INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

Performance Evaluation of IEEE 802.14/MCNS MAC Protocol for LMDS/LMCS Networks

by

“Raman Grover, B.Eng”

*A thesis submitted to the
Faculty of Graduate Studies and Research
in partial fulfillment of the requirements
for the degree of*

“Master of Engineering”

Ottawa-Carleton Institute for Electrical Engineering
Faculty of Engineering
Department of Systems and Computer Engineering
Carleton University

“August 1998”

© copyright

“1998, Raman Grover”

National Library
of Canada

Acquisitions and
Bibliographic Services

395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale
du Canada

Acquisitions et
services bibliographiques

395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-36887-4

The undersigned recommend to
the Faculty of Graduate Studies and Research
acceptance of the thesis

**“PERFORMANCE EVALUATION OF IEEE 802.14/MCNS MAC
PROTOCOL for LMDS/LMCS Networks”**

submitted by

“Raman Grover, B.Eng”

in partial fulfillment of the requirements
for the degree of

“Master of Engineering”

“Professor David Falconer”
thesis supervisor

“Georgios Stamatelos”
thesis co-supervisor

“R.Goubran”, Chair, Department of Systems and Computer Engineering

Carleton University

Date “August 20, 1998”

Abstract

The growing demand for broadband services by residential customers has motivated many industries to extend the information super-highway to home. Various alternative access technologies (also referred to as last mile distribution networks) have been designed for different network infrastructures (mediums) to bridge the gap between high-speed backbone network and home. Systems such as Hybrid Fiber Coax (HFC) and Switched Digital Video (SDV) provide ways of pushing fiber further and further towards customer premises; while Asymmetric Digital Subscriber Line (ADSL) and very high bit rate Digital Subscriber Line (VDSL) use the existing twisted-pair loop for digital broadband services. Local Multipoint Distribution Services (LMDS) and Direct Broadcast Systems (DBS) use wireless medium to extend these services.

Local Multipoint Distribution Services (LMDS) is a broadband wireless point-to-multipoint communication system operating at 28 GHz with a bandwidth of 1GHz that can be used to provide broadband services to residential and commercial customers. The LMDS is becoming popular among various telecommunication industries because of its immediate benefits of ease and speed of deployment. The only possible way to extend broadband services to areas, where deployment of coax, fiber or copper is difficult (because of geographic location or cost), is to use wireless communication. To make this communication possible, many challenges in both physical and medium access control layer, need to be solved first. Besides, to fulfill the Quality of Service requirement (QoS)

for some of these broadband applications, an end-to-end solution is required that includes access network and its connectivity to high speed backbone network.

Both HFC and LMDS networks are competing alternative broadband access networks with similarities in their access environment, available capacity and targeted broadband services. This similarity has motivated us to look for the multiple access protocols compatibility among these networks. The work presented in this thesis focuses on a comparative study of multiple access protocol designed originally for HFC systems but has the applicability for LMDS networks.

The role of Medium Access Control protocol is to efficiently allocate the scarce upstream bandwidth among various users. Various Standard bodies (ATM Forum, ETSI, DAVIC, IEEE 802.14-WG, SCTE, MCNS) are working to come up with the different solutions for HFC and LMDS networks. We have considered three different TDMA based MAC protocols with capability of providing bandwidth on demand, for a comparative study. The selected protocols include the IEEE 802.14 MAC, DAVIC MAC and MCNS MAC. The fair and comparative study of the three protocols has been done to highlight the key differences of the three protocols using similar assumptions and reference scenarios.

Acknowledgements

I am grateful to all the people who have guided and helped me during my graduate studies. First and foremost, I would like to extend my thanks to Dr. Langis Roy for his support during my early days at University Of Ottawa. He funded me as a Graduate Research Assistant, and through a joint project, he introduced me to my research Supervisor, Professor D.D. Falconer. Dr. Langis Roy encouraged and supported me when I needed it most.

My eternal gratitude to my research advisor Dr. George Stamatelos, for his long hours of discussions on various problems. He actually taught me how to do research. I would like to extend my warmest thanks to Professor Falconer, for his great guidance and encouragement throughout the course of this work. The learning exposure I got under his supervision was incredible. Certainly working with him has been an extremely rewarding experience. I am very thankful to Professor L.Orozco-Barbosa and Professor R.M. Hafez for their valuable suggestions during my thesis defence.

I acknowledge all the people who have influenced my research. Especially from the Broadband Lab (Carleton University), I would like thank to Professor Michael Devetsikiotis, for his help and suggestions on traffic modeling. I am thankful to my friends Ms. Nausheen Naz, and Mr. Tariq Ali. I am also thankful to the Canadian Institute for Telecommunications Research for funding this project. Finally, my deepest thanks to my wife Neelu, for her endless love and understanding.

Table of Contents

Abstract	iii
Acknowledgements.....	v
Table of Contents	vi
List of Tables	x
List of Figures	xi
List of Acronyms	xv
<i>Chapter 1:</i>	
Introduction	3
1.1 Thesis Motivation	3
1.2 Thesis Objective.....	4
1.3 Thesis Organization	5
<i>Chapter 2:</i>	
LMDS/HFC System Architecture.....	6
2.1 Local Multipoint Distribution Systems.....	6
2.2 Hybrid Fiber Coaxial System.....	10
2.3 Role of Medium Access Control Protocol	12
2.4 Service Requirement.....	13
2.4.1 Non Real Time Applications	14

Table of Contents

2.4.2	Real Time Applications	15
2.4.3	Traffic Characteristics.....	16

Chapter 3:

MAC Protocol Components	18
3.1 Random Access Algorithms	18
3.1.1 ALOHA Based Algorithms.....	18
3.1.2 Splitting (Tree) Algorithms.....	22
3.1.3 Contention Interleaving Mechanisms	36
3.2 Upstream Frame Structure	37
3.2.1 Clustered Mode.....	37
3.2.2 Continuous Mode.....	44
3.3 Contention Slot Allocator	45
3.3.1 Data Slot Priority Algorithm.....	46
3.3.2 Contention Slot Priority Algorithm	48

Chapter 4:

Candidate MAC Protocols.....	49
4.1 IEEE 802.14 MAC CONVERGENCE DRAFT PROPOSAL	50
4.1.1 Architecture.....	51
4.1.2 ATM Interface	52
4.1.3 Downstream Channel.....	53
4.1.4 Upstream Channel.....	54
4.1.5 MAC Operation	56
4.1.6 Contention State Machines	63
4.1.7 Transmission State Machine	65
4.2 MCNS MAC Standard.....	67
4.2.1 MAC Highlights	68
4.2.2 MAC Frame Formats	68
4.2.3 MAC Header Format	70
4.2.4 Variable-Length Packets.....	71

4.2.5	ATM Cell MAC Frames	71
4.2.6	MAC Management Header	72
4.2.7	Request MAC Header	73
4.2.8	Extended MAC Headers	73
4.2.9	Upstream Bandwidth Allocation.....	74
4.2.10	Contention Resolution	74
4.3	DAVIC 1.2 MAC Recommendation.....	75
4.3.1	Downstream Frame Structure	75
4.3.2	Upstream Frame Structure	76
4.3.3	MAC Operation	77
4.4	Comparison.....	79
4.4.1	Major Similarities Between the Candidate Protocols	79
4.4.2	Major Differences Between the Candidate Protocols.....	79

Chapter 5:

Simulation Model and Results

5.1	Simulation Assumptions	81
5.1.1	Frame Structure.....	81
5.1.2	CS-DS Size	82
5.1.3	Contention Slot Positions.....	86
5.1.4	Contention Resolution Machines Interleaving.....	86
5.1.5	Contention Slot Allocation	86
5.1.6	Contention (Collision) Resolution Process.....	86
5.1.7	Newcomer Entry Persistence	87
5.1.8	Bandwidth Request Procedure.....	87
5.1.9	Feedback/Grant Frequency	87
5.1.10	Bandwidth Scheduling and Buffer Size.....	88
5.2	MAC Protocol.....	88
5.2.1	IEEE 802.14 MAC Model General Description	89
5.3	Traffic Models	91
5.3.1	The WWW Traffic Model	91
5.3.2	Bursty Traffic Model (Random Short)	93
5.3.3	Batch-Poisson Traffic Model.....	93
5.4	Performance Bounds and Simulation Results.....	94

5.4.1	Definition of Performance Parameters.....	94
5.4.2	Simulation Results	95
5.4.3	WWW Traffic Result.....	96
5.4.4	Random Short Traffic Results	100
5.4.5	Batch Poisson Traffic Model Results	103
5.4.6	Heterogeneous Scenario A.....	108
5.5	Effect of Channel Errors	112
5.5.1	Effect on Collision Resolution Algorithm.....	113
5.5.2	Channel Errors Simulation.....	117
5.6	Support of Variable Length Data Slots.....	120
5.7	Comparative Results of MCNS and 802.14.....	126
5.7.1	Batch Poisson Traffic.....	127
5.7.2	Random Short Traffic	128
5.8	Comparative Results of DAVIC and 802.14	131
5.8.1	WWW Traffic Results	131
5.8.2	Random Short Traffic Results	132
5.8.3	Batch-Poisson Traffic	133
 <i>Chapter 6:</i>		
Conclusions and Recommendations.....		135
6.1	Summary Of the Dissertation.....	135
6.2	Thesis conclusions	139
6.3	Contributions of the Dissertation	140
6.4	Recommendations for further study.....	141
BIBLIOGRAPHY		145

List of Tables

Table 2.1:	ATM-Based CBR Traffic Characteristics and Performance Parameters...	17
Table 2.2:	Video Games, Data, and Multimedia Services	17
Table 3.1:	Capacity of Random Access Algorithms.....	30
Table 4.1:	Field Description.....	56
Table 4.2:	Contention States	65
Table 4.3:	Transmission States	67
Table 4.4:	Generic MAC Header Format.....	70
Table 4.5:	MAC Management Messages.....	72
Table 4.6:	Important Extended Header Element Formats.....	73
Table 4.7:	Standard MAC Messaging	77
Table 5.1:	Reference Configuration for WWW, Batch Poisson and Random Short..	95
Table 5.2:	Parameters specific to WWW Traffic Simulation	96
Table 5.3:	Parameters specific to Random Short Traffic Simulation	100
Table 5.4:	Parameters specific to Batch Poisson Traffic Simulation.....	103
Table 5.5:	Simulation Configuration For Heterogeneous Scenario A	108
Table 5.6:	Contention State Errors.....	115
Table 5.7:	Contention Feedback State Descriptions	116
Table 5.8:	Configuration for Channel Error Test on Batch Poisson Traffic.....	117
Table 5.9:	Packet Segmentation and Overhead for Variable Length Packets.....	121
Table 5.10:	Packet Segmentation and Overhead for VL Packets contd.	122
Table 5.11:	Packet Segmentation and Overhead for ATM Cells.....	122
Table 5.12:	Bandwidth Efficiency Calculations for packet sizes for ATM and VL...	124
Table 5.13:	Configuration for ATM and VL PDU Experiment.....	125
Table 5.14:	Configuration for 802.14 and MCNS MAC	127
Table 5.15:	Configuration for 802.14 and DAVIC MAC.....	131

List of Figures

Figure 2.1:	US and Canada LMCS/LMDS Spectrum	7
Figure 2.2:	Cellular Layout of LMDS	8
Figure 2.3:	LMDS System Architecture.....	9
Figure 2.4:	Traditional Analog Cable Television Network Architecture.....	10
Figure 2.5:	HFC System Architecture.....	11
Figure 2.6:	Spectrum Allocation for two-way HFC network.....	12
Figure 3.1:	Instability of Slotted ALOHA.....	20
Figure 3.2:	Binary Tree Algorithm.....	24
Figure 3.3:	Ternary tree splitting algorithm with depth first scanning.....	25
Figure 3.4:	Ternary tree splitting algorithm with breadth-first scanning order.....	25
Figure 3.5:	Implications of ternary feedback	27
Figure 3.6:	Obvious first transmission rule for blocked access tree collision resolution algorithm	29
Figure 3.7:	Non-Obvious first transmission rule for blocked access tree collision resolution algorithm	30
Figure 3.8:	Framed Implementation of Ternary Tree.....	32
Figure 3.9:	New Arrivals Unblocking technique with Admission Time Boundary	35
Figure 3.10:	Parallel Tree Implementation.....	36
Figure 3.11:	Frame Structure with clustered contention minislots.....	38
Figure 3.12:	Fixed Size Fixed Ratio Frame Structures	39
Figure 3.13:	Fixed Size Variable Ratio Frame Structures	40
Figure 3.14:	Late Request Effect.....	41
Figure 3.15:	Minimum Implicit Frame Structure	42
Figure 3.16:	Minimum Implicit Frame Structure with delay components.....	43
Figure 3.17:	Frame Structure with distributed minislots.....	44

List of Figures

Figure 3.18:	Contention Slot Allocation	46
Figure 4.1:	Layer Model.....	51
Figure 4.2:	Relationship between 802.14 and ATM	52
Figure 4.3:	Downstream Channel Hierarchy.....	53
Figure 4.4:	Request Minislot Allocation Elements	54
Figure 4.5:	Concatenation of Minislots.....	55
Figure 4.6:	Request Minislot Protocol Data Unit (RPDU).....	55
Figure 4.7:	Upstream ATM Cell Format.....	55
Figure 4.8:	Upstream Variable Length Packet	56
Figure 4.9:	Registration Message Flow.....	57
Figure 4.10:	Ternary Tree Stack Visualization	62
Figure 4.11:	Contention State Machine.....	64
Figure 4.12:	Transmission State Machine.....	66
Figure 4.13:	IP Traffic Flow.....	67
Figure 4.14:	Generic MAC Frame Format.....	69
Figure 4.15:	Upstream MAC/PMD Convergence.....	69
Figure 4.16:	MAC Header Format	70
Figure 4.17:	Ethernet/802.3 Packet PDU	71
Figure 4.18:	ATM Cell MAC Frame Format.....	71
Figure 4.19:	Management MAC Header.....	72
Figure 4.20:	MAP Format	72
Figure 4.21:	Request MAC Header Format	73
Figure 4.22:	Extended MAC Format.....	73
Figure 4.23:	Allocation Map	74
Figure 4.24:	A possible scenario of Downstream Frame Structure.....	76
Figure 4.25:	Upstream Time Slot Structure	76
Figure 4.26:	Network Entry and Upstream Time Slot Allocation.....	78
Figure 5.1:	IEEE 802.14 , MCNS and DAVIC Contention and Data Slot Structures (with the specified assumptions).....	85
Figure 5.2:	Upstream Frame for ATM Mode.....	89
Figure 5.3:	A Sample of WWW traffic pattern.	91
Figure 5.4:	Message size distribution for Batch Poisson Model.....	93
Figure 5.5:	Mean Access Delay for WWW.....	98

Figure 5.6:	Mean Access Delay for WWW.....	98
Figure 5.7:	Collision Multiplicity for 3000 WWW Traffic Users.....	99
Figure 5.8:	Improvement by Using Piggybacking in WWW Traffic.....	99
Figure 5.9:	Request Delay Component of Access Delay for WWW (Adaptive CS:DS ratio case).....	100
Figure 5.10:	Mean Access Delay for Random Short Traffic.....	101
Figure 5.11:	Collision Multiplicity for Random Short Traffic Load = 70 %	102
Figure 5.12:	Mean Access Delay performance of Batch Poisson Traffic	105
Figure 5.13:	Confidence Interval - Batch Poisson Traffic without Piggybacking and Dynamic CS/DS ratio.....	107
Figure 5.14:	Scenario A (80% Random Short, 20 % Batch Poisson) Effect of Piggybacking for Batch Poisson	109
Figure 5.15:	Scenario A (80% Random Short, 20 % Batch Poisson) Effect of Piggybacking for Random Short.....	110
Figure 5.16:	Scenario A - Random Short (80 % High Priority) and Batch Poisson Traffic (20 % Low Priority)	110
Figure 5.17:	Example of Ternary Tree.....	113
Figure 5.18:	Effect of errors in Contention and Data slots for Batch Poisson, Fixed CS:DS (2:1) ratio	118
Figure 5.19:	Effect of errors in Contention and Data slots for Random Short Traffic.....	119
Figure 5.20:	Effect of Channel Errors on Delay Variation	119
Figure 5.21:	Comparison of upstream overheads for carrying Batch Poisson packets in VL PDUs vs. ATM PDUs.....	123
Figure 5.22:	Throughput Improvement by Variable Length Data Slots	124
Figure 5.23:	Throughput Improvement by using Variable Length Packets	126
Figure 5.24:	MCNS vs. 802.14 Batch Poisson Traffic.....	128
Figure 5.25:	MCNS vs. 802.14 Random Short Traffic	129
Figure 5.26:	MCNS vs. 802.14 Random Short Traffic - CDF	129
Figure 5.27:	Coefficient of Delay Variation MCNS vs. 802.14, Random Short Traffic	130
Figure 5.28:	Access Delay MCNS vs. 802.14, Random Short Traffic	130
Figure 5.29:	Mean Access Delay Comparison under WWW Traffic for DAVIC & IEEE 802.14 (without and with PB and Dynamic CS/DS ratio).....	132
Figure 5.30:	Mean Access Delay Comparison under Random Short Traffic for DAVIC & IEEE 802.14	133

Figure 5.31:	Mean Access Delay Comparison under Batch Poisson Traffic for DAVIC & IEEE 802.14	134
Figure 6.1:	Control Architecture of the IEEE 802.14 MAC Model.....	138
Figure 6.2:	Bandwidth Scheduling at LMDS hub based on knowledge of requests..	142

List of Acronyms

AAL	ATM Adaptation Layer
ABR	Available Bit Rate
ADSL	Asymmetric Digital Subscriber Line
AIU	Air Interface Unit
APDU	ATM Protocol Data Unit
ARQ	Automatic Repeat Request protocol
ATB	Admission Time Boundary
ATM	Asynchronous Transfer Mode
CATV	Community Antenna Television or Cable Television
CBR	Constant Bit Rate
CIL	Contention Interleave Identifier
CLP	Cell Loss Priority
CM	Cable Modem
CMS	Contention Mini Slot
CMTS	Cable Modem Termination Systems
CPCS	Common Part Convergence Sublayer
CPE	Customer Premises Equipment
CRA	Contention Resolution Algorithm

CRC	Cyclic Redundancy Check
CRI	Collision Resolution Interval
CRPP	Contention Resolution Priority Profile
CS	Contention Slot
CSA	Contention Slot Allocation
DA	Destination MAC Address
DAVIC	Digital Audio Visual Council
DOCSIS	Data Over Cable Interface Specifications
DS	Data Slot
EBR	Extended Bandwidth Request
EHDR	Extended MAC Header
EK	Encryption Key
ETSI	European Telecommunications Standards Institute
FC	Frame Control Identifier
FCC	Federal Communication Commission
FCFS	First Come First Serve
FDMA	Frequency Division Multiple Access
FEC	Forward Error Correction
FFB	First Fragment Boundary
FIFO	First In First Out
FTC	Fiber to Curb
FTTH	Fiber to the Home
HC	Headend Controller
HCS	Header Check Sequence
HEC	Header Error Check

HFC	Hybrid Fiber Coaxial (Cable Network)
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
ITU	International Telecommunications Union
LLC	Logical Link Control
LMCS	Local Multipoint Communication System
LMDS	Local Multipoint Distribution Services
MAC	Medium Access Control (Layer)
MCNS	Multimedia Cable Network Systems
MPEG	Motion Picture Enhanced Group
MPEG-2 TS	Motion Picture Enhanced Group Compression Algorithm 2 Transport Stream
NIU	Network Interface Unit
NMS	Number of Minislots
NVOD	Near Video On Demand
OOP	Object Oriented Programming
PDT	Payload Data Type
PDU	Protocol Data Unit
PHY	Physical (Layer)
PMD	Physical Media Dependent (Layer)
PON	Passive Optical Networks
PRMA	Packet Reservation Multiple Access
PSTN	Public Switched Telephone Network
PT	Payload Type
QoS	Quality of Service

RF	Radio Frequency
RMS	Request Minislot
RPDU	Request MiniPDU
RQ	Request Queue Number
RS	Reed-Solomon
RTD	Round Trip Delay
SA	Source MAC Address
SAR	Segmentation and Reassembly
SCTE	Society of Cable telecommunications Engineers
SLI	Source Local Identifier
SSCS	Service Specific Convergence Sublayer
TC	Transmission Convergence (Sublayer)
TDMA	Time Division Multiple Access
UBR	Unspecified Bit Rate
URL	Uniform Resource Locators
VBR	Variable Bit Rate .
VC	Virtual Channel
VPI/VCI	Virtual Path Identifier/Virtual Channel Identifier
VL	Variable Length Packets
VOD	Video On Demand
VPDU	Variable Length Protocol Data Units
WWW	World Wide Web

This page is intentionally left blank

This page is intentionally left blank

Introduction

1.1 Thesis Motivation

The growing demand for bandwidth intensive telecommunication applications has led to the development of many new access technologies that promise cost-effective delivery of multimedia, and broadcast services to a wide range of residential and commercial customers.

