue.
    if bandwidth_req leqadmitted_bandwidth then
      // allocate requested number of slots
      ul_map[i++]=service_flow → cid;
    else
      // allocate Admitted number of slots
      ul_map(i++]=service_flow → cid;
    end if
  end if
  admission_control_elem=admission_control_elem → next;
end while
// 2.2 Allocate remaining slots to Best Effort service flows
// 2.3 mark remaining slots as contention slots
```

Chapter 5

Simulation Experiments and Results

In this section, we present the performance evaluation results obtained from several different scenarios.

Goal

- Building WiFiRe model in OPNET.
- Finding minimum slot length to support VoIP and Video services.
- Finding maximum number of users system can support.
- Analyzing queuing delay, end to end MAC delay and throughput for different types of flows with different slot lengths.

5.1 Experiments

The Table 5.1 shows common *WiFiRe* setup parameters for all simulations we have conducted. We are assuming modulation technique is BPSK and coding rate is $\frac{1}{2}$ according to 802.11b. With 11 Mbps, BPSK modulation and with $\frac{1}{2}$ coding rate, number of symbols per 10 ms frame is equal to 2,20,000.

5.1.1 VoIP user Scenario

5.1.1.1 Purpose

The aim of this scenario is to find the minimum slot size required to support voice applications, and the maximum number of users the system can support with this slot size. We can also compare end to end MAC delay with different slot sizes.

S.No	Parameter	value
1	Frequency Channel	2.4 GHz
2	Bandwidth	22 MHz
3	Data Rate	11 Mbps
4	Frame Duration	10 ms
5	Symbols per frame	220000
6	Symbol duration	.045μs
7	Scheduling Algorithm	Round Robin

Table 5.1: Common parameters for all experiments

S.No	Parameter	value
1	Slot Duration	32μs
2	Contentions Slots	10
3	DL-UL ratio	2:1
4	Voice codec	G729
5	Number of STs	1

Table 5.2: UGS Setup parameters

S.No	Parameter	value
1	Service Class Name	UGS
2	Maximum traffic rate	24 Kbps
3	Minimum traffic rate	24 Kbps
4	Max latency	4 seconds
5	Polling Interval	NA

Table 5.3: UGS service flow parameters

5.1.1.2 Setup parameters

ST is configured with one VoIP call having G729 codec and one downlink service flow and one uplink service flow with parameters shown in Table 5.3. Polling interval for UGS flows is not required. System is configured with 32 μs slot and 10 ms frame. Other setup parameters are shown in Table 5.2. Simulation setup is shown in Figure 5.1.

5.1.1.3 Results

We assumed BPSK modulation technique and $\frac{1}{2}$ coding rate as per 802.11b.
Channel capacity in symbols per second is data_rate $\times \frac{1}{bits\_per\_symbol \times coding\_rate}$
PHY capacity in sps $= 11 \times 10^6 \times \frac{1}{1 \times \frac{1}{2}} = 22 \times 10^6$.
Number of symbols per 10 ms frame= $(22 \times 10^6) \times (10 \times 10^{-3}) = 220000$. This will be doubled because frequency reused in opposite sectors. So total number of symbols per 10

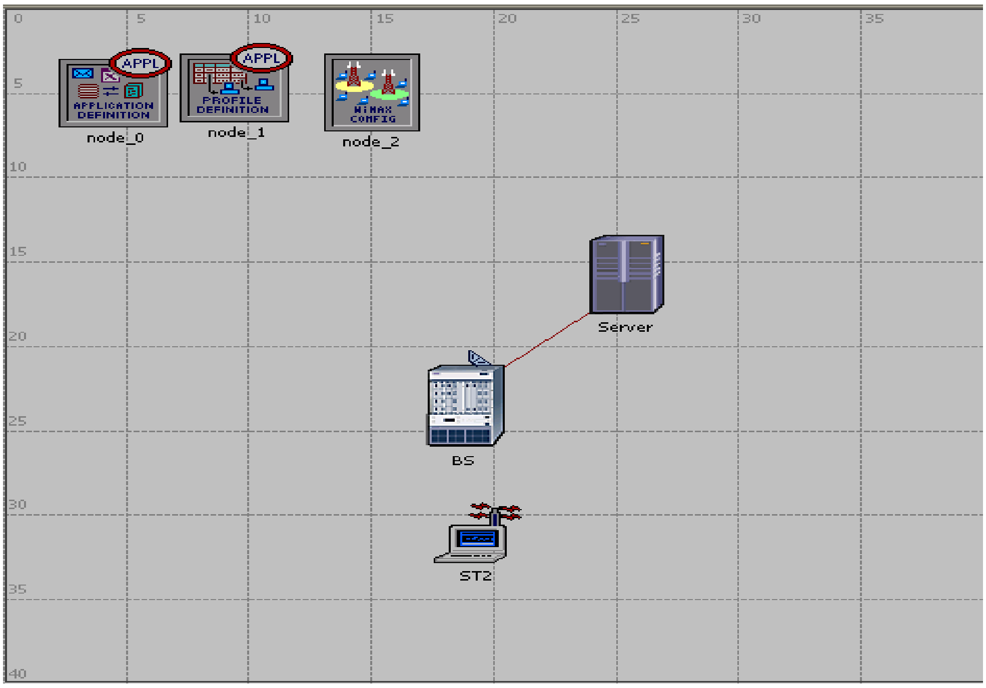


Figure 5.1: Simulation setup with one ST, and one BS

ms frame = 440000. We will calculate, how many VoIP connections BS will admit with frame capacity 440000 symbols.

Theoretical Maximum number of VoIP users:

Original bandwidth request from ST is 24 Kbps.

Requested rate with MAC overhead is $rate \times (\frac{\text{avg packet size+WiFiRe header}}{\text{avg packet size}})$

$24Kbps \times \frac{60+5}{60} = 26$ Kbps

Requested rate in symbols per second (sps) is $rate\_bps \times \frac{1}{\text{number of bits per symbol} \times \text{coding rate}}$

26 Kbps$\times \frac{1}{1 \times \frac{1}{2}} = 52 \times 10^3$ sps

Then convert symbols per second into symbols per frame

Number of symbols required per frame is (rate in sps) $\times$ (frame duration)

$(52 \times 10^3) \times (10 \times 10^{-3}) = 520$

Number of symbols per slot $\frac{\text{slot duration}}{\text{symbol duration}}$

$= \frac{32}{.045} = 711$ symbols

Number of symbols needed for WiFiRe PHY overhead, that is 96 μs is

$\frac{\text{PHY overhead duration}}{symbolduration}$

$= \frac{96}{.045} =$2133 symbols

Total number of symbols required is the sum of phy overhead and requested rate

$$\text{phy overhead symbols} + \text{number of symbols requested}$$
$$= 2133+520 = 2653$$

Convert requested symbols per frame into slots per frame.

Then number of slots required is $= \frac{\text{number of symbols requested}}{\text{number of symbols per slot}}$

$$= \frac{2653}{711} = 4$$

Therefore we need 4 slots, one is data slot and 3 PHY slots.

Maximum number of connections per frame =

$$\frac{\text{total symbols per frame}}{\text{number of symbols requested per frame}} \times \text{number of parallel tx}$$
$$\frac{220000}{2653} \times 2 = 165$$

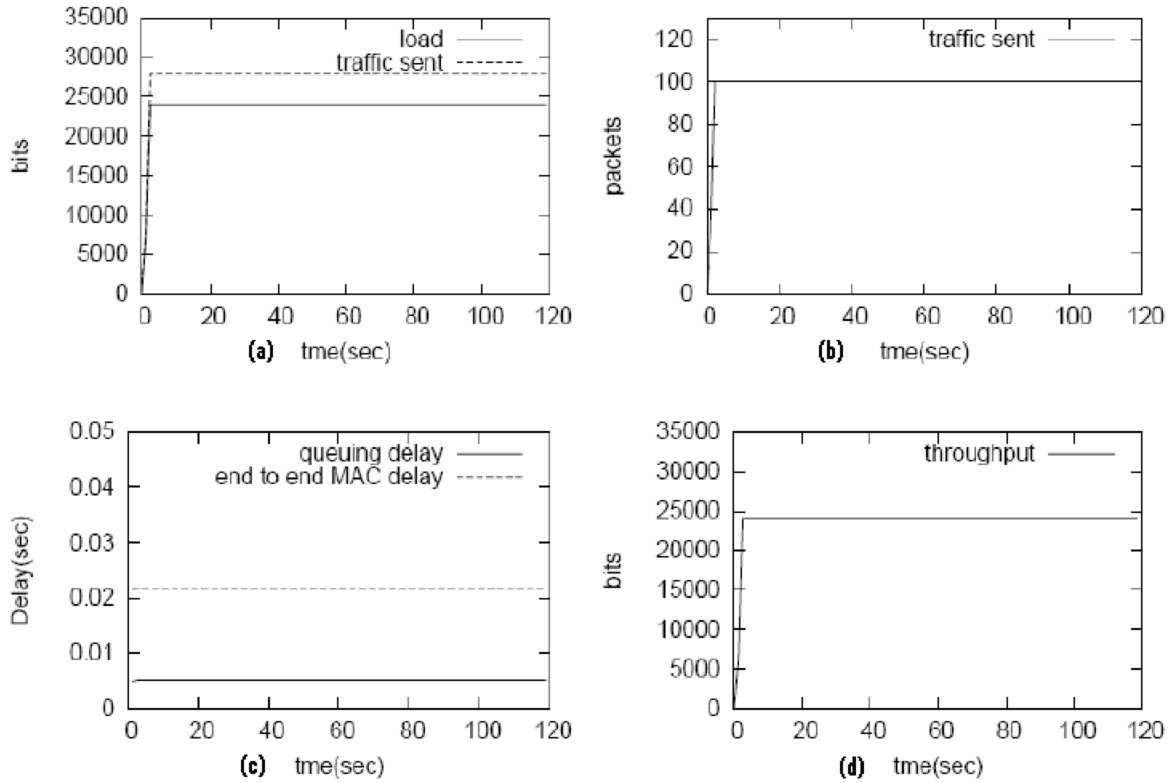
Because frequency is reused in opposite sectors.

Above calculations show that one Base Station can support upto 165 VoIP connections (both in uplink and downlink) with the configuration shown in Tables 5.2 and 5.3.

Figure 5.2 shows the simulation results with configuration shown in Tables 5.2 and 5.3. In this experiment we used G729 codec with 20 ms sampling rate and 8 kbps coding rate. Figure 5.2(a) shows the load is 24 kbps and traffic sent is 28 kbps. Using G729 codec with 20 ms sampling rate and 8 kbps coding rate, voice payload will be 20 bytes. So total packet size at MAC layer is 20+40(IP+UDP+RTPS) bytes. This VoIP packet will be served in two 10 ms frames, because with one 32μs slot, ST can send only 44 bytes. So 60 bytes VoIP packet is segmented into two and sends in two frames. Queuing delay and end to end MAC delay is constant shown in Figure 5.2(c), because arrival rate is equal to service rate that is one VoIP packet per 20 ms. Figure 5.2(a) shows traffic sent as 28 kbps because WiFiRe overhead is 5 bytes, this will give 4 kbps of header overhead, so total traffic sent is 24 kbps + 4 kbps is equal to 28 kbps.