A variety of access technologies (such as xDSL, PON based FTTH /FTC, HFC, [62] satellites and terrestrial wireless networks) have emerged to fill the demands. Of these technologies, a point-to-multipoint wireless terrestrial technology is becoming popular because of its economical viability and ease of deployment.

Local Multipoint Distribution Service (or Local Multipoint Communication System in Canada) is a fixed broadband wireless point-to-multipoint communication system using millimeter wave signals to provide two-way voice, data, Internet, and video services [60]. The system architecture is cellular and employs multiple contiguous, or overlapping wireless cells covering a specified geographical area (up to 5-7 km). LMDS systems are promising to deliver broadband services with quality of service (QoS) guarantees. A suitable choice of multiple access scheme is one of the key factors in fulfilling these promises [25]. Earlier deployment of these systems has been done with more traditional FDMA schemes, but as the customer base is increased more sophisticated wireless, media access

control (MAC) will be implemented to enhance the traffic capacity of the network for wide variety of services.

1.2 Thesis Objective

The main objective of this work is to evaluate the performance of a suitable MAC protocol for the upstream channel of LMDS/LMCS systems. The HFC network resembles the LMDS/LMCS networks in the sense that both provide a broadcast medium in the downstream direction and a multiple access shared medium in the upstream direction, although with different latency. Therefore, we believe that the MAC protocols which are being developed for HFC networks can be applicable to LMDS/LMCS networks. Currently, many standards activities are underway by ATM Forum, DAVIC, ETSI and ITU in the context of LMDS, and others such as DAVIC (Digital Audio Visual Council) [41], IEEE 802.14 Working Group [27], SCTE-MCNS (Multimedia Cable Network Systems) for HFC systems [47]. There are three major protocols evolved by different standard bodies whose suitability was considered for LMDS networks.

- (1) LMDS specific DAVIC 1.2 MAC recommendation [41].
- (2) HFC specific IEEE 802.14 MAC proposal [27].
- (3) HFC specific MCNS specifications [47].

Therefore, the first specified objective is to perform a comparative study of HFC-specific IEEE 802.14 MAC protocol in relation to LMDS-specific DAVIC 1.2 MAC recommendation under similar assumptions. The performance evaluation of LMDS specific DAVIC 1.2 MAC recommendation has been undertaken by M.T. Ali and is presented in [44]. The second objective of this work is to perform a comparative study of HFC-specific IEEE 802.14

MAC protocol with another HFC-specific MCNS-DOCSIS protocol under similar assumptions.

1.3 Thesis Organization

The organization of this thesis is described below.

In *Chapter 2*, the system architecture of LMDS and HFC network is presented with a brief review on service requirements. The role of medium access control protocol is also described in this chapter. Similar discussion is presented in Chapter 2 of reference [44].

Chapter 3 describes the major components of MAC protocols under consideration. This includes contention resolution algorithm (CRA), upstream frame structure, and contention slot allocation (CSA). Similar discussion is presented in Chapter 3 of reference [44].

Chapter 4 provides a survey of the candidate MAC protocols. This includes an overview of standard protocols like, HFC-specific IEEE 802.14 MAC, LMDS-specific DAVIC 1.2 MAC and HFC-specific MCNS MAC. The important similarities and differences among the three protocols are also highlighted. Similar discussion is presented in Chapter 4 of reference [44].

Chapter 5 describes the performance evaluation of IEEE 802.14 MAC protocol. This includes description of the simulation model, simulation assumptions, and a brief discussion on the traffic models. A comparative performance of the IEEE 802.14 MAC with MCNS and the DAVIC MAC has also included.

Finally, *Chapter 6* summarizes the work presented in this thesis with the conclusions and recommendation for further research.

LMDS/HFC System Architecture

The purpose of this chapter is to describe the architecture of LMDS and HFC systems with a brief review of the service requirements, and spectrum availability. The role of Medium Access Control protocol is also presented in this chapter.

2.1 Local Multipoint Distribution Systems

Local multipoint distribution service (LMDS) is a broadband wireless terrestrial point-to-multipoint communication system operating at 28 GHz that can be used to provide digital two-way voice, data, internet, and video services. In July 1995, the Federal Communication Commission (FCC) allocated frequency bands in 27.5-28.35 GHz and 29.1-29.25 GHz for LMDS networks [61]. In 1997, Industry Canada also started allocating spectrum in the 28 GHz band for a similar system called Local Multipoint Communication System [42]. Figure 2.1 shows the current available spectrum for these systems.

LMDS Background

Traditionally microwave frequencies at 28 GHz band were only used for point-to-point radio communication because technology was not mature to consider these frequencies for any other application.

The concept of point to multipoint transmission was first proposed in 1986 by a microwave engineer named Bernard Bossard from Cellular Vision Company who believed that 28 GHz (Ka-band) millimeter wave radio frequency can be used for TV broadcasting to fixed subscriber [60]. His idea was to send low powered, high frequency signals over short distances rather than sending high powered, low frequency signals over a long distance. Even at that time these systems were not considered practical because no one believed that video signals will be able to propagate through rain (as water absorbs microwave frequencies) but today technology is available to mitigate the effect of rain attenuation by using low-cost high-power-amplifiers, highly-directional smart-antennas and other techniques to provide coverage. The recent advancement in low-cost high-speed MMIC and DSP technologies has made high data rate services on radio links a reality.

Figure 2.1: US and Canada LMCS/LMDS Spectrum

LMDS System Architecture

LMDS uses microwave signals (or millimeter wave signals) in the 28 GHz spectrum to transmit voice, video, and data signals in both directions within small cells 3-10 miles in diameter. The extremely high frequency and point to multipoint architecture limits the distance between transmitter and receiver. Therefore LMDS is being designed to be a "cellular" system, based on multiple, contiguous, or overlapping cells [26, 19]. The major

factors affecting the cell size are line-of-sight, analog vs. digital, overlapping cells vs. single transmitter cells, rainfall, transmitter and receiver antenna height and foliage density.

Figure 2.2: Cellular Layout of LMDS

The major challenge to deliver reliable services by LMDS is the hostile propagation environment at millimeter wave frequencies [23]. The system coverage may be limited to direct line-of-sight between the transmitter and receiver. Locations without direct line-of-sight (shadow areas) can be covered by using reflectors or amplifiers. Use of directional antenna at the receiver end and mounting receiver and transmitter antenna at greater heights also help to increase the signal strength. Overlapping of cells also enhance the coverage significantly than single transmitting cells. Use of digital signals can provide additional advantage because they are more tolerant to noise in a signal, and less susceptible to rain and foliage. Amount of local rainfall also influence the cell coverage because LMDS uses microwave signals which react with water and lose strength.

The cellular layout of LMDS systems and highly directional antennas have made it possible to reuse the allocated frequencies in each cell. The use of highly directional antennas and orthogonal linear polarization in adjacent cells can reduce the effect of mul-

tipath and adjacent cells interference. Fig 2.2 shows the overlapping cell layout of LMDS systems using orthogonal polarization in adjacent cells. Fig 2.3 [4, 25, 19] describes the point-to-multipoint architecture of these systems.

Figure 2.3: LMDS System Architecture

2.2 Hybrid Fiber Coaxial System

Community Antenna Television (CATV or Cable TV) was first introduced in the late 40's to extend over-the-air TV broadcasts to remote subscribers where antenna reception was poor. In the mid 1970's satellite delivery signal was fed into Cable TV systems which increased its penetration to residential areas that did not necessarily have any problem in receiving regular programs over-the-air, but to provide wider selection of TV channels, some of which were not available over-the-air. At that time the Cable TV systems were engineered as unidirectional broadcast systems transmitting analog signal (54-750 MHz) using tree topology as shown in Fig. 2.4 [18, 63]. The cable system has four basic components: head-end, trunk network, distribution network, and drops. The Head-end (H/E) is a central facility that broadcast signals onto a cable network. The Head-end collects TV programs from a number of sources like over-the-air broadcast television transmission, satellite transmission, land-based microwave transmission and distributes them over the coaxial cable along the branches of a tree like network to customer sites, which are located at the leaves of the tree. The distance between H/E to a customer can be tens of kilometers which can cause signal attenuation. Hence to provide a good quality, signal amplifiers are cascaded along the cable from H/E to customer.

Figure 2.4: Traditional Analog Cable Television Network Architecture

HFC System Architecture

Since 1989 Cable TV network has been upgraded by replacing a larger section of coaxial cable with fiber to increase system capacity, reliability, and signal quality. The resulting network is called a hybrid fiber coaxial (HFC) network. In the HFC network, the main coaxial “trunk cable” from H/E to the community neighborhood is replaced with high-reliability, low-attenuation fiber. This reduces the requirement of large number of active components between the H/E and customer leading to increased network reliability and better signal quality. The new HFC network has technical viability of two-way communication by using two-way amplifiers to allow upstream transmissions from a customer to the H/E.

Figure 2.5: HFC System Architecture

Two-way operation of HFC networks practical for two reasons: First, the fiber itself is not subject to ingress of interfering signals. Secondly, the cable system is broken into a large number of small cable systems, each isolated from the others by its own fiber link to the head end. Interference caused by ingress noise in one of these small cable systems, does

not impair the performance of the other portions of the cable plant. The broadband capability of the architecture has come by removing the bandwidth-limiting cascade amplifiers from the trunk cable. The architecture of two-way capable HFC network is shown in Fig 2.5. [1].

Figure 2.6: Spectrum Allocation for two-way HFC Network

With this new bidirectional capability of HFC networks, the FCC reserved 5 to 42 MHz frequency spectrum for upstream signals. Spectrum above 54 MHz is used for downstream transmissions [63]. Diplexers, or tuning filters, are used to separate the upstream and downstream signals. In order to be compatible with traditional analog CATV systems, the frequency spectrum between 54 MHz and 550 MHz (or 450 MHz), is used to carry current analog transmissions while the frequencies above that are used to provide enhanced switched or broadcast data services. These frequencies allocations are illustrated in Fig 2.6.

2.3 Role of Medium Access Control Protocol

The advantages of LMDS systems over other competing access technologies are lower infrastructure cost, ease of deployment and large available bandwidth. One of the ways to

keep the cost low in these systems is by increasing system capacity that can only be achieved with the efficient use of the available bandwidth. As upstream bandwidth is limited in both HFC (e.g. 37 MHz for historical reasons) and LMDS (possibly 150 MHz) systems. Whether it is the tree topology of HFC systems or the cellular layout of LMDS systems, bandwidth has to be shared among different users. The role of Medium Access Control (MAC) is to efficiently allocate the upstream bandwidth to multiple users sharing the network by providing bandwidth on demand. In detail, the Medium Access protocol is responsible for various functions: ranging or acquisition process, supervision of upstream and downstream physical layer devices, encryption/decryption, frame format, support for higher layer traffic classes, managing bandwidth allocation, bandwidth request mechanism, and collision resolution mechanism. There is architectural similarity between both systems; for example, both systems provide a broadcast medium in the downstream direction and multiple access shared medium in the upstream direction. Therefore, MAC protocols designed for one system can be applicable to the other systems with rather moderate modifications. If compatible MAC protocols, IF frequencies and bandwidths are specified for HFC and LMDS networks, a subscriber might be able to switch from one of these competing media to the other with minimal hardware changes.

2.4 Service Requirement

Both LMDS and HFC networks are competing technologies that will support digital data transmission to home and business customers. The targeted applications by these networks range from one-way video distribution and narrow band telephony to fully-interactive switched broadband multimedia applications. The ability of these networks to support these services is measured by the amount of bandwidth required in both upstream and

downstream directions and traffic characteristics mainly in the upstream direction. These applications can be broadly classified as non real time and real time [1, 6].

2.4.1 Non Real Time Applications

A full duplex digital communication link between the user and the LMDS hub (or head-end) allows access to all resources available through the backbone global network. *High speed Internet access* is probably one of the key applications being targeted by these networks. With Internet access, a user can access other networks which enables him to request data on demand, browsing news, weather forecasts, shopping catalogues etc. This application is also capitalizing on a vast amount of already available information contents. Other examples of non real time application class include the following:

Remote Education: Tapping into digital libraries and other information resources.

Data on Demand: Downloading data such as digital audio, or other information.

Home Office: Where a person's office is their own home and business is conducted from home and connection to clients is made through telephones and computers.

Electronic Commerce: Which represents electronic transactions made by interconnected computers. The *virtual mall* and remote shopping are examples of electronic commerce.

Telecommuting: Where users can tap into their office computer resources using a high speed connection from home and work as productive as in the office environment.

Traditional E-Mail Services: Sending and receiving electronic messages.

2.4.2 Real Time Applications

Digital VOD or NVOD: A VOD (video on demand) application is defined as the ability to check on any movie electronically, watch at home and have full VCR control capability. NVOD (near video on demand) application is similar to VOD without VCR control capability.

Voice Telephony: Traditional voice communication.

Video Telephony: Video telephony is the use of a TV or PC screen to the person or other image on the screen while talking. This application includes *video visit, alarm security monitoring, merchandising products*.

Electronic Games: Bidirectional capabilities of these networks can allow users to play electronic games which can be downloaded and played interactively between other users on the network.

Telemedicine: Telemedicine is the ability to provide high quality video telephony service interactively for the purpose of transmitting digitized medical data, x-ray images etc.

Telemetry: Remote sensing and Remote monitoring of gas, electric, and water utility, initiating, and disconnecting services.

Besides these applications, both LMDS and HFC networks can be used to extend high speed interconnection to local area networks, and medium to extend asynchronous transfer mode and Sonet to business customers.

2.4.3 Traffic Characteristics

The performance and capacity analysis of LMDS an HFC networks largely depends upon the traffic characteristics of the supported services. The tables below present some of the traffic characteristics and performance requirements for various types of applications supported by these networks. [1, 29, ANSI X.3.102, ITU-I.350]

- CBR traffic performance requirement. Table 2.1 is for voice, video telephony, and video on demand (VOD) services.
- Table 2.2 is for video games, data, and multimedia services. Included are requirement such as video games involving fast button pushes or mouse clicking, work-at-home, home shopping, and remote library access etc.

The term *PDU* in this table refers to a protocol data unit, which may be, for example, an ATM cell when the protocol uses ATM cells to package user data in the upstream and downstream. The Upstream and Downstream bandwidth values are on per-user basis.

Data Traffic: Delay tolerance characteristics of data traffic can be classified as:

- *Delay Sensitive:* e.g. interactive data services (CBR services)
- *Moderate delay-tolerant:* e.g. On line file transfers, remote library access (ABR services)
- *Highly delay-tolerant:* e.g. Email, off-line bulk data transfer (UBR traffic)

Video Traffic: Video traffic has stringent end-to-end delay requirements and must have low packet loss probability. It is also bounded to delay jitter requirement (Cell Delay variation).

Multimedia Traffic: In multimedia traffic, synchronization is probably the major issue. Synchronization becomes particularly important when voice and video are transmitted over different streams, delay tolerance for either stream should be less than 150 ms. Examples include voice telephony (lip sync.), slide show, two way audio visual communication.

Table 2.1: ATM-Based CBR Traffic Characteristics and Performance Parameters

	VOICE end-to- end access	VIDEO TELEPHONY end-to-end access	MPEG-2 VOD end-to-end access
Calls/busy hour/subscriber line	3	1	0.1
Call holding time	3 min	9 min	120 min
Call blocking probability	<0.001	<0.01	<0.01
Forward bandwidth	64 kbps	384 kbps	5-8 Mbps
Reverse bandwidth	64 kbps	384 kbps	(non-CBR)
Call control and management: messages per call	28 (Q.931/2931)	34 (Q.931/2931)	NA (Q.931/2931)
Call control and management: cells per call	70 (Q.931)	76 (Q.931)	NA (Q.931 and VCR remote control)
Dial tone delay	<300 ms	<300 ms	<300 ms
Call control and management: cell delay (99.99%)	<20 ms	<20 ms	<20 ms
Bearer information cell delay (99.99% end-to-end)	<150 ms	<150 ms	<150 ms
Information cell loss rate, lower limit	<10 ⁻³	<10 ⁻³	<10 ⁻⁶
Cell delay variation	<2-3 ms	<2-3 ms	<0.3-3 ms

Table 2.2: Video Games, Data, and Multimedia Services

	VIDEO GAMES	DATA (work at home)	MULTIMEDIA (home shopping)
Calls/busy hour	1	0.8	0.5
Call holding time	30 min	10-120 min	5-60 min
Forward bandwidth	0.5-4 Mbps	64 kbps-10 Mbps	64 kb/s-20 Mbps
Reverse bandwidth	30 PDU/s peak 3PDU/s avg. (sporadic)	10 Mbps peak 10 kbps-500 kbps avg.	64 kb/s-500 kbps
Call control/management: messages per call	85 (Q.931)	145 (Q.931)	85 (Q.931)
Call control/management: PDUs per call	127 (Q.931)	187 (Q.931)	127 (Q.931)
Call control/management: PDU delay (99.99%, upstream)	<=20 ms	<=20 ms	<=20 ms
Upstream information PDU delay (99.99%)	<=20 ms	<=20 ms	<=20 ms
Downstream information PDU delay (99.99%)	<=40 ms	<=100 ms	<=100 ms
Information PDU loss for low priority	<=10 ⁻³	<=10 ⁻³	<=10 ⁻³
Information PDU loss for high priority	<=10 ⁻⁸	<=10 ⁻⁸	<=10 ⁻⁸

MAC Protocol Components

Media Access Control protocol is a collection of components; each performs a certain number of functions, which include ranging and acquisition, defining a frame format, deciding ratio of contention and data slots in a frame (contention slot allocation), procedures for bandwidth request, contention resolution process and bandwidth allocation (scheduling) mechanisms. This chapter describes briefly some of these MAC components.