With this experiment it has been observed that 32 μs slot is enough to provide VoIP services with G729 codec, 20 ms sampling rate and 8 kbps coding rate.

Figures 5.3 and 5.4 shows the simulation results with configuration shown in Tables 5.2 and 5.3. In this experiment we used G729 codec with 10ms sampling rate and 8 kbps coding rate. Figure 5.3(a) shows that load at ST MAC layer is 40 kbps and it is sending traffic at around 24 kbps. Voice packet size at MAC layer is 50 bytes(without *WiFiRe* header) using G729 codec with 10 ms sampling rate and 8 kbps coding rate. We have

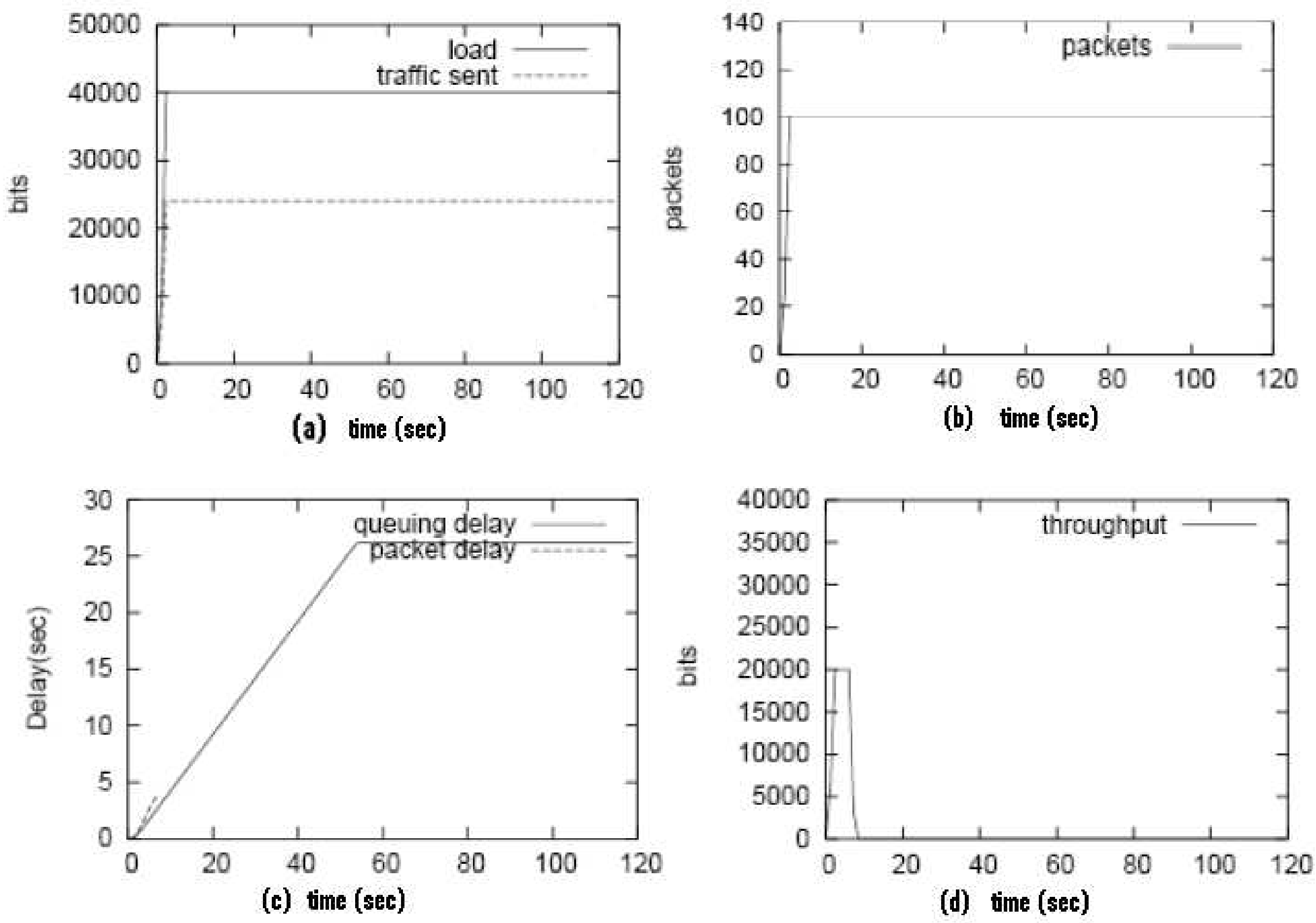


(a) load and traffic sent vs time. (b) packets sent vs time.(c) queuing delay and end to end MAC delay vs time. (d) throughput vs time.

Figure 5.2: UGS statistics with slot size 32 μs and VoIP application with G 729 codec, 20 ms sampling rate, 8 kbps coding rate.

44 bytes slot size, so this 50 byte voice packet is segmented into two parts with average segment size 25 bytes. Now ST will send one segment in one frame and another segment in another frame. On an average ST is sending 30 (SDU + WiFiRe header) bytes per 10 ms, which means that it is sending traffic at the rate of 24 kbps. This will increase queuing delay and packet end to end delay. After some time queue will be filled and all packets will be dropped after that. End to end MAC delay is increased upto 4 seconds shown in Figure 5.3(c) and it reached, maximum latency that is 4 seconds. Destination MAC will drop all those packets, for which MAC end to end delay is more than maximum latency, hence throughput dropped to zero shown in Figure 5.3(d). Base Station is serving one MAC PDU in each frame means 100 MAC PDUs per second with 10 ms frame, shown in above Figure 5.3(b).

Figure 5.4 VoIP application statistics at application layer. This Figure shows that



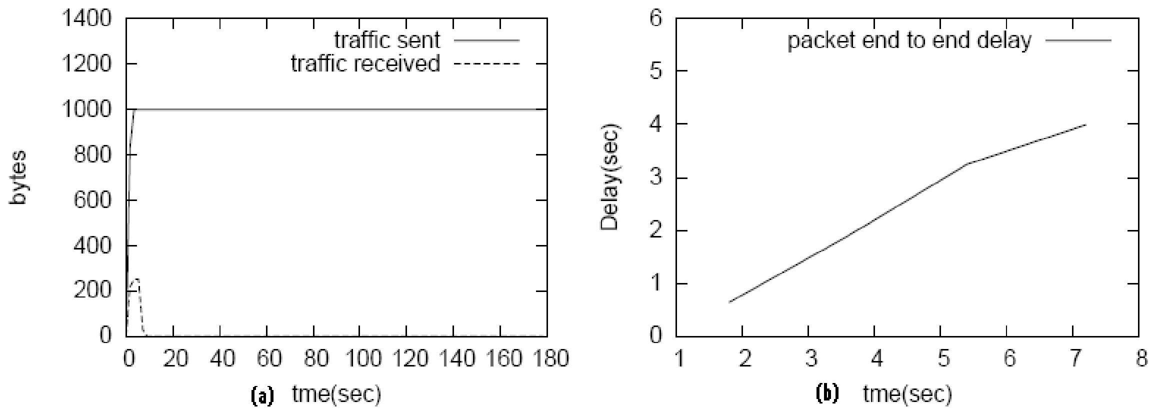
(a) load,traffic sent vs time. (b) Throughput vs time (c) packets sent vs time. (d) queuing delay and end to end MAC delay vs time.

Figure 5.3: UGS statistics with slot length 32μs and VoIP application with G729 codec, sampling rate 10 ms and 8 kbps coding rate.

packet end to end delay is increasing, because serving rate at MAC layer is less than arrival rate.

From these experiments it has been observed that 32 μs slot is not suitable for Voice applications using G729 codec with 10 ms sampling rate and 8 kbps coding rate. We can solve this problem by requesting 40 Kbps to the Base Station, then BS will allocate two data slots to it. But the total number of admitted connections will be decreased. Next experiment will discuss the simulation results with a slot size of 40 μs.

The Figure 5.5 shows the simulation results with configuration shown in Tables 5.2 and 5.3 except slot size and requested bandwidth. This experiment is configured with slot



(a) traffic sent and traffic received vs time. (b) packet end to end delay

Figure 5.4: VoIP statistics at application layer with G729 codec with 10 ms sampling rate.

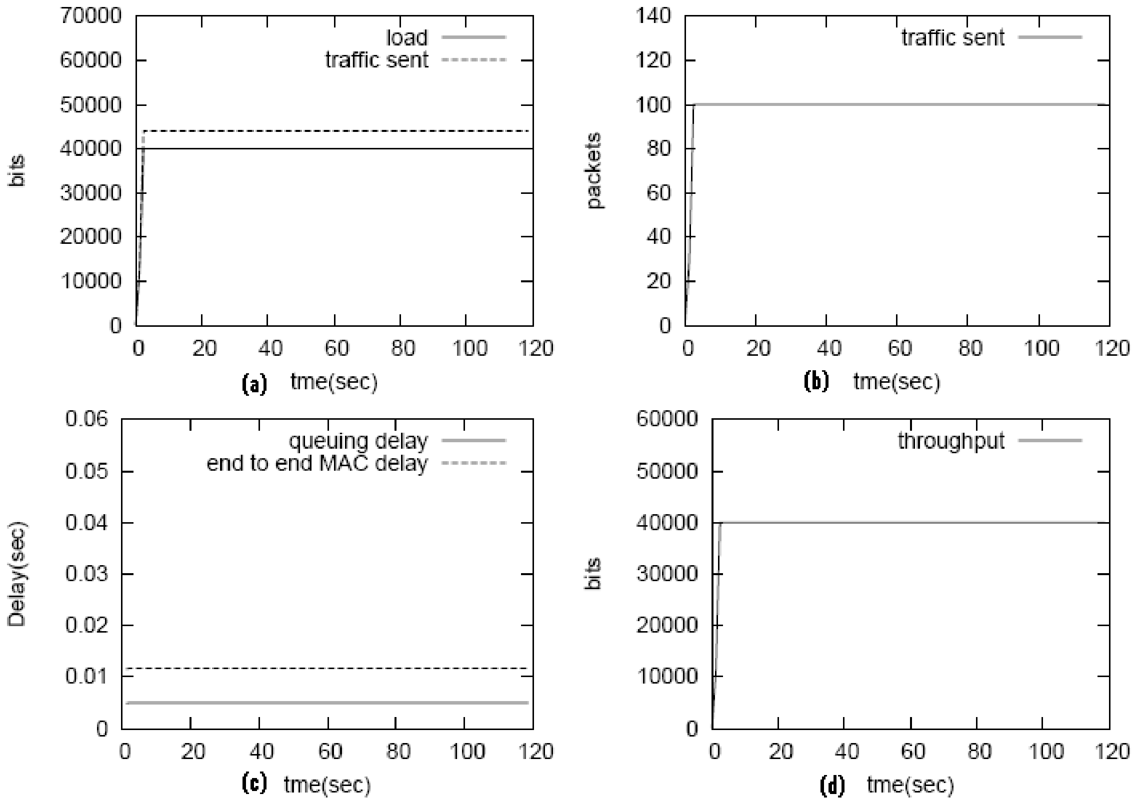
size 40 μs and 40 kbps requested bandwidth. In setup we used G729 codec with 10ms sampling rate and 8 kbps coding rate. Figure 5.5(a) shows that load at MAC layer is 40 kbps and traffic sent is around 44 kbps, this includes *WiFiRe* overhead. So queuing delay and end to end MAC delay are constant shown in Figure 5.5(c). With one 40μs slot, ST can sent data at 44 kbps. In this setup, throughput is 40 kbps is equal to load, because no packets are dropped.