3.1 Random Access Algorithms

All the candidate protocols we have considered for this work are based upon random access of upstream channel. In this section, we provide a brief overview of different random access algorithms applicable and proposed in the context of three candidate protocols [9, 36, 17]. We can categorize Random access algorithms into two major categories:

1. ALOHA Based Algorithms
2. Splitting (tree) Algorithms

3.1.1 ALOHA Based Algorithms

ALOHA based algorithms are among the most popular algorithms because of their straightforward implementation. These algorithms works with a single feedback, i.e users

only receive positive feedback of their own successful transmission. So user does not monitor the activity on the network continuously. If the transmission of a user is unsuccessful, he will not receive any feedback and will reschedule for future transmission and remains inactive until that time. There are many alternatives for selecting the retransmission time.

- uniformly select a time interval over the next K time slots;
- transmit at the next time slot with probability p .

The parameters K and p can be either static or dynamic depending on the number of times a user has experienced collisions. The second case has been implemented as binary exponential back-off algorithm, which has been employed in the 802.3 Ethernet LANs; a slight modification has been implemented in MCNS - DOCSIS protocol.

3.1.1.1 Binary Exponential Backoff Algorithm

This algorithm was designed [61] to provide stability for access networks with limited feedback situations; it implies that users receive feedback only about their own successful transmission. Similarly to slotted ALOHA algorithm, users do not listen to the network activity continuously. The algorithm works in the following way.

A newly active user transmits a request immediately at the beginning of the next slot. In case of collision, the user backs off exponentially. It implies that a user experiencing m collisions, (re)transmits randomly after 2^m slots. The variable m could start at 0 and extends up to an upper limit M , which is commonly around 10. When m is exceeded the upper limit M , the message is discarded. The rationale behind this algorithm can be observed by noting that when a user first becomes active, he doesn't know about the backlog; therefore, immediate first transmission is reasonable. With successive collisions, retransmission times of collided users are separated farther and farther apart, since

repeated collisions may be an indication the contention load in the network has increased. The exponential back-off algorithm gives more transmission opportunities to new arrivals, thus resulting in a *quasi-last-in-first-out* transmission order. A slight variation of this algorithm is proposed in MCNS MAC protocol [47].

3.1.1.2 Instability of Slotted ALOHA Based Algorithms

Purely acknowledgment based versions of slotted ALOHA has been proven to be unstable [17, 36] analytically for very large population for any arrival rate (as close to 0 as desired), if they are operated for a sufficiently long time. Even in the case of *truncated binary exponential back-off* algorithm ($M < \infty$), many packets eventually experience M collisions and will be discarded for any arrival rate and system becomes unstable for large population. The issue of stabilizing ALOHA is addressed in the following section.

Figure 3.1: Instability of Slotted ALOHA

3.1.1.3 P-persistence Algorithm

This algorithm is proposed to stabilize ALOHA [17]. The difference between this algorithm and ALOHA is that new arrivals are considered to be a part of backlog immediately on their arrival and are not allowed to transmit with probability 1 in the next contention slot. Therefore, every user (including new arrivals) is allowed to transmit in the next contention slot with probability p . If the value of this retransmission probability (p) is too high, which means that users are allowed to transmit very often; the collision rate increases and delay of collision resolution becomes asymptotically high. But if the value of p is very small, then contention slots are wasted and new arrivals are delayed unnecessarily. Therefore, the performance of the collision resolution depends on the value of p . Throughput performance of this algorithm is given as

$$\eta = P_{\text{succ}} = N_k \cdot p \cdot (1 - p)^{N_k - 1} \quad (3.1)$$

Where N_k is the number of users contending in the k^{th} contention slot. The maximum throughput is achieved when p is equal to $1/N_k$, and approaches asymptotically to $1/e$ as the user population increases. With this algorithm, stability is achieved when load is less than the maximum achievable throughput i.e. $\lambda < 1/e$. So the maximum throughput and stability can be achieved with this algorithm when the value of p is adjusted dynamically to its optimal value $1/N_k$. In reality, we do not know about the exact number of contending users per contention slot, so this number has to be estimated. Rivest proposed an algorithm (Pseudo-Bayesian Estimator) to estimate the number of contending users per contention slot [57]. The aforementioned algorithm is called ideal p-persistence algorithm.

3.1.1.4 Rivest Pseudo-Bayesian Estimator

This algorithm was proposed to estimate the backlog user population on a slot to slot basis [57]. It implies estimating $\tilde{N}_k$ in the beginning of k^{th} contention slot. Each user is then required to transmit with probability given by $p_k = \frac{1}{\tilde{N}_k}$, in the k^{th} contention slot. The backlog estimation $\tilde{N}_k$ is given by equation 3.2. This algorithm can achieve a stable throughput of $1/e$ for $\lambda < 1/e$. In this algorithm, the arrival rate λ is an unknown parameter which can be estimated from time-averaged rate of successful transmissions; but we need accurate estimation of $\tilde{N}_k$ only when the system is near overload. Thus, λ can be safely estimated as $1/e$, and this substitution has been proved to be stable by Mikhailov [17]. The estimation of backlog population can be done either at LMDS hub (headend) or by user. In the former case, the retransmission probability is broadcasted by LMDS hub. In the later case, a user can estimate retransmission probability by monitoring the activity of the network.

$$\tilde{N}_{k+1} = \begin{cases} \max(\lambda, \tilde{N}_k + \lambda - 1) , & \text{for Empty/Success in CS} \\ \tilde{N}_k + \lambda + (e - 2)^{-1} , & \text{for Collision in CS} \end{cases} \quad (3.2)$$

This algorithm was one of the candidate in the MAC proposals in IEEE 802.14 WG [14, 15, 49].

3.1.2 Splitting (Tree) Algorithms

To break the barrier of $1/e$ throughput-bound, splitting (tree) collision resolution algorithm was invented [35]. This algorithm also maintains stability without any complex estimation procedure. In this algorithm, users continuously observe the downstream for the feedback

of their transmission and retransmission. Whenever more than one user tries to access the channel (which is time slotted) at the same time, collision occurs. Users involved in original collisions split themselves into N subsets. If a collision occurs again, the same process is repeated until all the collisions are resolved. The rationale behind this algorithm is to resolve collisions as soon as they occur by exploiting feedback information and controlling the retransmission process. In one version of this algorithm, new packets arriving to the system are inhibited from transmission until all the previous collisions are resolved. The time interval associated with each collision resolution is called *Collision Resolution Interval* (CRI). The length of CRI is a random variable which largely depends upon the collision multiplicity and transmission rules for new arrivals. There have been many studies done to optimize the performance of a tree algorithm. We will briefly provide a survey of different alternatives presented in the literature. We can classify these algorithms based upon the type of feedback, splitting parameter, and admission rules for new arrivals.

3.1.2.1 N-Tree Protocol

The principle of the n -ary tree-based contention resolution algorithm is to split the collided users into n subsets. Each of the subset then selects a random number between 1 to n . The basic idea is to allow different subsets resolving their collisions sequentially [17, 36, 35]. The users in the first subset retransmit first, while subsets 2 to n wait for their turn. The waiting subset of users can be visualized as a stack. The position in the stack represents the number of slots the users have to wait before transmitting their request. If a second collision occurs, the first subset splits again. The subsets that are already waiting in the stack must be shifted up by $n-1$ position in the stack to leave room for the new users that collided. If no collision occurs, the users occupying the lowest stack level can transmit. When $n=2$, the algorithm is referred to as binary tree; when n is 3, the algorithm is referred to as ternary tree.

Figure 3.2 shows as an example of the binary tree (splitting value $n = 2$), with U representing the number of users that have transmitted in a slot. If two or more users attempt to transmit in a slot, collision occurs. This figure shows that a collision occurs in the first slot. Hence, users involved in that collision split themselves into two subgroups, one of the subgroups transmits and the other defers until the previous group is completely resolved. Here CRI is shown as 11 slots.

Figure 3.2: Binary Tree Algorithm

Figure 3.3 shows an example of ternary tree; in this case, the users involved in the collision are also shown in a slot. Initially, 6 users collided in the slot #1, these users split into three groups with each user selecting one of these subgroups randomly. Here users no. 1,4,6 select to join the 1st sub group, user no. 2,3,5 select to join the third group, and no one selects to join the second group. So now the first sub group will transmit in slot number 2, and others will wait until this subgroup is completely successful. As we see again collision occurs in slot no. 2, and ternary splitting is repeated until the complete subgroup is resolved in 8 slots, and then the scanning of the other branch starts. This is an example of *depth-first* scanning order. Fig 3.3 shows a similar example in *breadth-first* scanning order. Notice the difference in slot numbers.

Figure 3.3: Ternary tree splitting algorithm with depth first scanning

Figure 3.4: Ternary tree splitting algorithm with breadth-first scanning order

3.1.2.2 Hybrid N-Tree Algorithm

The splitting parameter N could be changed dynamically in each time-slot. Ideally, if we estimate the collision multiplicity in a slot, the splitting parameter can be changed accordingly to resolve the collisions effectively, but it is questionable of how well we can estimate the number of users colliding in a slot. There has been a recent study about the implementation of dynamic collision resolution for wireless medium using binary feedback [16]. The collisions were resolved by pre-defined *Transmission Splitting Vector*, $TSV = \{n_1, n_2, n_3, \dots, n_k\}$, in which all the collided users are first split into n_1 subgroups after the first collision, n_2 subgroups at the second collision, and n_i subgroups at the i^{th} collision. For the $(k+1)^{\text{th}}$ collision, the system rotates back to first value of n , i.e n_1 . This scheme has been proved to outperform the fixed $n=3$ optimum case. The TSV is increased exponentially to certain a value and could be fixed thereafter. Simulation results for $TSV = \{3, 3, 3, 3, 2\}$, $\{2, 4, 8, 2, 2, 2, 2, 2\}$, have shown in a recent publication. The throughput achievement was up to 0.5, which is almost 9% more than the fixed $n=3$. However, more investigation is required for Dynamic Collision Resolution Implementation.

3.1.2.3 Binary vs. Ternary Feedback

In a binary feedback system, only two types of status are notified: Collision and No Collision. While in a ternary feedback system, three types of status are notified: Collision, No Collision and Idle [9]. Ternary feedback provides some advantages over binary feedback system by skipping sure collisions, but the ternary feedback mechanism is less robust in the presence of feedback errors. A situation called *deadlock* can occur in this system. As shown in figure 3.5 slot #1 was empty but due to error, collision was detected and binary splitting was performed. Slot # 2 was empty again, so system assumes that sure collision will occur in the next attempt; there is a split again and so on. These situations can be

taken care of, by forced termination of CRI after a maximum allowable consecutive idle feedbacks.

Figure 3.5: Implications of Ternary Feedback

3.1.2.4 Blocked-Access Splitting

In this algorithm, new packets are not permitted into the system until a collision is resolved. They are allowed to transmit only when all the other users previously involved in the collision, have transmitted successfully. The maximum theoretical throughput for blocked access with ternary splitting is 36.62% [36]. In blocked access, the order of resolution is preserved. A user who makes the first transmission attempt, is guaranteed to transmit successfully before a user who makes the transmission later, but the order between the users involved in the same collision resolution is randomly changed. Therefore the final order is not FIFO. A slight modification of this algorithm is implemented in IEEE 802.14 MAC protocol.

3.1.2.5 Free Access

In this algorithm, new packet arrivals can get involved into the on-going collision resolution process [36]. They are allowed to transmit immediately at the beginning of the next slot following their arrival. The maximum theoretical throughput for free access algorithm

with ternary splitting is 40.15%. This algorithm was one of the candidate in MAC proposals in IEEE 802.14 WG [7, 11, 40].

3.1.2.6 Soft Blocked Access

In this algorithm, we can permit new packet arrivals to join the on-going Collision resolution with a certain probability p .

3.1.2.7 First Transmission Rule

First transmission rule is a rule that specifies first transmission attempt of a new arrival [48]. There are two types of first transmission rules specified in the literature.

3.1.2.7.1 Obvious First Transmission

A free-access algorithm allows new arrivals to transmit immediately in the beginning of the next available slot [9]. This implies that they can join any current on-going collision resolution process. In the case of blocked access, the situation is slightly different. In this algorithm, first-transmission rule needs to be defined for the blocked users to access the network. The obvious first transmission rule is very simple; it implies that all the new arrivals blocked during a CRI, will attempt their first transmission in the first available slot after CRI (See Fig. 3.6). As we can clearly interpret from figure 3.6, with the obvious first transmission rule, the number of packets transmitting in the first slot after the current CRI is finished is positively correlated to length of current CRI. This implies the next CRI which will start because of multiple transmissions in the same slot will depend upon the length of previous CRI. This correlation could degrade the performance of the system as larger CRI will lead to larger CRI.

Figure 3.6: Obvious first transmission rule for blocked access tree collision resolution algorithm

3.1.2.7.2 Non-Obvious First-transmission Rules (FCFS)

To solve this problem, the non-obvious first transmission rule is proposed in which only a subset of new packet arrivals are allowed to transmit [9]. The enabling interval Δ is decided based upon the arrival time of packets, and in case of collisions, the enabling interval is split into two and the packets which arrive into first sub-interval (first half), are allowed to transmit before the second sub-interval. Therefore, the arrivals are transmitted in the same order in which they have arrived, maintaining a FIFO (first-in-first-out) order. Note the splitting mentioned here is binary splitting. The value of the enabling interval Δ has to be transmitted in the downstream. The enabling interval can be changed dynamically, according to the load in the system. The maximum stable throughput that can be achieved with this algorithm is 46.2% [17]. A slight refinement to this algorithm called elaborate time splitting using ternary feedback can achieve 48.72% throughput. Table 3.1 summarizes the throughput of different random access algorithms [9].

Figure 3.7: Non-Obvious first transmission rule for blocked access tree collision resolution algorithm

Table 3.1: Capacity of Random Access Algorithms

Random Access Algorithm	Capacity	Feedback	Comments
ALOHA	0	ACK	easiest to implement, unstable
binary splitting (blocked access)	0.3465	C/NC	obvious first transmission rule for new arrivals
binary splitting (free access)	0.3678	C/NC	free access
stable "ALOHA"	0.3678	C/NC	estimating backlog
ternary splitting (blocked access)	0.3662	C/E/S	obvious first transmission rule
ternary splitting (free access)	0.4015	C/NC	free Access
ternary splitting (blocked access)	0.4294	C/NC	non-obvious first transmission rule
ternary splitting	0.4622	C/E/S	non-obvious first transmission rule
ternary splitting	0.4872	C/E/S	elaborate time splitting algorithm
ternary splitting	0.5323	k	k: number of packets collided (theoretical case)

3.1.2.8 Un-Blocking Ternary Tree Implementation (IEEE 802.14 Standard)

In a strictly blocked access algorithm, new arrivals are blocked until the current contention has been resolved. This algorithm becomes inefficient for system with round trip delay greater than one contention minislots ($RTD > 1$ slot). According to this algorithm, all the contention minislots appearing in the middle of contention resolution will be left unused to maintain a strict FIFO order. If these unused contention minislots are opened to new users, the utilization of contention minislots increases. This scheme is called *un-blocking* (or *quasi-blocking*) of new users [3]. In framed (frame size $> RTD$) implementation of ternary tree algorithm, feedback of all the request attempts in a frame is provided at inter frame boundary (section 3.2). So there is always a possibility of some contention minislots left unused in fully blocked access algorithm. Consider the same example (figure 3.3) of ternary tree contention resolution algorithm implemented in framed format for systems with $RTD > 1$ slot as shown in figure 3.8. In this example every frame has 4 contention slots, so according to this example, in frame i all the six users contend in the first contention minislots, now the feedback of this transmission attempt is provided at boundary of frame i and $i+1$, so all the users will have to wait until next frame ($i+1$) to retransmit again. In frame $i+1$, collided users split in a group of three. There are four contention minislots in frame $i+1$, out of which the first three contention minislots are used by three groups; (1,4,6), (Empty), (2,3,5), and the fourth contention minislots is left unused. As we can observe that collision occurs in first (1,4,6) and third (2,3,5) contention minislots. Frame $i+2$ again has four contention minislots. First three contention minislots are used for three groups made by splitting (1,4,6) group. According to *depth-first* tree scanning order, the fourth contention minislots should be left unused but this will be again a wastage of this contention minislots, because headend and collided users in group (2,3,5) have the information about the collision. One possible way is use this fourth contention minislots for one

of the subgroup made by splitting group (2,3,5). In the frame $i+3$, again *depth-first* scanning order is regained, and contention minislots are allocated to three subgroups of group (4,6). The resulted scanning order is called *depth priority* (or *best effort depth-first*) scanning order. Similarly allocation is done in frame $i+4$. Now in this process, if we open the unused contention minislots marked as “U” for new arrivals (shown as *entrance*), the algorithm is called unblocking ternary tree algorithm. Although not shown but it should be intuitive that collisions also occur when we open some of the contention minislots to the blocked users, especially if this opening is after long time (section 3.1.2.7.1). These collisions could change the way contention minislot are allocated, but whatever way the tree is evolved, *depth priority* [20] scanning order should be preserved in this algorithm. (see example in section 4.1.5.3.1). The issues related to collisions in entrance contention minislots and first transmission rules are discussed in the next section.

Figure 3.8: Framed Implementation of Ternary Tree

3.1.2.9 Newcomer Unblocking Algorithms

In an unblocking ternary tree algorithm, unscheduled contention minislots are opened to new arrivals [48]. Under heavy load conditions it is possible that a few contention minislots are opened for new arrivals after a long time. This implies that all the blocked new arrivals will transmit their request in these contention minislots, which could lead to very high collision multiplicity, which in turn could trigger very long collision resolution interval (CRI) and further blocking of new arrivals (section 3.1.2.7.1). To avoid this problem, it was proposed [48] to allow new arrivals to transmit in the opening contention minislots with a certain probability p which can be dynamically adjusted by the headend controller (HC), depending upon network load condition. This scheme was implemented by using a *Range Parameter* R in the MAC proposal. The significance of R is that it provides a range to random generator for each new arrival. For example at certain instances, if the number of contention minislots (NMS) opened for new arrivals are 4, and the Range parameter R is 8. This means the probability for each new arrival to transmit its request is 0.5. The Five different methods to calculate “ R ” that have been suggested with various degree of complexity are:

3.1.2.9.1 Method 1

$$R(j+1) = \max \left\{ \min \left[n, R(j) - \text{CMS}(j) + \text{col}(j) \cdot \frac{(e-1)}{(e-2)} + \frac{\text{CMS}(j)}{e} \right], \text{CMS}(j+1) \right\} \quad (3.3)$$

where n is the total number of users supported by the access network, $\text{CMS}(j)$ is the number of contention minislots opened for new arrivals in the j th frame, and $\text{col}(j)$ is the number of collided minislots in the j th frame. $R(j)$ is the range parameter in j th frame, $R(j+1)$ is the range parameter in the $(j+1)$ th frame. $\text{CMS}(j+1)$ is the number of contention minislots opened for new arrivals in $(j+1)$ th frame and e is exponential constant. This algorithm was

originally proposed by Zenith for the MAC proposal in the IEEE WG, the idea was to implement the Rivest algorithm on frame to frame basis, and estimate new arrivals, not the total backlog which includes collided users as well. In this algorithm, The range parameter adapts itself to the traffic load, and never exceeds the number of contention minislots opened for new arrivals in a current frame ($p \leq 1$).