From this experiment, it has has been observed that 40 μs slot size is suitable for VoIP service with G729, 10 ms sampling rate and 8 kbps coding rate. Maximum number of admitted connections by BS is 138, in this scenario. It shows that the number of admitted connections has decreased compared to previous experiment with slot 32 μs.

5.1.2 Video Conference user Scenario

5.1.2.1 Purpose

The aim of this experiment is to find the minimum slot size required to support video applications. The experimental results will also show the effect of slot size on queuing delay and end to end MAC delay. Results will also show the effect of polling interval on queuing delay and end to end MAC delay.



(a) load and traffic sent vs time. (b) packets sent vs time. (c) queuing delay, end to end MAC delay vs time. (d) throughput vs time

Figure 5.5: UGS statistics with slot size 40 μs and VoIP application with G729 codec, 10 ms sampling rate and 8 kbps coding rate.

5.1.2.2 Setup parameters

This experiment is configured with parameters shown in Tables 5.4 and 5.5. The common setup parameters are shown in Table 5.1. ST is configured with one Video conference application. This application is configured with .2 sec frame interval time exponentially distributed and frame size is 2028 bytes. One uplink and one downlink rtPS service flow with parameters shown in Table 5.5 is configured at ST.

5.1.2.3 Results

In this section we will calculate Theoretical maximum number of users system can support in this scenario. And we will also compare queueing and end to end MAC delay with different slot sizes.

S.No	Parameter	value
1	slot duration	32 μs
2	contention slots	10
3	DL-UL ratio	2:1
4	frame duration	10 ms
5	Number STs	1

Table 5.4: rtPS Setup parameters

S.No	Parameter	value
1	Service Class Name	rtPS
2	Max Sustained traffic rate	100 Kbps
3	Min reserved traffic rate	90 Kbps
4	Max latency	8 Secs
5	Polling Interval	80 msecs

Table 5.5: rtPS service flow parameters

Theoretical Maximum number of rtPS users:

Original bandwidth request from ST is 90 Kbps.

Requested rate with MAC overhead is, $rate \times (\frac{\text{avg packet size+ MAC header size}}{\text{avg packet size}})$

$$90 \text{ Kbps} \times \frac{1500+5}{1500} = 90.3 \text{ Kbps}$$

Now convert rate from bits per second into symbols per second (sps),

rate in symbols per second $= \text{rate in bps} \times \frac{1}{\text{number of bits symbol} \times \text{coding rate}}$

$$90.3 \text{ Kbps} \times \frac{1}{1 \times \frac{1}{2}} = 180.6 \times 10^3 \text{ sps}$$

Then find out number of symbols required per frame

Number of symbols required per frame = rate sps $\times$ frame duration

$$180.6 \times 10^3 \times 10 \times 10^{-3} = 1800$$

Number of symbols per slot is, $\frac{\text{slot duration}}{\text{symbol duration}}$

$$= \frac{32}{.045} = 711 \text{ symbols}$$

Number of symbols needed for WiFiRe PHY overhead (96μs) is,

$$\frac{\text{phy overhead duration}}{\text{symbol duration}}$$

$$\frac{96}{.045} = 2133 \text{ symbols}$$

Total number of symbols required is, sum of the phy overhead and number of symbols requested

$$= 2133+1800 = 3933$$

Convert symbols per frame into number of slots per frame

$$\frac{\text{number symbols required per frame}}{\text{number of symbols per slot}}$$

$$\frac{3933}{711} = 5.5$$

So we need 6 slots. three data slots and three PHY slots.

Maximum number of connections per frame is

$$\frac{\text{total symbols per frame}}{\text{number of symbols requested}} \times \text{number of parallel tx}$$

$$\frac{220000}{3933} \times 2 = 111$$

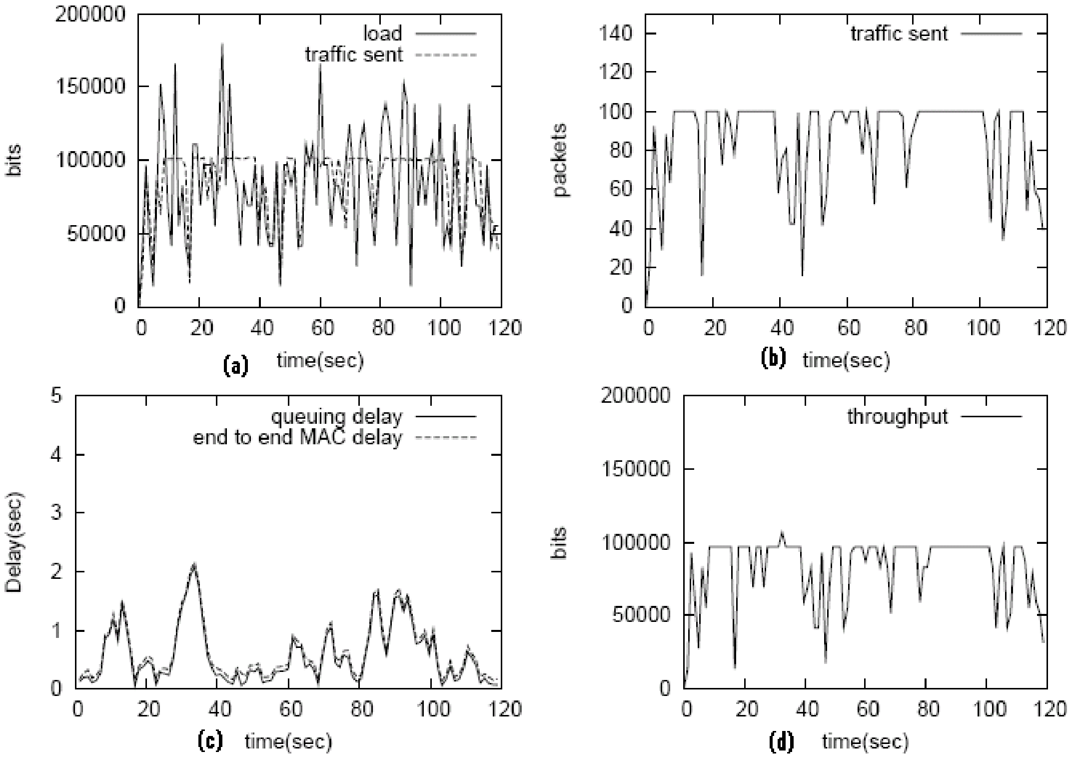
Because frequency reused in opposite sectors.
So BS can admit 111 Maximum number of rtPS connections with configuration shown in Tables 5.4 and 5.5.

The Figures 5.6 and 5.7 shows simulation results with configuration shown in Tables 5.4 and 5.5. The Figures 5.6 (a) shows load and traffic sent statistics for the rtPS service flow. In that Figure load is fluctuating between 150 Kbps and 50 kbps on average, but MAC is serving with almost constant rate at 100 Kbps.
MAC can send $((32 \times 10^{-6}) \times (11 \times 10^{6})) \times 3$ bits in one frame with three data slots.
traffic sent per second $= \frac{((32\times10^{-6})\times(11\times10^{6}))\times3}{10\times10^{-3}} = 105$ kbps.
So with three 32μs data slots it can send data at 105 Kbps. MAC can not send more than 105 Kbps, this can be observed in the Figure 5.6(a).

Figure 5.6(c) shows that queuing delay and end to end MAC delay are increasing upto 15^{th} second, becuase if you observe the load, it is increasing upto 15^{th} second and it is more than service rate 105 Kbps. So queuing delay started to increase, thus increasing the end to end MAC delay. Maximum end to end MAC delay is 2.2 seconds, so packets are not exceeding maximum latency 8 seconds. So no packets are dropped by destination MAC, hence throughput is same as load shown in Figure 5.6(d). Next experiment will discuss simulation results with slot size 45μs.

Above simulation results showing that, three 32 micro second slots are required to provide Video service.

Figure 5.7 shows the simulation results with slot size 45μs and remaining all simulation parameters are same as shown in the Tables 5.4. 5.5. Figure 5.7(c) shows that, queuing delay and end to end MAC delay is increased compared to previous experiment. With 45μs, 90 kbps needs two data slots. MAC can send $((45 \times 10^{-6}) \times (11 \times 10^{6})) \times 2$ bits in one frame with two data slots. Then traffic sent per second $= \frac{((45\times10^{-6})\times(11\times10^{6}))\times2}{10\times10^{-3}}$ $= 99$ kbps. Even though ST is requested 90 kbps, BS is allocating 99 kbps. This will cause bandwidth wastage. So bandwidth is going to be wasted if we allocate bandwidth in terms of slots. In previous experiment ST can send traffic at 105 kbps with three data slots. Service rate is less than previous experiment with slot length 32μs. So queuing delay and end to end MAC delay are slightly higher than previous experiment.



(a) load and traffic sent vs time. (b) packets sent. (c) queuing delay and end to end MAC delay vs time. (d) throughput vs time.

Figure 5.6: rtPS statistics with slot size 32μs.