3.1.2.9.2 Method 2

$$R(j+1) = \min\{\max([CMS(j+1), \alpha \cdot RQ(j+1)], n)\} \quad (3.4)$$

where n is the total number of users supported by the access network, $CMS(j+1)$ is the number of contention minislots in frame $j+1$, α is the design parameter whose value is set to be 1.9 [48]. $RQ(j+1)$ is the length of Request Queue at headend at the beginning of $(j+1)$ th frame. RQ is an identification associated with each collision. For example, if two collisions occur in a frame, RQ number of 1 and 2 will be associated with each collision. In ternary splitting, three contention minislots are allocated to each RQ number. Normally an RQ number 0 is associated with contention minislots for new arrivals. (see section 4.1.5).

3.1.2.9.3 Method 3

$$R(j+1) = \max\{3, CMS(j+1)\} \quad (3.5)$$

This method simply allocates Range R to be the number of contention minislots in a current frame except in the cases when $CMS < 3$, R is set to be 3, so the value of p varies in steps of 0.33.

3.1.2.9.4 Method 4

In this method, R is set as the number of available contention minislots in the current frame and the value of p is 1.

$$R(j+1) = CMS(j+1) \quad (3.6)$$

3.1.2.9.5 Method 5

This method was recently proposed in the 802.14 WG [8]. The rationale behind this algorithm was to provide transmission opportunities to new arrivals in a FIFO order. In the previous methods, it was possible that some late arrivals get the opportunity to transmit, while early arrivals are waiting. To implement this scheme every time a parameter called Admission Time Boundary (ATB) is calculated by Headend and broadcasted, and new arrivals compare ATB with their own packet arrival time and transmit, if their packet arrival time is lesser than ATB as shown in figure 3.9. The parameter ATB can be calculated from the range parameter ' R ' by using equation 3.7. This algorithm is under the process of approval in 802.14 MAC draft [28].

Figure 3.9: New Arrivals Unblocking technique with Admission Time

$$ATB_{\text{new}} = ATB_{\text{old}} + \frac{CMS(j+1)}{R(j+1)+1} \cdot [T_{\text{current}} - ATB_{\text{old}}] \quad (3.7)$$

3.1.3 Contention Interleaving Mechanisms

The concept of contention interleaving or parallel tree was introduced for access systems where propagation delays are large. Certainly HFC systems fall into this class, as the propagation delay is larger than 400 μ s, and so these mechanisms have application to HFC systems. The rationale behind these mechanisms is that each user attempting transmission gets feedback after a round trip delay. So users cannot transmit because they are waiting for feedback. Massey proposed [36] to implement parallel trees, which means several duplicates of the tree algorithm running in parallel. Each duplicate tree is also referred as an interleave. In this mechanism, the number of interleaves used is equal to the number of contention minislots in a round trip delay. So all consecutive contention minislots assigned to a particular tree algorithm are separated by a round trip delay. This approach achieves the maximum efficiency bound at the cost of increased delay. Figure 3.10 shows the implementation of a ternary tree example provided in section 3.1.2.1, figure 3.3 using one and two parallel tree algorithms for a system with $RTD < 1$ and $RTD = 2$ slots respectively. As we can see, in the first case, the original collision is resolved in 16 contention minislots, and in the second case it is resolved 31 slots. In this mechanism unscheduled contention minislots can also be opened for new arrivals (*unblocking, first transmission rules*) and scanning order variation (*breadth first or depth first*) are also applicable.

Figure 3.10: Parallel Tree Implementation

3.2 Upstream Frame Structure

This section focuses on the issue of the frame structure in the context of IEEE 802.14 MAC draft. IEEE 802.14 MAC has not imposed any explicit upstream frame structure, so the protocol can be implemented with an “implicit” frame structure. 802.14 MAC protocol is based upon the concept of minislots in the upstream channel, (or a *sea of minislots*). Minislots are primarily used to request permission on contention basis to transmit information packets in the upstream channel (hence referred to as contention or request minislots), and the information is subsequently transmitted in data slots (generated by concatenating minislots) allocated by the headend controller (HC). Collisions can occur in contention minislots and a contention resolution algorithm is used to resolve collisions. For each transmission in the contention minislot, the headend transmits a feedback message in a downstream channel to inform the user about the contention status (collision or success) of transmission attempts in the contention minislots. The feedback message is subjected to a round trip propagation delay ($2t_p$) due to the physical distance between headend and station. To process every transmission attempt the headend takes some finite time, known as headend controller processing delay (t_{HC}). Similarly, a station also process feedback/grant message known as station processing delay (t_s). Based upon this information, few implicit frame structures can be generated by headend in the following ways:

- By grouping contention minislots in a frame (clustered mode).
- By distributing contention minislots in a frame (continuous mode)

3.2.1 Clustered Mode

In this case contention minislots are clustered adjacently at the beginning of each upstream frame[53, 13]. Further in this case we can have fixed length and variable length frames,

and the ratio of number of contention minislots to the number of data slots in a frame can be fixed or variable.

Figure 3.11: Frame Structure with Clustered Contention Minislots

3.2.1.1 Fixed Size Fixed Ratio Frame Structures

In fixed size, fixed ratio frame structure, the frames are always of the same length, and the contention periods are always the same, i.e. the ratio between contention minislots and data slots is always same. This frame structure is very simple to implement and requires less downstream overhead (feedback/grant once every frame at the boundary of frame), but the disadvantage is that there is always the possibility of some data slots remaining vacant. This implies that some bandwidth will be wasted, which leads to lower efficiency. The frame duration T_F , for this case is lower-bounded to (see figure 3.11 and figure 3.12)

$$T_F = t_C + t_D \geq t_C + 2t_p + t_{HC} + t_S \quad (3.8)$$

where t_C is the total time duration of the contention minislots placed at the beginning of a frame; $2t_p$ is the round trip propagation delay; t_{HC} and t_S are the headend controller and station processing times, t_D is total time duration of data slots.

Figure 3.12: Fixed Size Fixed Ratio Frame Structures

3.2.1.2 Fixed Size Variable Ratio Frame Structures

In this case, the size of each frame is fixed but the ratio of contention minislots and data slots varies from frame to frame [53]. This enhancement leads to the use of vacant data slots by converting them into contention minislots. The improvement is obvious as more contention minislots are available for the same load and therefore suffer fewer collisions in them. The ratio of contention minislots and data slots has to vary dynamically with a real time algorithm (Contention Slot allocation algorithm, section 3.3) which has to operate at the headend. The frame duration T_F , for this case is lower-bounded as follows (see figure 3.11 and figure 3.13)

$$T_F = t_C + t_D \geq t_C + 2t_P + t_{HC} + t_S \quad (3.9)$$

where t_C is the total time duration of the contention minislots placed at the beginning of a frame; $2t_P$ is the round trip propagation delay; t_{HC} and t_S are the headend controller and station processing times, t_D is total time duration of data slots. Here t_C and t_D are both variables, but T_F is fixed. If $t_D < t_R$ where $t_R = 2t_P + t_{HC} + t_S$ is total round trip delay, then the *late request effect* will be observed. By referring to figure 3.14, we can see that all the requests in the frame j received before transmission of *Inter-Frame-Boundary* (IB) grant packet, will be granted data slots in the frame $j+1$, but if a request is transmitted less than t_R seconds before the reception of the IB packet, then the data slot would be granted two frames later. This effect is called *late request effect* [53].

Figure 3.13: Fixed Size Variable Ratio Frame Structures

Figure 3.14: Late Request Effect

3.2.1.3 Variable Size Variable Ratio Frame Structures

In this case neither the size of frame nor the ratio of contention minislots and data slots is fixed. Both of these parameters are updated by the headend with some complex algorithms. We will discuss only “*The Minimum Implicit Frame Structure*” under this class.

3.2.1.4 Minimum Implicit Frame Structure

In this case feedback/grant is provided asynchronously rather than at every inter frame boundary (see figure 3.15). The contention periods are implicitly closed by the reception of the grant in the downstream [53]. Contention periods restart when all current schedules have expired. The advantage of this frame structure is that upstream frame structure can be independent of any downstream frame structure, and the upstream frame structure can adapt itself to the traffic conditions. There is no requirement of any real time contention slot algorithm and most important of all is that the access delay is independent of the frame size. The disadvantage is that the feedback/grant messaging is very frequent and may lead to waste of bandwidth in downstream, since now a separate grant message is transmitted for every successful request, rather than transmitting grant message at every inter frame boundary for many requests.

Figure 3.15: Minimum Implicit Frame Structure

Figure 3.16: Minimum Implicit Frame Structure with delay components

Figure 3.16 shows the minimum implicit frame structure with delay components. In “frame- i ”, the first contention minislot results in a collision. The second contention minislot results in a successful transmission. The headend processes this information and takes a finite amount of processing time (t_{HC}). Then headend processes a grant for this request. The grant message is received by a station (user) after the propagation delay (t_p). Then the station requires (t_s) seconds to process this information, and the data is received after (t_p) duration at headend. The duration (t_D) represents the total time duration of the data slots transmitted by this station (transmission time). So the reception of a grant by a station automatically starts the data period, and the expiration of grant will again restart the contention period. Figure 3.16 also shows the delay components for contention slot resolution in the case of collisions. In a nutshell, the headend has the complete information of the distance of each station, the amount of allocated grant, and the processing time. So the headend schedules the contention and data slots accordingly.

3.2.2 Continuous Mode

In this case, contention minislots can be placed anywhere in the upstream frame [13, 52], and hence possibly it would require less frequent regrouping of idle time slots in a frame. This frame structure requires two instances of the CRA to be run in parallel - one over odd numbered frames and other over even numbered frames (section 3.13). The advantage of this scheme, is increased contention minislots utilization, but the disadvantage is increased complexity and access delay (see section 3.1.3). The frame duration T_F , for this cases is lower-bounded to follows (see figure 3.17)

$$T_F \geq 2t_p + t_{HC} + t_s \quad (3.10)$$

Figure 3.17: Frame Structure with distributed minislots

where t_p is the propagation delay and t_{HC} and t_s are the headend and station processing delay.

3.3 Contention Slot Allocator

Contention slot allocation (CSA) is an algorithm operating at the LMDS hub (or headend) that specifies the type of minislots in the upstream channel [12, 38]. Recall that the upstream channel consists of contention minislots and data minislots. Contention minislots are used for transmitting bandwidth allocation requests in the upstream, and data minislots are concatenated to transmit actual data in the form of ATM cells or Variable length PDUs.

The performance of the MAC protocol is strongly affected by the number of slots opened for contention purposes (contention minislots). Since the capacity allocated to the contention channel is control messaging and not information, therefore this capacity has to be minimized and used as efficiently as possible. The capacity of the contention channel is solely guided by the traffic type and average request size. Average request size is defined as average size of bandwidth allocation requested per successful request transmission.

The simplest possible case is to allocate a fixed capacity to the contention channel, but this approach cannot take into account the sudden traffic load variation. A better way to handle this variation is to change dynamically the capacity of contention channel depending upon load condition.

Figure 3.18 shows a simple approach for contention slot allocation algorithm. As we can see, the inputs to this algorithm are the number of collisions occurred in the previous frame, outstanding bandwidth allocation, average request size (estimate), new arrivals (estimate). The algorithm shown here is suitable for framed implementation with contention slots clustered in the beginning of the frame, although on similar guidelines contention slot algorithm can be made for distributed contention slots over a frame. The

implemented CSA has some dependencies upon collision resolution process un-blocking ternary tree), but it can also be adapted for any other CRA (for e.g p-persistence).

Figure 3.18: Contention Slot Allocation

3.3.1 Data Slot Priority Algorithm

In this algorithm, the headend calculates the number of reserve data slots it requires in the current frame from its request queue. It marks the required slots as data slots (or data minislots), and marks the rest of the slots as contention minislots. The contention minislots are then marked with RQ number specifying whether they are for new arrivals or for retransmissions.

$$C_t(i) = C_r(i) + C_n(i) \quad (3.11)$$

where $C_t(i)$ is the total number of contention minislots required in the i^{th} frame, $C_r(i)$ is the number of contention minislots required for retransmission in the i^{th} frame, $C_n(i)$ is the number of contention minislots required for new arrivals in the i^{th} frame. The calculation

of $C_T(i)$ is straightforward, as it represents the current status of tree (or collision stack), accounting for backlog allocation of contention minislots to be done and any new collision occurred in the previous frame. The calculation of $C_n(i)$ is more difficult. It can be estimated by means of the Rivest/Zenith algorithm [57, 56, 14, 15]. It depends upon the number of contention slots opened for new users in the previous frame and the number of collisions/success/idle occurred in them (see equation 3.3). But it is not always possible to fulfil the complete requirement in a single frame, that is allocation of all the required data and contention slots (retransmissions and new openings). Therefore, a trade off has to be done. One possible way is by setting a threshold range for request queue. In this case we keep a pre-calculated number of minislots open for contention; any excess reservations are to be put in a data queue. If the data queue is within a threshold range, the number of contention minislots can be kept at the pre-calculated number. If the data queue is over a limit, contention minislots will be reduced from the steady state number and converted into data slots to block out new requests and allow more data to be transmitted. If a data queue falls back into threshold range, the number of minislots will increase to the steady state value. This algorithm provides priority to allocation of data slots. If we set the threshold value for data queue to 0, implying that the requirement of a single data slot is fulfilled first, then the resulting algorithm becomes the *absolute data slot priority algorithm*. This algorithm provides a near optimum performance for systems with $RTD < 1$ slot (or 0 delay systems). But if we consider a system with a finite delay (for e.g processing delay, large round trip delays), then the headend has to provide estimation (or forecast) for the next two frames. In that case the more generalized version of data slot priority algorithm becomes effective. All other factors like, estimation of population, and number of expected successful requests in a frame, estimate of average request size from grant queue, and backlog of grant queue have to be considered in that case to optimize the performance.

The *absolute data slot priority algorithm* is a self regulating algorithm because when the number of contention minislots in a frame is too low, the bandwidth requests will not get to the headend and the requirement of data slots will be reduced, which implies that more than necessary contention slots will be provided in the next frame. If the number of contention minislots is too high, more successful requests will reach the head end and the number of slots that will be allocated as contention minislots will decrease.

3.3.2 Contention Slot Priority Algorithm

This algorithm was an experimental algorithm we studied, in which the headend calculates the number of contention slots it requires for retransmissions (with $RQ > 1$). As explained earlier in the ternary tree each collision results to splitting in three slots. Based upon this calculation the LMDS hub estimates the number of contention slots required for retransmission and allocates them. Then the headend calculates its current data slots requirement, and allocates them, and if still some slots are left idle, they are converted to contention slots for new users ($RQ=0$). As we have given priority to contention slots for retransmissions, this algorithm was named *Contention slot priority*.

In both Data slot priority algorithm and Contention Slot priority algorithm, the piggybacking capability of IEEE 802.14 MAC and MCNS MAC will change the situation to some extent. Piggybacking allows an active user to request bandwidth without actually contending, which implies that a fewer contention minislots are required. The extent of piggybacking depends upon traffic type, active user population and congestion level, but certainly some estimates can be done for a particular traffic type, user population and load conditions. This can affect the calculation of the required number of contention minislots in both the algorithms.

Candidate MAC Protocols

The MAC protocol is responsible for sharing scarce resources among users. The classical solution has been provided by allocating fixed division of resources among users. The access systems evolved with these ideas are known as TDMA, and FDMA. A fixed division of resources becomes inefficient when the requirement of users vary with time. The inefficiency of the above scheme has been addressed by demand assignment of resources. The scheme with demand-assignment also tends to become inefficient when resources are shared among a large number of users who do not require it most of the time. The other solution which has been suggested is to employ some form of random-access, i.e., allowing a user to seize the entire communication resource when he happens to have information to transmit. On similar guidelines, various protocols based upon pure random access (ALOHA), or combination of reservation and random-access (PRMA) [24] have been proposed for wireless channels, and PODA for satellite channels [39].

In the context of LMDS, we have been interested in a suitable MAC which should not only be able to share the resources (bandwidth) effectively among large number of users, but also have the capability to provide quality of service (QoS) guarantees for this multi-service network. The most obvious choice for us was to look for a protocol proposed specifically for LMDS systems like DAVIC 1.2 recommendation. But we have also considered protocols proposed for HFC networks, which provide a similar environment and

services to users. When this project started, three major MAC protocols were evolving from different standard bodies, which we considered were suitable for LMDS networks.

- HFC specific - IEEE 802.14 MAC draft proposal. [27, 28, 30]
- LMDS specific -DAVIC MAC recommendation. [41]
- HFC specific - MCNS MAC standard. [47]

A brief description of each MAC protocol is presented in this chapter.

4.1 IEEE 802.14 MAC CONVERGENCE DRAFT PROPOSAL

The main features of the MAC protocol currently being specified by the IEEE 802.14 group are; its ability to support ATM cell transfer and variable-length packet transfer while maintaining the ability to provide high quality of service (QoS). The upstream channel is divided in time into basic units called *minislots*. The minislot is the smallest unit of transmission and there are several types of them. Their functions are defined by the headend and conveyed to each station by means of *downstream control messages*. Several minislots can be concatenated in order to form a single data PDU (protocol data unit), ATM, or variable-length (VL) fragment. The protocol does not specify any frame structure. An upstream channel can be viewed as a stream of minislots. Similar to upstream, the downstream channel is divided into a stream of 6 bytes long allocation units which can be concatenated to make ATM cells or variable-length packets. Downstream is used to provide feedback and grant to the request made by stations along with data. The contention resolution is based on a ternary-tree algorithm with variable new packet entry persistence. This protocol completely specifies the behavior of station while most of the headend algorithms for computing bandwidth allocation, and granting requests are left as implementation spe-

cific. The MAC protocol provides specification to the following issues which are discussed below.

- **Station Addressing:** 48 bit MAC address including 14 bit local identifier.
- **Upstream bandwidth control format:** request minislots types and structure.
- **Upstream PDU formats:** ATM and VL PDUs.
- **Downstream formats:** Data Grant/Contention Feedback/Contention Slot Allocation/ other management.
- **MAC operation:** Registration/Ranging and Power leveling/ time synchronization/bandwidth request/collision resolution/handling priorities/ error recovery mechanisms/data transmission operation.

4.1.1 Architecture

The IEEE 802.14 has defined a layered architecture for cable modem, and the headend controller. A reference model is shown in the figure 4.1 which contains: the Physical layer (PHY), the MAC layer, and Upper layers. The PHY layer is broken into two sublayers, Transmission Convergence (TC) sublayer and Physical Media Dependent (PMD) sublayers. UP PMD and UP TC corresponds to upstream (station to Headend Controller), and DN PMD and DN TC corresponds to downstream (Headend Controller to station).

Figure 4.1: Layer Model

4.1.2 ATM Interface

The IEEE 802.14 specification provides a complete support for native Asynchronous Transfer Mode (ATM) layer services (ITU-T.I.150). The protocol specifies the transport of ATM cells across the MAC. It also supports all ATM service classes with their associated quality of services parameters. As shown in figure 4.2, 802.14 MAC layer is considered to be a part of physical layer in the ATM layer architecture. In the context of ATM architecture, an 802.14 station is an element of an ATM end system or an ATM node.

Figure 4.2: Relationship between 802.14 and ATM

4.1.2.1 ATM Traffic Management

802.14 MAC layer supports the ATM service categories, traffic contract parameters and quality of service. MAC dynamically allocates resources among all the users of the MAC layer service, (including ATM VCs). Besides ATM layer, bandwidth scheduling is also performed at MAC layer.