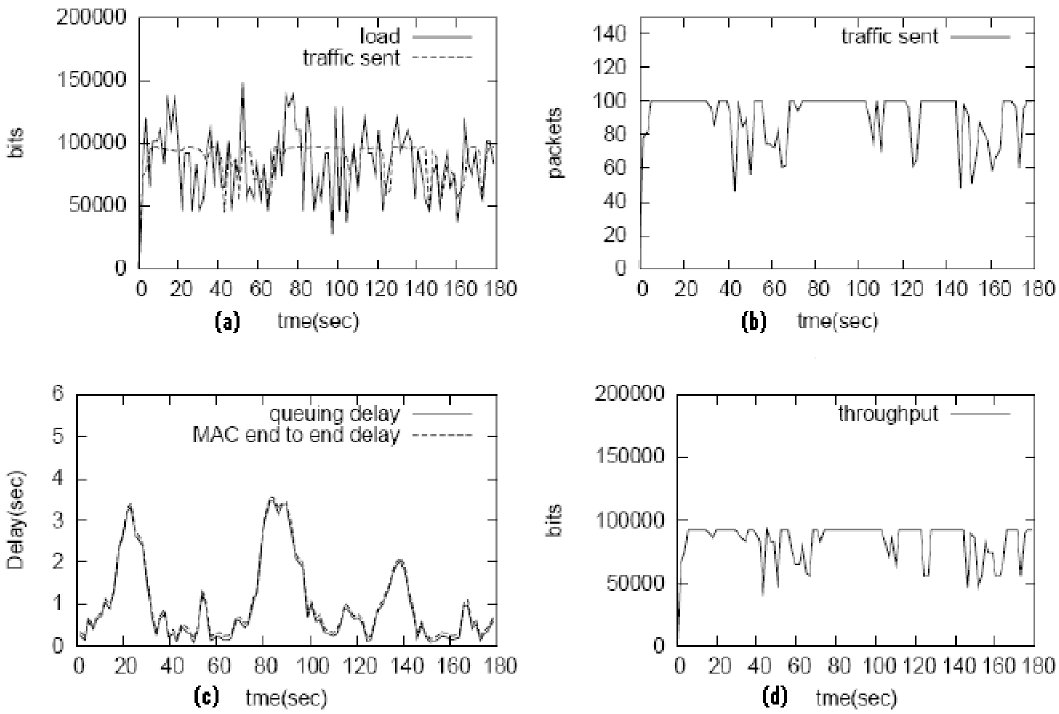
5.1.3 nrtPS Scenario

5.1.3.1 Purpose

This experiment discusses the effect of slot size on queuing delay and end to end MAC delay. This experiment also discusses the effect of polling interval on queuing delay and end to end MAC delay.

5.1.3.2 Setup parameters

The simulation parameters for this experiment are shown in the Tables 5.6 and 5.7. In this setup, polling interval is two seconds, which means that the BS will poll nrtPS flows in every two seconds. Then ST will send a bandwidth request equal to the queue size, if queue is not empty. In this experiment, FTP packet inter request time is exponentially distributed with a mean of 15 seconds and packet size is 5000 bytes.



(a) load and traffic sent vs time. (b) packets sent. (c) queuing delay and end to end MAC delay vs time. (d) throughput vs time.

Figure 5.7: rtPS statistics with slot size 45μs.

5.1.3.3 Results

Figure 5.8 shows the simulation results of FTP user scenario with setup parameters shown in Tables 5.6 and 5.7. In the Figure 5.8(c) queuing delay is more than two second for most of the time, because polling interval is two second. But end to end MAC delay is always more than 2 seconds because STs can send their traffic only after BS polled their connections.

In one slot MAC can send $(32 \times 10^{-6})(11 \times 10^{6}) = 352$ bits including WiFiRe headers. and 312 bits without WiFiRe overhead. in every 10 ms BS will serve 312 bits.

number of frames needed to send 8000 bit average SDU $= \frac{8000}{312} = 26$ approximately.

service time for 8000 bit SDU $= 10 \times 10^{-3} \times 26 = .26$ seconds.

minimum waiting time in the queue for each packet = 2+ .26+.01 (bandwidth request delay 10ms) =2.27 seconds.

But total waiting time of a packet in the queue, also depends on queue length at that time. Figure 5.8(c) shows that most of the time, queuing delay is varying between 2

S.No	Parameter	value
1	slot duration	32μs
2	contention slots	10
3	DL-UL ratio	2:1
4	frame duration	10ms

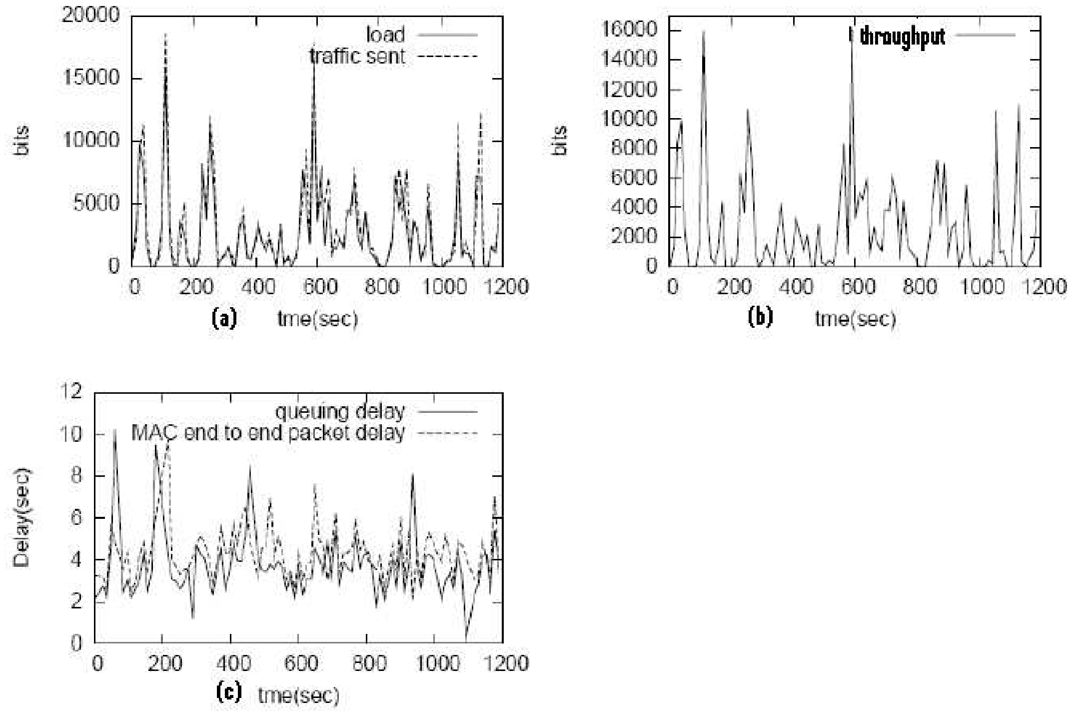
Table 5.6: nrtPS Setup parameters

S.No	Parameter	value
1	Service Class Name	nrtPS
2	Max Sustained traffic rate	20 Kbps
3	Min reserved traffic rate	10 Kbps
4	Max latency	10 Sec
5	Polling Interval	2 sec

Table 5.7: nrtPS service flow parameters

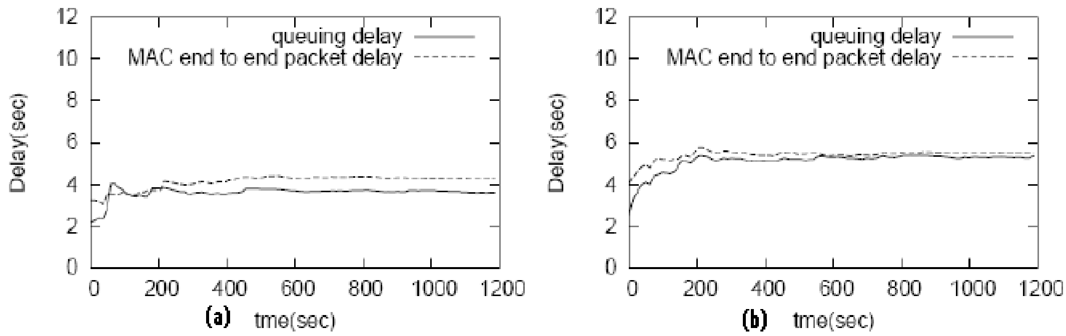
seconds and 8 seconds approximately.