4.1.3 Downstream Channel

- The Downstream Channel is comprised of a series of 6 octet allocation units, which may contain Idle PDU's or they may be concatenated to form ATM cell PDU's, TC PDU's and Variable Length Fragments.
- ATM cells are used to transmit user information and MAC layer control messages, which are encapsulated in AAL5 (ATM Adaptation Layer Type 5).
- The Stream of octets is encapsulated in an MPEG-2 transport stream for multiplexing and transmission to network which is demultiplexed by stations to extract the stream of octets.
- The 802.14 MPEG-2 transport stream can be multiplexed with other types of traffic (e.g MPEG packets from other networks), provided 802.14 MPEG packets are multiplexed at constant rate (e.g every Nth packet) with other MPEG packets.

Figure 4.3: Downstream Channel Hierarchy

4.1.3.1 Downstream Bandwidth Management Cells

Grant Information Elements: It is used for allocating upstream bandwidth to a station. Various types of grants correspond to ATM Cell, Variable Length Fragment, and User Defined format.

Allocation Information Element: It provides information about the allocation of upstream bandwidth per upstream carrier e.g *Request Minislot Allocation elements* specify the location of minislots available for contention as well as parameters about these minislots which aid in the collision resolution algorithm.

Figure 4.4: Request Minislot Allocation Elements

Feedback Information Element: It provides feedback for all types of bandwidth requests (piggyback, contention and CBR).

4.1.4 Upstream Channel

In the Upstream direction, communication path is shared by many users, all transmitting data to the headend. The upstream channel time is partitioned into a sequence of fixed-size time slots (minislots). The duration of one minislot is equal to the time required to transmit 8 octets of data plus the time required to transmit PHY overhead and the guard band. A MAC layer Protocol Data Unit (PDU) that occupies a single minislot, is referred to as a mini PDU. Mini PDUs are primarily used to provide contention opportunities for users to request bandwidth; actual data transmission is done in reserved data PDU's (ATM Cells or Variable Length Fragments), which are made by concatenating several minislots as shown in figure 4.5. The number of mini slots required to transmit an ATM cell, PDU and Variable Length PDU (VL) depend on the PHY overhead and guard time. However, an integral number of mini slots are allocated by the headend and used by the station to transmit ATM/VL PDUs. The upstream channel is a multiple access shared medium in which one

or more stations share the upstream bandwidth, which is dynamically allocated by the Head End. Upstream Channel Bandwidth is acquired by using a Random Access scheme based upon contention. A Contention Resolution Algorithm based upon *the ternary tree with variable entry persistence* is used to resolve collisions. Additional bandwidth is obtained by piggybacking the bandwidth request on data PDU's, which can avoid the necessity of contention for bandwidth request.

Figure 4.5: Concatenation of Minislots

4.1.4.1 Upstream Formats

Request Mini Slot PDU (RPDU): RPDUs are used by the station to request bandwidth (by contention) in the shared upstream medium on behalf of a logical queue within that station.

RTY: 0x00 - ATM PDU; 0x01 - Variable length fragment; 0x02 - Network Defined MiniPDUs

Figure 4.6: Request Minislot Protocol Data Unit (RPDU)

Upstream ATM Cell PDU (APDU): APDUs are used to transmit ATM cells to Headend by a station. The structure of the APDU is shown in figure 4.7. The NNI cell format has been used (no GFC field).

Figure 4.7: Upstream ATM Cell Format

Upstream Variable Length Packet Fragment (VL PDU): Variable length Packet support is used to allow a more efficient carriage of LLC traffic and native protocols which generate variable length data units (e.g., 802.3, IP, IPX, IPv6). (see section 5.5)

Downstream Format is similar to upstream format except no EBR field in downstream and DLI field instead of SLI

Figure 4.8: Upstream Variable Length Packet

Table 4.1: Field Description

Field	Usage	Field	Usage
RTY	Request Type	VCI	Virtual channel identifier
QID	Queue Identifier	PT	Payload type indicator
LEN	Size of allocation requested	CLP	Cell loss priority
CRC	CCITT-CRC16	HEC	Header error check
PDT	PDU Type	FFB	First Fragment Bounder
EK	Encryption key even/odd identifier	NFS	Number of Frame Starts
EBR	Extended bandwidth request field	Payload	ATM or Fragment payload
SLI	Source station local ID		

4.1.5 MAC Operation

4.1.5.1 Registration

In the beginning, the MAC allocates bandwidth in both upstream and downstream directions for registration and ranging of stations. This information is passed by the headend controller (HC) PHY/TC by MA-ALLOCATE.indication primitive. Thereafter, HC MAC

generates idle cells in downstream direction to allow PHY/TC to perform registration and ranging. The registration process begins by TC_CONNECT.request from station MAC to PHY. The station TC performs the channel acquisition, ranging, power levelling and return TC_CONNECT.indication to MAC. Once the station is properly ranged, HC assigns a station MAC address, Local ID and other upstream parameters. Finally, a CONNECT message is transmitted by HC MAC to the station MAC as an indication for connection establishment. The detail of registration message flow is explained in figure 4.9.

Figure 4.9: Registration Message Flow

4.1.5.2 Bandwidth Request

The Headend Controller (HC) is the central entity which supervises all the transmissions from users (stations) and resource allocation in the upstream channel. A station requests bandwidth on behalf of the logical queue it supports, by providing the outstanding payload information associated with the logical queue to the HC. Based on quality of service (QoS) criteria, the HC then decides for each user's logical queue when and how to transmit its data. A station has various means of requesting bandwidth from HC. Initially, when a station becomes active, it requests bandwidth through contention based transmissions on the upstream channel. Once an initial request is accepted and requested allocation is obtained, additional bandwidth can be requested by means of concurrent request or by use of an *Extended Bandwidth Request* (EBR) field in a data PDU. The method of concurrent request, which permits multiple requests associated with a single logical queue to be sequentially accepted by and queued at the HC, this is particularly useful when contention access delay is low. The method of extended bandwidth request, commonly known as "piggybacking", is particularly useful when contention access delay is high. A constant bit rate (CBR) permanent allocation can be requested by a station for a logical queue, such that periodic grants at a desired frequency are allocated until the station sends a CBR allocation release message for the logical queue.

Request Minislots (RMSs) are allocated in the upstream channel by the HC to the users for contention access, as they receive RMS Grant messages in the downstream channel. An RMS Grant message identifies a number of RMSs divided into groups for different distinct sets of users which are at various stages of contention resolution. Typically, the HC allocates more than one RMS to each user, and an RMS may be allocated to multiple users. To reserve for transmission resources in the upstream channel, a user randomly selects an RMS from the group of RMSs available to the user, and then attempts to send a request message in the selected RMS. The request message referred to as a Request

MiniPDU (RPDU), identifies the user and the size of its requested allocation. Since the user does not necessarily have exclusive access to the RMS, collisions can occur. When there is a collision, a contention resolution algorithm is used to resolve the collision. The contention resolution algorithm specified for this standard is based on a tree splitting algorithm.

4.1.5.3 Contention Resolution

To initiate its transmission and acquire upstream bandwidth for data transmission, a station attempts to send a request (RPDU) in a designated upstream request mini slot (RMS). Since multiple users may attempt to transmit their request in the same request mini slot, a collision is possible. Collisions are resolved using “*ternary tree contention resolution algorithm with variable entry persistence for new comers*”.

Each collision and request mini slot is identified by a Request Queue number (RQ). Request Queue number is a parameter used in a contention resolution algorithm to identify users which are eligible to contend in a particular group of request mini slots. A RQ of value 0 is reserved for “*newcomers*”. Newcomers are users entering the contention process due to the generation of new upstream bandwidth request.

The contention resolution process operates in the following manner. Initially, newcomers (with $RQ=0$) transmit their bandwidth request in the group of request mini slots designated by headend and identified as request mini slot group with $RQ=0$ by selecting one request mini slot at random. The Number of mini slots in a group are identified by the parameter “NMS”. NMS together with the Range Parameter “R”, defines the probability of a newcomer to enter the contention process. R provides the maximum range for generating a random number by each user. For example, if a particular request minislots group

is associated with $NMS = 4$, and $R=8$, then every user has 50 percent probability to select a request minislot. The value of R is changed dynamically by headend.

Having allowed newcomers to enter the system, the headend identifies each request mini slot in a COLLISION state, and subsequently assigns to all the users that experienced a collision in each request minislot a unique RQ number greater than zero ($RQ>0$). The assignment of RQ number is done in such a way that backlogged users associated with the largest RQ are allowed to transmit first. For each collision associated with a particular RQ number, the headend allocates a group of three contention minislots in an appropriate time in the future. Users with a particular RQ number now transmit their request in a group of three request mini slots associated with the same RQ number by selecting one request minislot at random. In general, the larger the RQ assigned to a user, the earlier it would receive allocation of group of request mini slots. Hence, new arrivals with $RQ=0$ are always served last. Each RQ assigned to a request minislot group is unique at a given time. To avoid deadlock, a user waiting contention resolution (after encountering a collision) is allowed to use the first RMS group with a RQ of a value less than or equal to the RQ that has been assigned to the user. The process of contention resolution is explained with the help of the following example.

4.1.5.3.1 Example

Figure 4.10 shows a simple example of the operation of contention resolution algorithm employed. It shows four different clusters of request minislot allocated by HC. Each request minislot is identified by its RQ number and position. For example 0:1 indicates a RMS with $RQ=0$, and position = 1. The alphabets X,Y,Z,V etc. written in each RMS represents a group of stations who select that RMS transmit their request. The number of stations in each group could be 0, 1 or >1 , which implies Idle, Success, and Collision in a RMS respectively. In cluster (j-2), number Z and X are more than 1, represented by C

(Collision), and Y, W and V are either 0 or 1 and represented by NC (No collision). HC allocates $NMS = 5$ (No of minislots) in the cluster (j-2) with $RQ=0$ with Range $R = 8$. New stations randomly pick a number from 1 to 8.

Stations who select a number from 1 to 5 transmit in their respective positions but users who select number 6,7 or 8 are not allowed to transmit. So R/NMS defines the probability of entering (entry persistence - see section 3.1.2.9). RPDUs (if any) transmitted in RMSs (0:2), (0:4) and (0:5) are successfully transmitted and these RPDUs exit the contention resolution process. On the other hand, RPDUs that were transmitted in RMSs (0:1) and (0:3) experience a collision. Hence, their stations have to go through the steps of the contention resolution algorithm until the collided RPDUs are transmitted. The HC assigns $RQ=2$ to Z-RPDUs that have collided in (0:1), and $RQ=1$ to X-RPDUs that have collided in RMS(0:3). Following the collision, each of the Z-(resp. X-) stations that transmitted in RMS (0:1) (resp. (0:3) of cluster (j-2) chooses randomly (a process usually referred to as splitting) any of the first three RMSs of cluster (j-1) and schedules its next RPDUs transmission attempt in the selected RMS, as shown in step (2) of Figure 4.10. We denote with Z_a, Z_b and Z_c the number of the Z-RPDUs that will be transmitted in RMSs (2:1), (2:2) and (2:3), respectively, of cluster (j-1). Similarly, we denote with X_a, X_b and X_c the number of the corresponding RPDUs that result from the splitting of the X-RPDUs.

According to the ternary tree algorithm, the splitting of the collided RPDUs is done positionally in groups of three, hence, the name is ternary tree algorithm. Since the Z-collision is the first (or, leftmost) to occur in cluster (j-2), the corresponding Z-RPDUs split into the first three RMSs of cluster (j-1), see step (2) in Figure 4.10. Since the X-collision is the second to occur in cluster (j-2), the corresponding X-RPDUs split in the following three RMSs of cluster (j-1), thus avoiding any RMSs to be used by the split Z-RPDUs, see step (3) in Figure 4.10. Note though that there is only one RMS, RMS (1:1),

remaining in cluster (j-1) for the split X-RPDUs. RMS (1:1) will be occupied by the Xa-RPDUs.

Figure 4.10: Ternary Tree Stack Visualization

To accommodate the Xb- and Xc-RPDUs, an imaginary stack is introduced which is logically attached to the right of the cluster. The Xb- and Xc-RPDU will “move” into the lowest two levels of this stack and are next-in-line to be transmitted when RMSs become available in future clusters. If the number of RMSs in a cluster is not sufficient to accommodate the needs of collided RPDUs split from a previous cluster, then the RPDUs already residing in the stack “move-up” the stack to make space for the latter RPDUs, see step (7) in Figure 4.10. If there are more RMSs in a cluster than are necessary to meet the needs of any “splitting” RPDUs following their collision, then the RPDUs in the stack “move-down” the stack and toward the (transmission permitting) RMSs, see steps (8), (9) and (10) in Figure 4.10. By splitting the following collisions and moving “up” and “down” the stack according to the ternary tree algorithm rules, collided RPDUs eventually resolve their collisions and are successfully transmitted.

4.1.6 Contention State Machines

A user maintains a contention state machine for each concurrent contention based transmission attempt for its each established logical queue. A contention state machine for a particular Contention Interleave (CIL), consists of six states as shown in figure 4.11 and defined in table 4.2. In this contention state diagram, solid black arrows identify the sequence of steps taken normally by a user to resolve RPDUs it experiences. Hollow gray arrows identify the steps taken by a station from possible abnormal conditions encountered during an RPDUs contention resolution. Finally, black hollow arrows identify the steps taken by a station following a successful contention-based RPDUs transmission. In addition, the unique steps followed by a newcomer are highlighted in figure 4.11 with a gray background. A user moves from one state to the next, and a set of operation takes place that facilitates the contention resolution. A station whose RPDUs transmission experience a collision stores the RQ value assigned with a storage variable RQs.

Figure 4.11: Contention State Machine

Table 4.2: Contention States

State	State Definition
<i>Contention_Idle</i>	The standby state of the contention state machine. In this state, user is in idle state.
<i>Wait_For_RMS_Grant_New Comer</i>	The initial state of the contention state machine. In this state, a user become active and ready to enter contention access. Users wait for the assignment (or identification) by the HC of an "entry" RMS group, i.e., an RMS group with an RQ=0 value assigned to them.
<i>Wait_For_RMS</i>	The state in which a user awaits the arrival of a particular RMS in which it has been granted permission to send an RPDU (request).
<i>Wait_For_RMS_Feedback</i>	The state in which a user awaits contention feedback from the HC.
<i>Wait_For_RMS_Grant</i>	The state where user wait for assignment (or identification) by HC of the RMS group that stations, whose RPDU collisions was assigned a particular RQ value, can use for their RPDU retransmission
<i>Transmission_Ready</i>	The state in which a user has its request accepted after a successful contention, and the user is ready for data transmission. In this state, a concurrent request may be initiated for contention access.

4.1.7 Transmission State Machine

Following a successful contention-based RPDU transmission, a user would transmit actual payload according to the procedure explained by transmission state machine in figure 4.12 and states are defined in table 4.3. A user in the *Wait_For_DataMiniSlot_Grant* state is waiting for a grant information element to be sent by the HC (ATM or VL PDU grant). After receiving the grant, the user waits for the designated upstream minislots to transmit. Also depending on whether new payload PDUs become available at the station since the last time it registers payload PDUs with the HC, the station can also piggyback in its upcoming payload PDU transmission by an extended bandwidth (EBR) message. Following the payload PDU transmission the station decrements accordingly its payload amount variable and eventually returns back to *Wait_for_DMS_Grant* state or become idle depending whether its queue is empty or not. If an EBR was sent by a station, the station first moves to *Wait_for_EBR_Feedback* for an acknowledgment by the HC that it has received the stations's EBR message. When the acknowledgment is received, the station increments again the payload amount variable accordingly.

Figure 4.12: Transmission State Machine

In case of abnormal situations like not receiving the EBR acknowledgment in time, or timer expiration while waiting for an event to happen, the active station goes to transmission idle state and makes a new bandwidth request via a contention based RPDU transmission.

Table 4.3: Transmission States

STATE	DEFINITION
<i>Transmission_Idle</i>	The standby state of the transmission state machine.
<i>Wait_For_DataMiniSlot_Grant</i>	The state in which a user with an accepted request awaits a DMS grant that assigns an opportunity for the user to send its data payload for which the accepted request has successfully made bandwidth reservation.
<i>Wait_For_DataMiniSlot</i>	The state in which user awaits the arrival of a particular DMS in which it has been granted permission to send a data PDU.
<i>Wait_For_EBR_Feedback</i>	The state in which a user who has an outstanding extended bandwidth request awaits feedback from the HC regarding the status of the request.

4.2 MCNS MAC Standard

MCNS (Multimedia Cable Network Systems) Data Over Cable Interface Specification (DOCSIS) recommendation (recently also ITU (J.122) Standard) is the packet based IP protocol for RF transport between CMTS (Cable Modem Termination System) located at the Headend and CM (cable modem) at customer premises (specified for HFC networks)

Figure 4.13: IP Traffic Flow

4.2.1 MAC Highlights

- Transparent bi-directional transfer of IP (version 4 with migration to version 6) traffic between CMTS and CM.
- Capability of ATM Support
- A stream of minislots in upstream.
- Support of MPEG packets in the downstream, multiplexing video and data.
- Dynamic mix of contention based and reservation based upstream transmit opportunities. (polling of individual users can also be implemented)
- Bandwidth efficiency through support of variable length packets.
- Class of service support (Inelastic: Delay intolerant, delay tolerant, Elastic).
- Support of a wide range of data rates.
- Programmable RS FEC, preamble and guardband lengths.

Upstream transmission time is divided into equal time segments (*minislots*). A minislot is the granularity unit for upstream transmission opportunities. The use of each and every minislot is controlled by the Cable Modem Termination System (CMTS) at the headend. The CMTS assigns contiguous intervals of minislots to individuals CMs, or make them available for contention to a group of CMs, or open them for contention to all CMs.

4.2.2 MAC Frame Formats

The MAC frame is defined as the basic unit of information transfer between MAC sublayer at the CMTS and CM. The same basic structure is used in both upstream and downstream directions. MAC frames are variable in length. This term frame is not the same as the term “framing” that indicates some fixed time (periodic) relationship. Each frame is not shared among CMs but is defined for a particular CM or CMTS. Figure 4.14 shows a MAC frame; preceding with either a Physical Media Dependent (PMD) sublayer overhead

for upstream or a MPEG transmission convergence header in the downstream. The first part of the MAC frame is the MAC header that uniquely identifies the contents of the MAC frame. Following the header is an optional data PDU region. The format of data PDU and whether it is even present is described in the MAC header.

Figure 4.14: Generic MAC Frame Format

Figure 4.15 shows the transportation of MAC frame by the PMD sublayer in the upstream channel.

Figure 4.15: Upstream MAC/PMD Convergence

4.2.3 MAC Header Format

Figure 4.16 shows the general format of MAC header. The *frame control* FC field is used to identify the rest of contents of MAC header. The FC field is followed by 3 bytes of MAC control, an optional *extended header field* (EHDR); plus a *Header Check sequence* (HCS).

Table 4.4: Generic MAC Header Format

MAC Header Field	Usage	Size
FC	Frame Control: Identifies type of MAC Header	8 bits
FC_TYPE	Packet PDU, ATM PDU, Reserved PDU or specific MAC Header	2 bits
FC_PARAM	Parameter bits, use dependent on FC_TYPE	5 bits
EHDR_ON	Flag to indicate whether EHDR is present or not/	1 bits
MAC_PARM	Parameter field whose use is dependent on FC: if EHDR_ON=1; used for EHDR field length (ELEN) else if for concatenated frames used for MAC frame count else (for Requests only) indicates the number of minislots and/or ATM cells requested	8 bits
LEN (SID)	Length of the MAC frame - The length is defined to be the sum of the number of bytes in the extended header (if present) and the number of bytes following the HCS field. (For a REQ Header, this field is the Service ID instead)	16 bits
EHDR	Extended MAC Header (where present; variable size).	0-240 bytes
HCS	MAC Header Check Sequence	2 Bytes
	Length of a MAC Header	6 Bytes + EHDR

Figure 4.16: MAC Header Format

4.2.4 Variable-Length Packets

The MAC layer supports variable-length Ethernet (ISO-8802-3) type Packet Data PDUs. A unique MAC Header is appended at the beginning of each packet as shown in figure 4.17.

Figure 4.17: Ethernet/802.3 Packet PDU

4.2.5 ATM Cell MAC Frames

The MAC protocol also provides support of ATM transportation as shown in figure 4.18.

Figure 4.18: ATM Cell MAC Frame Format

4.2.6 MAC Management Header

Various MAC management messages are used for different functions as explained in table 4.5. The part of bandwidth allocation (MAP) message is shown in figure 4.20. The details of bandwidth allocation procedure are explained in section 4.2.9

Figure 4.19: Management MAC Header

Figure 4.20: MAP Format

Table 4.5: MAC Management Messages

Message Type	Comments
Timing Synchronization	CMTS→CM, periodic
Upstream Channel Descriptor	CMTS→CM, periodic, define minislot size, symbol rate, frequency, preamble, FEC, guard time etc.
Upstream Bandwidth Allocation (MAP)	Bandwidth Allocation start time, Ranging and contention Back off window, allocation type (short grant, long grant, data ack, maintenance etc.)
Ranging Request	CM→CMTS, transmit at initialization and periodically on request from CMTS to determine network delay
Ranging Response	CMTS→CM, Response to above message, timing, frequency change/adjustments
Registration Request	CM→CMTS, transmit at initialization
Registration Response	CMTS→CM, response of above message, service ID/service class assignment
Upstream Channel Change Request	CMTS→CM, Request by CMTS to change upstream channel
Upstream Channel Change response	CM→CMTS, Response to above message

4.2.7 Request MAC Header

The Request MAC Header is used by a cable modem to request bandwidth in the upstream channel. Each request is made for unique service queue (SID) for a given station.

Figure 4.21: Request MAC Header Format

4.2.8 Extended MAC Headers

Provision of an Extended MAC Header is provided with certain MAC Frames. The most important use of Extended MAC header from a multiple access point of view, is to request bandwidth by piggybacking on a CMTS acknowledgment; other uses include upstream/downstream security and privacy functions.

Figure 4.22: Extended MAC Format

Table 4.6: Important Extended Header Element Formats

EH_TYPE	EH_LEN	EH_VALUE
1	3 bytes	Request: minislots requested (1 byte), SID (2 bytes) [CM-->CMTS]
2	2 bytes	Acknowledgment requested; SID (2 bytes) [CM-->CMTS]

4.2.9 Upstream Bandwidth Allocation

Upstream bandwidth allocation is completely controlled by CMTS. Upstream allocation is done by transmitting an allocation map in the downstream by CMTS. An allocation map (section 4.2.6) is a MAC management message describing the use of some time interval in the future. This includes the use of all the upstream minislots in that interval. For example, a map may describe some slots as grants for a particular stations to transmit data and some other slots as contention slots. The basic mechanism for assigning bandwidth is shown in figure 4.23.

Figure 4.23: Allocation Map

4.2.10 Contention Resolution

This protocol specifies a *truncated binary exponential back off* contention resolution with initial back off window and maximum back off window controlled by CMTS. The algorithm can be set as distributed ethernet style back off by using a fixed initial back off to zero and maximum back off to ten in the map. On the other end, CMTS can choose to set up initial and maximum back off values to be identical, and update these values frequently

in the map so that all CMs are using the same and optimal back off window. Hence, the collision resolution algorithm becomes more centralized.

4.3 DAVIC 1.2 MAC Recommendation

The MAC specified in DAVIC 1.2 recommendation document describes a bi-directional flow between modems located in Air Interface Units (AIU) at headend access node and a Network Interface Unit (NIU) at the customers premises. This MAC is another example of centralized control protocol, where AIU controls all the transmission and retransmissions of supported NIUs. A brief summary of this protocol is as follows.

The specified MAC protocol essentially supports a framed structure in both upstream and downstream direction with a constant frame size between 3 to 6 ms. The MAC protocol is based upon ATM (Asynchronous Transfer Mode) transportation in both upstream and downstream with a provision of MPEG-2 TS (Motion Picture Enhanced Group Transport Stream) in downstream. In case of ATM transport, upstream and downstream frames are divided into time slots that encapsulate one ATM cell. In case of MPEG2-TS, downstream frame is divided into time slots that can encapsulate one MPEG2-TS packet. Upstream frame is divided into time slots that can encapsulate one ATM cell. In both cases, a fixed relationship between the number of upstream and downstream time slots is specified.

4.3.1 Downstream Frame Structure

Downstream frame structures are Time Division Multiplex with two types of time slots: *Frame Start Time Slot* (indicated by FS) and *Random Access Time Slots* (indicated by NIU). *Frame Start time slots* are present in the beginning of every downstream frame to

provide synchronization with upstream frame. *Random Access Time Slots* are used to unicast, broadcast, or multicast information and management cells to NIUs.

Figure 4.24: A possible scenario of Downstream Frame Structure

4.3.2. Upstream Frame Structure

The upstream channel is divided into fixed length TDMA frames (3 to 6 ms). Each TDMA frame is subdivided into Poll Time Slots (indicated by P), Contention Time Slots (indicated by C), and Reserved Time Slots (indicated by NIU). The length of all three types of time slots is 68 bytes to encapsulate an ATM cell as shown in figure 4.25. Allocation of time slots is controlled by AIU. Allocation of reserved time slots is done to a particular NIU after the request is made to transmit upstream payload, but allocation decision of contention and poll time slots are solely decided by AIU without any request from NIU's. Moreover, contention and poll time slots can be allocated to one or many NIU. As evident from their names, contention slots are used to request bandwidth in upstream by NIU, and Polling Slots are used by NIU to send a poll response after receiving a poll request from AIU. Since multiple request can be transmitted in a contention slot by various NIUs, collision occurs which can be resolved by AIU using well known Contention resolution algorithm (CRA).

Figure 4.25: Upstream Time Slot Structure

4.3.3 MAC Operation

Table 4.7: Standard MAC Messaging

MAC_message_type	Slot Type	Delivery	Description
reservation_request	[C]	AIU $\leftarrow$ NIU	NIU request for reserved time slots.
reserved_slot_allocation	[FS,NIU]	AIU $\rightarrow$ NIU	Allocation/deallocation of reserved time slots.
contention_slot_allocation	[FS,NIU]	AIU $\rightarrow$ NIU	Allocation/deallocation of contention time slots.
poll_slot_allocation	[FS,NIU]	AIU $\rightarrow$ NIU	Allocation/deallocation of poll time slots.
sign_on_request	[FS]	AIU $\rightarrow$ NIU	Parameters for an NIU to enter the network.
tx_calibration	[FS,NIU]	AIU $\rightarrow$ NIU	NIU transmit freq, power, time corrections.
status_request	[FS,NIU]	AIU $\rightarrow$ NIU	Request specified NIU status items.
status_response	[P]	AIU $\leftarrow$ NIU	Requested NIU status items.
mac_session_request	[C]	AIU $\leftarrow$ NIU	NIU request a session/control connection.
mac_session_connection	[FS,NIU]	AIU $\rightarrow$ NIU	AIU assigns VPI/VCI for session connection.
mac_session_release	[FS,NIU]	AIU $\leftrightarrow$ NIU	AIU or NIU releases a session connection.
mac_session_msg_ack	[FS, NIU,C,P]	AIU $\leftrightarrow$ NIU	Acknowledgment of receipt of the 3 above messages.
new_downstream_channel	[FS,NIU]	AIU $\rightarrow$ NIU	Command NIU to enter net on new channel.
contention_slot_feedback	[FS]	AIU $\rightarrow$ NIU	Contention slot stabilization feedback.

To enter the network, each NIU acquires a downstream frequency channel and listens to the poll directed to it by AIU, which transmits every 2 seconds on one of the frequency channels for each NIU. If NIU does not receive a poll request on one frequency channel within 2s, it tunes to a different channel and listens again. After receiving the poll request message, NIU performs calibration by transmitting in the allocated polling slots. Once the NIU is calibrated, AIU sends a mac_session message to NIU and connection is established; time slot allocation in upstream is performed by AIU as depicted in figure 4.26. Each user requests for bandwidth by sending a reservation_request message in contention slot, and receives a contention_feedback (collision, idle and success), and reservation_slot_allocation, thereafter. In case of collisions, AIU resolves the collision by using a well known Contention Resolution Algorithm. Messages are defined for allocation/deallocation of one or more contention slots to single or many users with a retrans-

mission probability. The protocol also provides flexibility in designing more messages, if required. Hence, any CRA explained in section 3.1 can be used (e.g p-persistence, tree algorithm). Besides contention polling response messages can also be used to request bandwidth.

Figure 4.26: Network Entry and Upstream Time Slot Allocation

4.4 Comparison

4.4.1 Major Similarities Between the Candidate Protocols

This section has been taken from [45]. All three protocols (DAVIC, 802.14 and MCNS) are capable of active users physical parameters calibration (Power, Range, Frequency control etc.) and can assign users to frequency channels dynamically (for load balancing). Each protocol uses encryption (and decryption) of individual data flows to ensure privacy in the broadcast environment. All three protocols employ a request/grant procedure for bandwidth allocation. Another common characteristic is the provision of demand assignment bandwidth allocation and the use of random access for registration and bandwidth request. For traffic regulation and priority management, each MAC protocol can assign a subset of contention slots to a group of users. When operated in the ATM mode, each protocol can effectively support all ATM service classes with the capability of temporary and permanent bandwidth allocation (MCNS also intends to support IP service classes). All three protocols allow the use of contention slots for short information transfer and support an MPEG-2 transport mode in the downlink direction.

4.4.2 Major Differences Between the Candidate Protocols

For contention purposes, both the IEEE 802.14 draft and MCNS use minislots; a minislot though in IEEE 802.14 MAC is defined as slot size large enough to transmit 8 bytes of information and overhead, whereas in MCNS a minislot is defined as a power-of-two (2^M , $M=0-7$) multiple of $6.25 \mu s$, a nominal value could be 16 bytes ($M=0$, QPSK with symbol rate of 2.56 M symbols/s) in both cases. DAVIC uses contention slot with size of one ATM cell (53 bytes and 15 bytes of overhead). Besides using contention to request bandwidth initially, piggybacking can be used for additional bandwidth request in 802.14 and MCNS

while polling is an additional option in DAVIC. For piggybacking, a 4 bit EBR field (extended bandwidth request) is used in the header of every Data PDU in IEEE 802.14 MAC (irrespective of whether users are requesting for bandwidth or not), while in MCNS MAC, 4 bytes (user ID + bandwidth requested) are used. An extended header is used with Data PDU, whenever additional bandwidth request has to be made. IEEE 802.14 (and MCNS) do not impose any frame structure whereas a 3 to 6 msec frame structure is recommended in DAVIC.

In terms of signalling, IEEE 802.14 and DAVIC protocol use ATM cells (802.2 LLC encapsulated in AAL5 frame - ITU-T I.363.5 recommendation). MCNS, instead uses variable length MAC messages (802.2 LLC frames). Various MAC messages and calculation of MAC header are entirely different in each of the protocol. Both IEEE 802.14 and MCNS can transport variable length packets and ATM cells, whereas DAVIC relies on ATM transportation only.

IEEE 802.14 specifies a “ternary tree Contention Resolution Algorithm” (CRA), MCNS specifies a “truncated binary exponential back-off” CRA (low cost ethernet type) for the resolution of collisions in the random access slots, whereas DAVIC does not specify the use of a particular CRA. Finally IEEE 802.14 and DAVIC protocols employ a ternary feedback mechanism(i.e. collision, success and idle), while the MCNS protocol employs binary feedback.

Simulation Model and Results

This chapter describes the comparative performance evaluation of the IEEE 802.14 MAC protocol, MCNS MAC protocol and DAVIC MAC protocol under similar assumptions. However, the main focus of this chapter is the IEEE 802.14 MAC protocol. The performance of the DAVIC MAC has been taken from [44, 45]. The comparative performance of the three protocols is described under a reference set of parameters over a wide range of conditions. Finally, we explore the possible performance limitations in case of the channel errors. The simulations to characterize the performance were written in C++ and accurately modeled the real system with the listed assumptions.

5.1 Simulation Assumptions

5.1.1 Frame Structure

The upstream and downstream are segmented into fixed size frames of constant duration longer than the maximum round trip delay plus processing delay in the network. The cell radius in LMDS networks can go up to a maximum of 5 km [19], Therefore the round trip propagation delay from user to hub is $10 \text{ km} \times (3.33\mu\text{s} @ \text{km}) = 33.3\mu\text{s}$. The LMDS Hub can also be connected to a central office via a fiber or point to point radio link. An upstream frame consists of contention and data slots. The IEEE 802.14 and MCNS protocols have

not specified any fixed frame structure, but DAVIC protocol has specified a fixed frame size of 3-6 ms. For the comparative study, we have assumed a frame size close to 3 ms.

5.1.2 CS-DS Size

5.1.2.1 Request Protocol Data Unit (RPDU) Size

In an upstream channel of 802.14 protocol, a minislot is defined as the time required to transmit 8 bytes of data (which includes 2 bytes of CRC) and physical overhead. The primary use of upstream minislots is to provide contention opportunities to the users. The length of the minislot in time can be calculated in integer units of master time base ticks. The frequency of master clock at LMDS hub (headend) is specified as:

$$64 \times S_{\min} = 64 \times 160 \text{ksym/s} = 10.24 \text{MHz}, \quad (5.1)$$

where $S_{\min}$ represents the minimum symbol rate specified in the protocol, which is 160 Ksym/s. With this clock rate, the master time tick can be calculated to 97.65625 nano seconds. A nominal value of 16 bytes mini slots has been assumed in the present work. For example, at 5.12 Mbps (2.56 MSym/s, QPSK), the 16 bytes minislot can be calculated as 256 time-ticks. In upstream channel of the MCNS protocol, a minislot is defined as a power-of-two (2^M , $M=0-7$) multiple of 6.25 μs time-tick). We have implemented a nominal value of 16 bytes, by assuming a data rate of 5.12 Mbps, and by using $M=2$ (4 time-ticks). The required information to be transmitted in a contention (or request) minislot is 6 bytes (including 2 bytes of header check sequence) in MCNS protocol. The remaining bytes are assumed to be overhead. For DAVIC, the contention slot size is defined as 68 bytes (53 bytes ATM cell plus 15 bytes overhead).

5.1.2.2 Physical OverHead Specifications and Assumptions

5.1.2.2.1 Preamble

In IEEE 802.14 and MCNS protocols, the preamble length is variable and is specified between 0-1024 bits.

$$\text{Preamble} = 0 - 1024 \text{ bits} \begin{cases} 0, 2, 4, \dots, 1024 \text{ bits for QPSK, max 512 QPSK symbols} \\ 0, 4, 8, \dots, 1024 \text{ bits for 16 QAM, max 256 QAM symbols} \end{cases}$$

In DAVIC protocol the preamble length is specified to be 4 bytes long.

5.1.2.2.2 GuardBand

In IEEE 802.14 and MCNS protocols, the Guardband is variable and is specified between 5-255 symbols. In DAVIC protocol, the guardband is specified as 1 byte.

5.1.2.2.3 Forward Error Correction

In IEEE 802.14 and MCNS protocols, the forward error correction (t) is a variable and is specified between 0 to 10 bytes (0 implies no FEC). The FEC codeword length (k) can be fixed between 1 to 253 bytes or shortened between 16 to 253 bytes.

In DAVIC protocol, the FEC is specified as shortened RS(63,53, $t=5$) with block length of 255 bytes and configured as a RS(255,245, $t=5$) code with information bits preceded by 192 zero symbols. The FEC specified in DAVIC is to provide error protection to ATM cells.

5.1.2.3 ATM Protocol Data Unit (APDU) Size

In 802.14, for the case of ATM transport, the size of data slots is fixed and defined as time requires to transmit 48 bytes of data, plus 6 bytes of MAC header along with physical overhead (preamble, guard-band, FEC). The size of the minislot and the data slots are chosen to accommodate an integer number of minislots in a data slot. This is a restriction of the IEEE 802.14 MAC protocol, because of synchronization consideration. In MCNS protocol, ATM cells can be transported as a single unit or by concatenating many ATM cells. The size of CS and DS with the overhead assumptions are depicted in figure 5.1. Since we have assumed a minislot of 16 bytes, we have considered the following two cases:

- APDU of 64 bytes (e.g for 802.14; RS(53,59), 1 byte of MAC overhead, preamble = 3 bytes, guard band = 1 byte an optimistic case in terms of overhead for wireless channel)
- APDU of 80 bytes (e.g for 802.14; RS(63,53), 1 byte of MAC overhead, preamble = 4 bytes, guard band = 1 bytes, and rest of bytes are used in quantization; for MCNS; RS(63,53), 6 bytes of MAC overhead, 4 bytes of Extended header (if piggybacking), 4 bytes of preamble, 1 byte of guardband and quantization)

5.1.2.4 Variable Length Protocol Data Unit (VPDU) Size

For 802.14, in case of Variable Length Packet transport. Size of Data Slot is variable with maximum payload of 215 bytes (including 6 bytes of MAC Overhead - with our assumptions for wireless channel) - RS(255,215) because of FEC considerations GF(256) [see Section 5.6 for details].

IEEE 802.14

Figure 5.1: IEEE 802.14 , MCNS and DAVIC Contention and Data Slot Structures (with the specified assumptions)

5.1.3 Contention Slot Positions

Contention minislots are clustered in the beginning of each frame.

5.1.4 Contention Resolution Machines Interleaving

A single Contention Resolution Machine is considered.

5.1.5 Contention Slot Allocation

We have implemented Data Slot Priority Allocation algorithm, as its performance is slightly better than the Contention Slot Priority Algorithm.

5.1.6 Contention (Collision) Resolution Process

We have implemented the *ternary tree collision resolution algorithm with un-blocking and depth priority scanning order*, and is compliant to the latest official draft of IEEE 802.14 [27] at the time the thesis was written.

For MCNS-DOCSIS we have implemented a “*truncated binary exponential backoff*” algorithm with initial backoff window of 16 (2^4) and Maximum backoff window of 256 (2^8). In MCNS MAC, the backoff window can be adjusted dynamically by the LMDS hub, every time an upstream bandwidth allocation MAP is transmitted by LMDS hub in the downstream, it provides a more controlled behavior to contention resolution (see section 4.2.6., 4.2.9., Figure 4.23). We have not exploited the bandwidth MAP updating capability of the protocol in this work. Although, we have implemented a simple distributed Ethernet type algorithm with a fixed initial and maximum backoff window.