Figure 5.9 compares queuing delay and end to end MAC delay with 2 seconds polling interval and with 3 seconds polling interval. Figure 5.9(a) and (b) shows that the end to end MAC delay and queuing delay increase when polling interval is increased. Delay values shown in this figure are averaged over time.



(a) load and traffic sent vs time. (b) throughput vs time. (c) queuing and end to end MAC delay vs time.

Figure 5.8: nrtPS statistics with slot size 32 μs and with 2 seconds polling interval.



(a) polling interval 2 seconds. (b) polling interval 3 seconds.

Figure 5.9: nrtPS queuing and end to end MAC delay comparison with slot size 32μs for different polling intervals.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Extensive simulation results shown that 32 μs slot length is sufficient to provide voice services. However rtPS flows need more than one slot which will increase *WiFiRe* overhead. If slot size is small, then MAC overhead will be increased due to segmentation. If slot size is large, then the packet will be delivered quickly to the destination MAC. Suppose if slot size is equal to SDU size + mac_header_size, then source MAC will deliver entire MAC PDU in one slot. So no additional segmentation overhead is incurred. But increasing slot length will decrease the maximum number of admitted connections. The above simulation results also shows the model correctness.

All results shown in Chapter 5, are single ST and single BS, we don't have results for multiple STs with single BS and multiple STs with multiple BSs.

With current slot scheduling algorithm, bandwidth is going to be wasted if the slot size is large. With this algorithm, even if ST requests less than one slot bandwidth, BS will allocate one slot bandwidth. We propose some extensions to solve this problem in future work.

6.2 Future Work

- Physical layer effects needs to be added to the simulation model. This needs changing pipeline stages of 802.11b PHY layer to simulate behavior of WiFiRe PHY layer.

- Changing existing static DSA Req procedure such that, ST sends DSA Req when MAC receives first data packet belongs to some service type.

- Dynamic Service Deletion Req needs to be added to the model.

- With present slot scheduling algorithm, there are chances of bandwidth wastage, as bandwidth assignment is in terms of slots. As described above, even though if ST has requested less than one slot per frame, BS will still allocate one slot for it. We can save bandwidth by allocating bandwidth in symbols per frame, like $< CID, number\_of\_symbols\_allocated >$ instead of $< CID, number\_of\_slots >$.

Bibliography

[1] Anand Kannan Ashok Jhunjhunwala and Bhaskar Ramamurthi. Interim broadband-wireless system for india. In *CEWiT Technical Report*, 2005.

[2] Sridhar Iyer (IIT Bombay), Krishna Paul (Intel), Anurag Kumar (IISc Bangalore), and Bhaskar Ramamurthi (IIT Madras). Broadband wireless for rural areas– *WiFiRe:* Medium Access Control (MAC) and Physical Layer (PHY) Specifications . August 2006.

[3] LAN/MAN standards Committee. Part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications. In *IEEE-SA Standards Board*, June 2003.

[4] Bhaskaran Raman and Kameswari Chebrolu. Revisiting mac design for an 802.11-based mesh network. In *San Diego, CA, USA: HotNets-III*, November 2004.

[5] LAN/MAN standards Committee, IEEE Microwave Theory, and Techniques Society. Part 16: Air interface for fixed broadband wireless access systems. In *IEEE-SA Standards Board*, June 2004.

[6] OPNET Documentation. http://www.opnet.com

[7] Xinjie Chang. Network simulations with OPNET. In *Winter Simulation Conference*, pages 307–314, 1999.

[8] Sridhar Iyer Bhaskar Ramamurthi Krishna Paul, Anitha Varghese and Anurag Kumar. Wifire: Rural area broadband access using the WiFi PHY and a multisector TDD MAC. In *IEEE Communications Magazine*, pages 111–119, 2007.

[9] Anirudha Bodhankar, 2006. Thesis on Implementation of WiFiRe Model in OPNET; A WiMAX-like MAC over WiFi PHY.