5.1.7 Newcomer Entry Persistence

We have implemented a variable entry persistence scheme using Method-1 (Section 3.1.2.9), as it has been shown to perform better than Methods 2, 3 and 4. Method 5 has not been implemented, because this method is under proposal and revision in the IEEE 802.14 Working Group [8, 28].

5.1.8 Bandwidth Request Procedure

An active terminal with a non-empty buffer sends a bandwidth request in a Contention (request) minislot. If the user is involved in a collision resolution process, his bandwidth request is updated in retransmission according to the current buffer status of the user. Once the initial request is accepted, a user can request additional bandwidth by piggybacking on data slots. We have considered only one outstanding request per user at any time. Although, in the IEEE 802.14 draft the provision of concurrent requests (more than one outstanding request per user) has been specified.

5.1.9 Feedback/Grant Frequency

Feedback of Contention and Grant for bandwidth request are provided at the beginning of each frame. Feedback of frame i is provided in frame $i+1$. This assumption is valid for fixed CS:DS ratio frame structure, but is rather optimistic for the adaptive CS:DS ratio frame structure, in which case the LMDS hub may not be able to provide the feedback and grant for all the transmission attempts in the previous frame (Round trip delay and Head end Processing delay). This implies that the effect of *late-request* has not been considered in this work (See section 3.2.1.2).

5.1.10 Bandwidth Scheduling and Buffer Size

Bandwidth allocation of data slots is done in a Round Robin service discipline. In the case of mixture of traffic, priority traffic is served first, and bandwidth allocation for each type of traffic is done in a round robin service discipline.

5.2 MAC Protocol

MAC protocol is a collection of components each performing a certain number of functions which vary from ranging, acquisition process, to defining a frame format (including defining contention and data slots), bandwidth request, allocation procedures, and contention resolution process. Since IEEE 802.14 MAC supports an ATM mode, it also defines different classes of traffic support such as Constant Bit Rate (CBR), Variable Bit Rate (VBR) and Available Bit Rate (ABR). To evaluate the comparative performance of the IEEE 802.14 MAC protocol with DAVIC 1.2 MAC and MCNS MAC protocols, we build the protocol model under similar assumptions which are listed above. For the initial performance evaluation, we have used three traffic source models as explained in section 5.3. We have simulated the MAC protocol using a fixed upstream frame structure although the protocol can be simulated without specifying any explicit frame structure (see section 3.2). We have implemented bandwidth allocation procedure at the LMDS hub to controls the bandwidth grant to the successful request. The implemented Contention (collision) resolution process (ternary tree in the case of IEEE 802.14 MAC and binary exponential backoff in the case of MCNS) controls the transmission and retransmission of bandwidth requests.

5.2.1 IEEE 802.14 MAC Model General Description

The upstream frames are divided into a sequence of contention minislots and data slots. Data traffic is carried via ATM cells or Variable Length Packets as such, by concatenating a variable number of minislots. Data packets are generated at the user end by traffic models. In case of ATM transport, data packets are segmented into 48-byte chunks and encapsulated in a 53-byte ATM cell. Each data slot can carry only one ATM cell. In case of Variable Length Packet, the size of data packets is variable, subject to the maximum size limit imposed by FEC. Contention minislots are used for bandwidth request in contention mode. The size of the minislot and data slots are chosen so that an integer number of minislots can be accommodated in a data slot. This is the restriction of the protocol, imposed for synchronization purposes. In the simulated upstream slot structure, a minislot size is assumed to be 16 bytes (8 bytes of data plus overhead). In the case of ATM transport, the payload of an ATM cell fits in three minislots: ATM header of 5 bytes, MAC overhead (1 byte), and PHY overhead requires a further minislot. Thus, it requires four minislots to accommodate an ATM cell. This assumption is little optimistic for the wireless environment, therefore we have done simulations with a data slot size of 80 bytes (5 minislots) as well.

Figure 5.2: Upstream Frame for ATM Mode

In each frame if the contention minislots are present, they are *clustered* at the beginning of the frame. This is only one possible way of implementing the protocol, as the Contention minislots could be scattered uniformly in a frame. A sample upstream frame structure for ATM transport mode is shown in Fig 5.2. with the assumptions of 16 bytes contention slot and 64 bytes data slot and upstream data rate of 2Mbps. The ratio of CS:DS per frame is shown to be 16:8 (2:1).

The simulation works in the following manner, When a user has data to transmit, he waits for a contention slot specifically allocated by the LMDS hub for newcomers. Then he randomly selects one of the contention slots out of the group of specified contention slots and transmits with probability p . The user then waits for acknowledgment of his request. Users are not allowed to transmit another request while waiting for acknowledgment of their previous request.

If more than one user attempts to transmit their request in the same contention slot, collision occurs. Collisions are resolved by using the *ternary tree collision resolution process* (see section 3.12, 4.1.5.3). The downstream frame provides the feedback information about the collisions and grants of bandwidth.

If a user is successful in selecting a contention slot to transmit, he transmits his request for bandwidth in that contention slot. If the user gets involved in a the collision resolution process, the bandwidth request size is updated every time he retransmits his request (depending upon the current buffer status of the user). Finally when the user is able to successfully transmit his request, he waits for bandwidth allocation. Bandwidth allocation is entirely controlled by the LMDS hub. First the LMDS hub decides the proportion of contention slots and data slots per frame or CS/DS ratio. This ratio can be fixed or can vary depending upon the Contention Slot Allocation algorithm (CSA). The LMDS hub then allocates data slots to the user using a specified scheduling discipline.

5.3 Traffic Models

5.3.1 The WWW Traffic Model

This traffic model has been taken from paper by S. Deng [59], which describes the data traffic characteristics of individual WWW users. Data was collected from the access link between GTE Laboratories and the Internet in two sessions, from 2pm to 4pm on August 2, 1995 and 9:30am to 12:00pm on August 3, 1995.

When a user requests a page, several URL (Uniform Resource Locator) requests may follow. The first one transmits the user's direct request to the server. The following requests may be automatically generated by the browser program, e.g. each in-line image (logos, icons, buttons, etc.). This requires a separate request to be sent automatically by the client program during the download of the page. The suggested model is described by the distributions of three random variables:

Figure 5.3: A Sample of WWW traffic pattern.

ON period Distribution (X in seconds)

The actual data collected appear to belong to the Weibull distribution, having the following probability density function.

$$p(x) = \frac{k}{\theta} \left(\frac{x}{\theta}\right)^{k-1} \exp\left(-\left(\frac{x}{\theta}\right)^k\right) \quad (5.2)$$

$$k = 0.88 \quad \text{and} \quad \theta = 90.017 \quad (5.3)$$

OFF period Distribution (Y in seconds)

It represents the thinking time of the user, which results in the significant pause in the communication activities. The empirical results suggest that the duration of the OFF period belongs to a Pareto distribution having the following probability density function.

$$p(y) = \alpha \cdot \frac{k^\alpha}{y^{\alpha+1}} \quad (5.4)$$

$$k = 60 \quad \text{and} \quad \alpha = 0.5 \quad (5.5)$$

Packet Inter-arrival Distribution (Z in seconds)

The empirical results show that the distribution of packets interarrival during the ON period is Weibull distributed with $p(z)$ and parameters k and θ as follows.

$$p(z) = \frac{k}{\theta} \left(\frac{z}{\theta}\right)^{k-1} \exp\left(-\left(\frac{z}{\theta}\right)^k\right) \quad (5.6)$$

$$k = 0.5 \quad \text{and} \quad \theta = \exp(1.5) \quad (5.7)$$

5.3.2 Bursty Traffic Model (Random Short)

This type of traffic has been referred to as the Random Short by IEEE 802.14 WG and has been proposed for the initial performance evaluation of the IEEE 802.14 MAC proposals [37]. The message length of each generated packet is 48 bytes (and hence requires one ATM cell for its transmission). Active sources generate this traffic according to a Poisson distribution with a varying mean arrival rate. The ration behind the use of this traffic model is to see the performance of the protocol under stressed contention channel conditions. This type of traffic source could be used to model video game applications.

5.3.3 Batch-Poisson Traffic Model

This model is used by IEEE 802.14 WG for the performance evaluation of MAC proposals. Figure 5.4 shows the packet length distribution of Batch-Poisson traffic model with batch size corresponding to observed Ethernet PDUs [5, 37]. The message size distribution is defined as follows. Messages of sizes 64, 128, 356, 512, 1024, and 1516 bytes are generated by each source with a probability of 0.6, 0.06, 0.04, 0.02, 0.25, and 0.03, respectively.

Figure 5.4: Message size distribution for Batch Poisson Model

The average packet length can be calculated as 368.1 bytes. The message interarrival $T = 1/\lambda$ is taken to be exponentially, distributed with a mean arrival rate of λ packets per second.

5.4 Performance Bounds and Simulation Results

5.4.1 Definition of Performance Parameters

Offered Load: Offered Load is defined as the total load generated by all the users in a network including their overhead. This load corresponds to data packets generated by the users, and does not include the load on contention channels. We have shown Offered Load as a fraction of Link Capacity in the simulation results. For example, if the Link Capacity is 2 Mbps, then Offered Load of 0.5 means that total traffic rate of all the users is 1 Mbps, including MAC and Physical Over Head.

Access Delay: In the upstream direction, the access delay for a packet generated by a user is defined as the difference between the time of arrival of the first byte of a packet at the user queue to the time the packet is completely received (until last byte) at the Head-End (Hub). The major components of Access Delay are contention request delay, queuing delay, Head-End processing delay (HPD), round trip propagation delay (RTD), and packet transmission delay. For the case of Fixed CS:DS ratio frame structure the HPD and RTD are covered by the fixed portion of data slots in a frame (See Section 3.2.1, Figure 3.11). For the Adaptive CS:DS ratio case we have considered HPD and RTD = 0.

Throughput: Throughput is defined as the actual information getting through the network. This does not include MAC or Physical Overhead.

Request Delay: Request Delay is defined as the time taken by a user to send a bandwidth request and get a positive acknowledgment. The two major components of the request delay are the delay caused by the Collision resolution, and the delay in receiving the feedback.

Coefficient of Delay Variation: The coefficient of delay variation is defined as the ratio of standard deviation to the mean access delay.

Collision Multiplicity: The collision multiplicity is defined as the number of users collided in a contention slot.

5.4.2 Simulation Results

The common input parameters for the simulations and the set of reference values for three different traffic models are shown in table 5.1. The parameters and values specific to a particular experiment are provided along with the discussion of that experiment. In the case of absence of any parameter, table 5.1 has to be referred.

Table 5.1: Reference Configuration for WWW, Batch Poisson and Random Short

<i>Simulation Parameter</i>	<i>Value</i>
Upstream data transmission rate	2 Mbps
Contention Slot Size	16 bytes
Data Slot Size	64 bytes
Frame Size	3.072 ms
No. of max. CS per frame	48
No. of max DS per frame	12
Size of DS: Size of CS	1:4
Processing Delay	0
Entry persistence	P-persistence - Method 1 (see section 3.1.2.9)

5.4.3 WWW Traffic Result

Table 5.2: Parameters specific to WWW Traffic Simulation

<i>Simulation Parameter</i>	<i>Value for WWW Traffic Model</i>
Number of Active Stations	200 - 5000
CS:DS Ratio per frame & Piggybacking	Fixed (16:8) and Adaptive - DS Priority algorithm with and w/o PB
Transport Mode	ATM

This section describes the performance evaluation of IEEE 802.14 MAC protocol using the World Wide Web traffic model (figure 5.3). The traffic model is an on-off source with on, off and interarrival distributions described in section 5.3. Simulation parameters for this experiment are described in table 5.1. and 5.2. The experiment is conducted with two schemes of Contention Slot Allocation.

- Fixed CS/DS Ratio per frame
- Dynamic (Adaptive) CS/DS ratio per frame

In Fixed CS/DS Ratio scheme, we have implemented a ratio of 2:1 (16 Contention slots and 8 data slots per frame). In terms of capacity, one-third of frame duration is allocated to contention slots and the other two-thirds of the frame duration is allocated to data slots. In the dynamic contention slot allocation, number of contention slots are varied on a frame to frame basis. The contention slot algorithm implemented here is a data slot priority algorithm (see section 3.3). Fig 5.5 and Fig. 5.6 show the mean access delay performance of the IEEE 802.14 MAC protocol by varying the user population and offered load.

5.4.3.1 Discussion on WWW Traffic Results

As we observe from figure 5.5, the mean access delay remains less than 500 ms for a population up to 3000 users for Fixed CS:DS ratio case. After that the mean access delay increases exponentially because of the large user population, and increased collision multiplicity. However from figure 5.5 we see that the use of dynamic (or adaptive) contention slot allocation improves the situation considerably. Further it can be noted that the mean access delay remains less than 500 milliseconds for 4200 users, and 1000 millisecond for 5000 users. The reason is obvious, we provide as much contention slots as possible (by converting all the unused data slots into contention slots). Fig. 5.7 shows the collision multiplicity for 3000 users for adaptive CS:DS ratio case. The highly unpredictable behavior, variability of traffic source, and large user population, are major reasons for the exponential increase in access delay. The effect of piggybacking that is shown in Fig 5.7, provides only 5 to 10 percent improvement in this case here. The reason for this is that due to the large user population, and high variability and unpredictable behavior of the WWW traffic source model, longer CRI's are triggered. Therefore the request does not go through for a long period of time, and the user updates its request size whenever retransmitting its request (depending upon the current buffer status). Finally, when the request goes through the small request size does not produce large queuing delay. Therefore, users do not get frequent opportunity to piggyback. Figure 5.9 shows the request delay component of the access delay. We can observe, the request delay is a very significant component of the access delay. Although we do not see a very significant improvement by using piggybacking.

Figure 5.5: Mean Access Delay for WWW

Figure 5.6: Mean Access Delay for WWW

Figure 5.7: Collision Multiplicity for 3000 WWW Traffic Users

Figure 5.8: Improvement by Using Piggybacking in WWW Traffic

Figure 5.9: Request Delay Component of Access Delay for WWW (Adaptive CS:DS ratio case)

5.4.4 Random Short Traffic Results

Table 5.3: Parameters specific to Random Short Traffic Simulation

<i>Simulation Parameter</i>	<i>Value for Random Short Traffic Model</i>
Number of Active Stations	200
Length of Simulation Run	307 sec
Length of run prior to gathering statistics	10 % of simulated time
CS:DS Ratio per frame & Piggybacking	Fixed (16:8) with and without piggybacking Fixed (20:7) with and without piggybacking Adaptive - DS Priority algorithm with and without piggybacking
Transport	ATM

This experiment is conducted for random short traffic with configuration parameters shown in table 5.3. To find the optimum CS:DS ratio for this traffic, we have conducted the experiment for Fixed 16:8 (or 2:1) and Fixed 20:7 (~ 3:1) ratio cases. Both ratios were

tested with and without piggybacking to see the effect of piggybacking on an optimum ratio. An adaptive CS:DS experiment was also conducted with and without piggybacking.

5.4.4.1 Discussion of Results

Figure 5.10: Mean Access Delay for Random Short Traffic

Figure 5.10 shows the mean access delay performance for fixed and adaptive CS:DS ratio cases. The maximum achievable bound (asymptotically) for offered load is 56% (or throughput bound of 42% by noting that 48 bytes of information out of 64 bytes Data Slot) for Fixed 3:1 ratio, and 65% for Fixed 2:1 ratio case. The theoretical limit on the link capacity for Fixed 3:1 (20:7) is $7/12$ or 58.33%, and for Fixed 2:1 (16:8) is $8/12$ or 66.7%. The access delay for Fixed 3:1 ratio case is less than Fixed 2:1 ratio case. When we use piggybacking in Fixed 2:1 case, some of the contention load is carried by piggybacking, so

the delay performance of Fixed 2:1 with piggybacking becomes close to Fixed 3:1 case. The advantage of adaptive CS:DS ratio scheme over fixed CS:DS ratio can be easily seen under this traffic condition. The reason is that adaptive CS:DS ratio structure adapts itself to the traffic conditions by converting all the unused data slots to contention slots. The use of piggybacking further enhances the performance at higher load conditions and reduce the requirement of contention slots. The maximum achievable bound of offered load for adaptive CS:DS ratio case is 70% without piggybacking and 75% with piggybacking.

Figure 5.11: Collision Multiplicity for Random Short Traffic Load = 70 %

The multiplicity of collisions (number of users colliding in a contention slot) is shown in figure 5.11. The multiplicity histogram is shown for stressed contention channel (Random short traffic at 70%, no piggybacking). In this sample, 28.28% of contention slots offered in this simulation run results into collision. 32.4% of the offered contention slots were idle, and 39% of the contention slots results into successful transmission. The distribution of collisions is shown in figure 5.11 (does not show the idle and successful contention slots). We see from this figure, the first two bins (corresponding to multiplicity of 2 and 3) contain the majority of the sample (82.15%). The maximum multiplicity

occurring in this experiment is 15 and the percentage of samples for more than 8 users transmitting in the same contention slot is less than 0.5% [corresponding probability is 0.00134, by including idle and success]. The reason for this is due to the implementation of variable entry persistence (p-persistence) in the protocol. The probability for each newcomer to enter into the network has been provided by estimating the backlog populations, and the availability of contention slots in a frame. (Section 3.1.2.9.1).

5.4.5 Batch Poisson Traffic Model Results

The results shown in this section are for bursty source with a batch Poisson arrival process and a message size distribution as shown in Fig. 5.4. The average size of a message is 368.1 bytes or 8.3 ATM cells (including PHY, MAC and quantization overhead). This experiment is conducted with both fixed and dynamic (adaptive) contention slot allocation. In the fixed scheme, the number of contention minislots and reservation data slots in a frame is fixed. In the dynamic scheme, the LMDS hub (headend) varies the ratio of contention minislots to reservation data slots based on the traffic in the system. The simulation parameters for these experiments are listed in table 5.1 and 5.4.

Table 5.4: Parameters specific to Batch Poisson Traffic Simulation

<i>Simulation Parameter</i>	<i>Value for Batch Poisson Traffic Model</i>
Number of Active Stations	200
CS:DS Ratio per frame & Piggybacking	Fixed (16:8) , Fixed (8:10) , Fixed (4:11) with no piggybacking Adaptive - DS Priority algorithm with and without piggybacking
Transport	ATM
Entry persistence	P-persistence (Method 1) (see section 3.1.2.9.1)

5.4.5.1 Fixed CS/DS (Contention Slot/Data Slot) Ratio

In the case of fixed CS/DS allocation, every frame has a fixed number of contention mini slots and data slots. However, the optimum ratio depends on the traffic pattern. Figure 5.12 shows the mean access delay performance of the protocol when the load is varied for CS:DS ratios of 16:8 (2:1), 8:10 (4:5), and 4:11 (~1:3) for Batch Poisson traffic type. It is readily observed that a CS/DS ratio of 4:11 (~1:3) does well for this type of traffic. The reason for this is that due to large average packet size in this type of traffic (8.3 ATM cells), fewer contention slots are needed.

An approximate idea can be taken from this calculation. On an average we need $1/\eta_{\text{CRA}}$ CS to have one successful transmission, where η_{CRA} denotes the efficiency of the contention resolution algorithm (CRA). If we assume that the efficiency of CRA is approximately 33.3%, then on average we need 3 contention slots for each successful request. If the average request is of N data slots, then ratio of 3: N is optimum. In the Batch Poisson traffic model the average request size can be calculated from the message size distribution as 8.3 ATM cells (assuming no piggybacking). Therefore, CS/DS ratio close to 3:9 (1:3) will show the optimum performance. The experiments with fixed CS/DS ratio were conducted without the use of piggybacking.

5.4.5.2 Dynamic (Adaptive) CS/DS Ratio

The dynamic contention slot allocation scheme adapts the CS/DS ratio according to the traffic (see section 3.3). At low load conditions, all the unused data slots are converted into contention slots. Therefore, these excess contention slots keep the access delay close to the lower bound. At higher loads, queues start to build up, and the availability of contention slots is reduced, therefore the CS:DS ratio begins to settle down close to the optimum

value. Therefore we see the performance of adaptive schemes become close to the optimum case of Fixed 4:11 at higher load conditions. As we can see from this figure, piggybacking also provides some improvement at higher loads (load > 0.6). The reason again is due to the large average request size and lower contention load, and we don't see any performance improvement at lower loads. At higher loads queues start to build up, and the users operate in piggyback mode. So more bandwidth (DS priority algorithm) is being used to fulfil the existing bandwidth requirement in the grant queue of LMDS hub(head-end), and further reducing the availability of contention slots.

Figure 5.12: Mean Access Delay performance of Batch Poisson Traffic

5.4.5.3 Confidence Intervals

We have calculated the confidence intervals for one of the experiments conducted for Batch Poisson traffic model. The simulation parameters are the same as described in table 5.1 and 5.4. The experiment is conducted using dynamic contention slot allocation (DS priority) without piggybacking. We have conducted 10 simulations for each point with different seeds to the random number generator, and calculated the upper and lower bounds of the mean access delay for 95% confidence interval.

5.4.5.3.1 Procedure for Finding Confidence Intervals

The point estimation of mean access delay $\hat{M}$ is found by taking the mean of n samples of mean access delay from different simulations runs with different seeds (m_i) [32]. The collection of the n samples $\{m_1, \dots, m_n\}$ is done in the following way.

For various load conditions, each sample is collected by dividing each simulation run into two phases.

- Initialization phase from time 0 to time T_0
- Data collection phase from time T_0 to time $T_0 + T_E$

The length of the Initialization phase has been found by running various pilot runs in order, to find the steady-state of the simulation.

$$\hat{M} = \frac{1}{n} \sum_{i=1}^n m_i \quad (5.8)$$

The confidence interval estimation with $100(1 - \alpha)\%$ confidence is calculated as:

$$\hat{M} - \{t_{\alpha/2,f} \cdot \hat{\sigma}(\hat{M})\} \leq M \leq \hat{M} + \{t_{\alpha/2,f} \cdot \hat{\sigma}(\hat{M})\} \quad (5.9)$$

where $t_{\alpha,f}$ is the 100 (1- α) percentage point of a *Student-t* distribution with f degrees of freedom (this value can be found from *Student-t* distribution table). The $\hat{\sigma}(\hat{M})$ is called *standard-error* of the point estimator $\hat{M}$ and is calculated as:

$$\hat{\sigma}(\hat{M}) = \frac{S}{\sqrt{n}} \quad (5.10)$$

Where S^2 is the sample variance which can be calculated for statistical independence and identically distributed as;

$$S^2 = \sum_{i=1}^n \frac{(m_i - \hat{M})^2}{n-1} \quad (5.11)$$

Figure 5.13: Confidence Interval - Batch Poisson Traffic without Piggybacking and Dynamic CS/DS ratio

As we see from figure 5.13, the upper and lower bounds are very close for load conditions less than 70% (or 0.7) of the total capacity. After that we observe the slight variability, we notice that because these points are close to the breakdown region.

5.4.6 Heterogeneous Scenario A

This scenario is proposed by IEEE 802.14 MAC Working group for initial performance evaluation of the MAC proposals [37]. In this scenario 80% of the Traffic Load is generated by high priority Random Short (1 ATM Cell), and 20% of the Traffic Load is generated by the low priority Batch Poisson Traffic. The rationale behind this scenario is two fold. First, it allows to test the protocol behavior for delay sensitive traffic in the presence of other traffic. Second, it creates a situation, where the load on contention and data slots, allows us to see the advantage of piggybacking. The simulation configuration parameters for scenario-A are listed in Table 5.5

Table 5.5: Simulation Configuration For Heterogeneous Scenario A

<i>Simulation Parameter</i>	<i>Scenario A</i>
Upstream data rate	5.12 Mbps
Number of Active Stations	160 - Random Short Traffic Users , 40 - Batch Poisson Traffic Users
Contention Slot Size	16 bytes
Data Slot Size	80 bytes
Frame Size	3 ms
No. of max. CS per frame	120
No. of max DS per frame	32
Size of DS: Size of CS	1:5
Contention Slot Allocation & PB	Dynamic - Data Slot Priority With and Without Piggybacking
Traffic Model	80% Random Short (High Priority) , 20% Batch Poisson (Low Priority)
Transport Mode	ATM

5.4.6.1 Discussion Of Results- Scenario A

This experiment is performed for IEEE 802.14 MAC protocol with and without the use of piggybacking. This scenario is a typical example of a situation where piggybacking feature of the protocol shows some advantage. Figure 5.14 shows the Mean Access Delay behavior for low priority Batch Poisson traffic (Figure 5.4), in the presence of high priority Random Short Traffic. We observe that piggybacking starts showing advantages after the offered load becomes more than 0.5 of the link capacity. (The offered load shown is the total load generated by both traffic sources)

Figure 5.14: Scenario A (80% Random Short, 20 % Batch Poisson) Effect of Piggybacking for Batch Poisson

Figure 5.15 shows the Mean Access Delay performance of high priority Random Short (1 ATM cell) Traffic in the same experiment. In this case, also we also observe significant advantage of using piggybacking when offered load is beyond the 0.5 of the link capacity. Figure 5.16 shows the Mean Access delay performance of Random Short Traffic (high pri-

ority) and Batch Poisson Traffic (low priority). As we see, due to the higher priority given to Random Short Traffic, the access delay is lower than the low priority Batch Poisson Traffic, for almost complete range of the offered load. The only exception we see is at very low load conditions (Offered Load < 0.2), where the Mean Access Delay for Batch Poisson traffic (low priority) is slightly less than the Random Short Traffic.

Figure 5.15: Scenario A (80% Random Short, 20 % Batch Poisson)
Effect of Piggybacking for Random Short

Figure 5.16: Scenario A - Random Short (80 % High Priority)
and Batch Poisson Traffic (20 % Low Priority)

The reason for this is that at very low load conditions, the access delay is dominated by the request delay (delay incurred by a bandwidth request to become successful). The average request size for Batch Poisson Traffic model is 8.3 ATM cells and that of Random Short traffic model is 1 ATM cell. Therefore Random Short Traffic generates more requests, and contends for the same contention slots with the low priority Batch Poisson Traffic. After offered Load > 0.2 of the link capacity, the high priority scheduling of data slots for Random Short Traffic starts showing its effect, and access delay (which is sum total of request delay and queuing delay) becomes lower than the low priority Batch Poisson Traffic.

At this point we would like to discuss the concept of *Access Priority* and *Service Priority*. It is evident from this experiment that we have given higher priority to Random Short Traffic over Batch Poisson Traffic in terms of serving bandwidth to their data packets. The concept is called *Service Priority*. If we allow high priority users and low priority users to contend for the same contention slots, it could cause higher delay and delay variation in certain situations, which may not be acceptable for some delay sensitive applications to meet their QoS requirements.

This situation can be improved by providing priority to contention slots allocation to high priority users than to low priority users. A separate Contention Resolution Algorithm (Ternary tree in the case of 802.14) can operate for each traffic type. In this scheme, we are giving priority to the user to access the network, i.e send their bandwidth request, this concept is known as *Access Priority*.

5.5 Effect of Channel Errors

No channel is free from the presence of errors, whether it is HFC channel affected by ingress noise and interference or LMDS channel affected by fading, multipath, and interference. The wireless environment is considered to be the most hostile among other environments (Fiber, Coax, Copper); therefore special consideration has to be given to the effect of channel errors for designing LMDS systems. Typical characteristics of the LMDS channel could be a channel with highly variable raw bit error rate, of the order of 10^{-2} to 10^{-3} with relatively small propagation delay with respect to packet size. Since LMDS systems are going to serve as a bridge between high speed backbone networks and customer premises, therefore some of the service requirement demand very low bit error rate (e.g. 10^{-10}). To provide reliability in the wireless channel, a combination of FEC (Forward Error Correction) and ARQ (Automatic Repeat Request) has been proposed [33]. All three candidate protocols have proposed CRC sequences to detect the errors in contention and data slots. The Forward Error Correction by the use of Reed Solomon coding has been proposed in all the three protocols (IEEE 802.14, MCNS, DAVIC) suitable for their respective environment (See section 5.1.2.2.3).

Nevertheless, the following section considers that even after coding, there is always a possibility of some error, however small it may be. The focus of this section is to discuss the error management and performance of the protocols. Both LMDS and HFC systems are asymmetric in nature, with limited power, and smaller processing capabilities in Customer premises equipment, (CPE) than the LMDS hub (headend). So the design proposals are required to put as much intelligence and processing in the LMDS hub as possible, and make the CPE as simple as possible. So on these guidelines, the approach taken by 802.14 WG in designing the MAC protocol, is to discard any unexpected message by the user and provide proper recovery mechanisms to avoid error propagation for a

period of time. An error can affect both control or data slots in an upstream and downstream direction. Both the issues have been addressed briefly here.

5.5.1 Effect on Collision Resolution Algorithm

Following are the situations when collision, idle, and success is detected by the LMDS hub (headend).

- **Collisions** occur in a contention slot when two or more users transmit in the same slot. The slot status is declared to be a collision also in the event of noise causing corruption of data in a contention slot. The detection of collision in a contention slot will be by corrupted preamble, and uncorrected errors by FEC.
- **Idle** is declared in a contention slot if no RF energy is estimated in an upstream contention slot by the LMDS hub (headend).
- **Success** is declared for a contention slot if RF energy has been detected in the slot and the preamble of that slot is not corrupted and FEC in the slot either detects no error or detects and corrects errors.

5.5.1.1 Discussion on Effect of Errors on Contention Slots

Figure 5.17: Example of Ternary Tree

An example of detectable error is shown in Fig 5.17. This figure shows five contention minislots which are declared collisions in frame i (labeled as C_1 to C_5). S represents a success in a contention slot, and I represents an idle contention slot. For simplicity we have not shown data slots in a frame. Since the CRA is a ternary tree, and for every collision we will open 3 contention minislots, in the next frame (considering we have required contention mini slots available in the next frame). Every user reads the feedback map from the downstream to know the status of their attempt, and associates RQ (request queue) number in case of collision (see section 4.1.5.3). In the frame $i+1$, we have shown 15 contention slots, all marked with their respective RQ numbers. In order to resolve a collision we provide priority to the oldest collision (largest RQ number first), to maintain FIFO as far as possible. (see section 3.1.2.8 and 4.1.5.3). Suppose that C_4 is declared as a collision because of the error condition, and there was no user transmitting in that slot. So in the next frame the assignment of contention minislots with RQ number 2 are wasted, as there was no user to transmit in those slots. Suppose that there was only one user attempting to transmit in slot marked C_4 , although it was supposed to be declared successful but because of errors it was declared a collision. In both the cases we have delayed the data transmission (and increased the access delay) of the user and wasted contention slots.

An example of an undetected error could be when the LMDS hub (headend) broadcasts a feedback map indicating the availability of contention slots with a particular RQ number and a user receives a corrupted feedback MAP. Either the user will interpret a wrong RQ number or may not be able to interpret the RQ number at all. If a user keeps on waiting for the assigned RQ number, he will end up in a deadlock situation. In order to recover from deadlock, 802.14 MAC suggests that a user should be allowed to transmit in RQ number less than or equal to the RQ number assigned to him. In this example, a user with RQ number 4 can transmit in any slot marked less than or equal to 4 (for example RQ = 3). Table 5.6 lists all the possibilities of Contention state errors and their outcome and

table 5.7 describes the description of feedback states. Situations 1A, 5B and 9C are situations where there is no error (shown by *Italics* in the table). All other situations denote certain error conditions in feedback. Two of these situations are explained below.

Table 5.6: Contention State Errors

Index	Feedback Sent by H/E	Feedback Received	Actual Contention Outcome		
			A	B	C
			IDLE	SUCCESS	COLLISION
1	IDLE	IDLE	<i>IDLE</i>	FB_ERR	FB_ERR
2	IDLE	SUCCESS	TX_ERR	TO_WDG	TO_WDG
3	IDLE	COLLISION	TX_ERR	SC_ERR	RQ_ERR
4	SUCCESS	IDLE	DMS_VOID	FB_ERR DMS_VOID	FB_ERR DMS_VOID
5	SUCCESS	SUCCESS	TX_ERR DMS_VOID	<i>SUCCESS</i>	TO_WDG *
6	SUCCESS	COLLISION	TX_ERR DMS_VOID	SC_ERR DMS_VOID	RQ_ERR DMS_VOID
7	COLLISION	IDLE	RMS_VOID	FB_ERR RMS_VOID	FB_ERR RMS_VOID
8	COLLISION	SUCCESS	TX_ERR RMS_VOID	RMS_VOID TO_WDG	RMS_VOID TO_WDG
9	COLLISION	COLLISION	TX_ERR RMS_VOID	SC_ERR	<i>COLLISION</i>

* In the event that HC happens to incorrectly detects a successful transmission from a particular user which has transmitted one of the request, then the user will have a CAPTURE instead of TO_WDG

5C: In this case actual outcome of the contention is collision, and collision is detected as a success by LMDS hub (headend). The LMDS hub (headend) transmits a feedback of successful transmission and the same feedback is received by user(s). The involved users will then go to *wait for data grant state*. If the grant is not received within specified time set by *DataGrantTimer*, users will reset their states and contend for bandwidth again. In case of capture, one of the involved users in the collision will capture the contention slot and receive the grant, and the other user(s) will contend for bandwidth again.

7B: In this case actual outcome of contention is success, the feedback transmitted by the LMDS hub (headend) is success, but the feedback received by the user declares a collision. This means that the user will try to interpret an RQ number associated with this feedback, and will contend for bandwidth again. This situation will waste contention mini slots.

Table 5.7: Contention Feedback State Descriptions

Feedback	Definition
IDLE	No RPDU was transmitted in the RMS, and the contention outcome has been correctly detected and communicated to the MAC User.
SUCCESS	A single RPDU was transmitted in the RMS, and the contention outcome has been correctly detected and communicated to the MAC User which transmitted the RPDU.
COLLISION	Multiple RPDUs were transmitted in the RMS and the contention outcome has been correctly detected and communicated to one or more MAC Users which transmitted the RPDUs
FB_ERR	A particular MAC User transmitted an RPDU in the RMS, but the feedback received incorrectly indicates that there was no such transmission, regardless of whether the actual contention outcome was IDLE, SUCCESS, or COLLISION. This error is detectable by the MAC User.
TX_ERR	No RPDU was transmitted in the RMS, but the feedback received indicates that at least one RPDU was transmitted in the RMS. This error is not detectable by the MAC User.
RQ_ERR	Multiple RPDUs were transmitted in the RMS and a MAC User which transmitted one of the RPDUs receives an RQ assignment that was not intended by the HC. This error is not detectable by the MAC User
SC_ERR	A single RPDU was transmitted in the RMS, but the MAC User which transmitted the RPDU receives feedback indicating a COLLISION status in the RMS and an RQ was assigned. This error is not detectable by the MAC User.
DMS_VOID	A MAC User will receive a DMS Grant when it is not expected, either because the MAC User has not transmitted any RPDU in the RMS or it has not received any feedback indicating SUCCESS.
RMS_VOID	The HC will grant an RMS associated with $RQ > 0$ to a particular MAC User which is not expecting any such RMS grant, either because the MAC User has not transmitted an RPDU in the RMS or it has not received feedback indicating COLLISION.
TO_WDG	A MAC User which transmitted an RPDU in the RMS receives feedback indicating SUCCESS even though the HC has incorrectly detected contention outcome. The MAC user will wait for a DMS grant that will never be sent.

All the error conditions mentioned in table 5.6 can be handled by introducing proper recovery mechanisms with the help of various timers at different states of the users.

5.5.2 Channel Errors Simulation

The following section shows the performance of IEEE 802.14 protocol in the presence of channel errors. The experiments are performed for Batch Poisson Traffic and Random Short Traffic using ATM mode. In ATM transport every packet generated by a user is segmented into ATM cells. So if errors occur in any of the data slots one possibility is to retransmit the whole packet, (Go-back-N ARQ), or just retransmit the selective segment (data slot) in error (Selective Repeat ARQ). We have implemented the selective ARQ protocol in our simulation. It is obvious that if we retransmit a complete packet (instead of just a segment), throughput will further decrease.

Table 5.8: Configuration for Channel Error Test on Batch Poisson Traffic

<i>Simulation Parameter</i>	<i>Value for Channel Errors Test for Batch Poisson Traffic Model</i>	<i>Value for Channel Errors Test for Random Short Traffic Model</i>
Upstream data transmission rate	2 Mbps	5.12 Mbps
Number of Active Stations	200	200
Contention Slot Size	16 bytes	16 bytes
Data Slot Size	64 bytes	80 bytes
Frame Size	3.072 ms	3 ms
No. of max. CS per frame	48	120
No. of max DS per frame	12	24
Size of DS: Size of CS	1:4	1:5
CS:DS Ratio per frame	Fixed (16:8) with no piggybacking	Dynamic DS-Priority with Piggybacking
Transport	ATM	ATM
Error Percentage	5% , 10%, 15%, 20% errors in data slots transmission and Contention slot detection (Table 5.6, 9A, 9B)	5%, 10%, 15% errors in data slots transmission and Contention slot detection (Table 5.4, 9A, 9B)

Figure 5.18 shows a test conducted to see the combined effect of data and contention slot errors for Batch Poisson Traffic. Only two cases of contention slot errors were considered

out of the cases listed above (9A and 9B) in Table 5.6. The first case was when idle slots were declared collision slots, and the other was when successful slots were declared as collision slots (as explained above). The effect of contention slot detection was not very prominent in this case because for this traffic average request size is 8.3 packets, so a relatively smaller number of CS are required. The errors in data slots cause retransmissions and produce additional load on the upstream channel, which decreases the maximum load a channel can carry.

Figure 5.18: Effect of errors in Contention and Data slots for Batch Poisson, Fixed CS:DS (2:1) ratio

Figure 5.19 shows a test conducted to see combined effect of data and contention slot errors for Random Short Traffic. We can observe from the figure that the performance degradation as the percentage of error increases from 5 to 15%. The maximum achievable throughput decreases as the percentage of error increases. The effect is more prominent at higher loads, as at lower load conditions, bandwidth is available for retransmissions.

Figure 5.20 shows the coefficient of delay variation for this test. We can clearly see that the delay variation increases as the percentage of errors increases. This is due to uncertainty caused by collision detection. This traffic scenario stresses the contention

channel (single ATM cell request), therefore the effect of errors in contention detection is more prominent.

Figure 5.19: Effect of Errors in Contention and Data Slots for Random Short Traffic

Figure 5.20: Effect of Channel Errors on Delay Variation

In this experiment, we have assumed uniform error distribution for every user; however, in practice, various users will exhibit different probability of errors depending on the channel condition (e.g Line of sight, distance from LMDS hub etc.).

5.6 Support of Variable Length Data Slots

In this experiment, we have considered Variable Length (VL) PDUs and AAL-5 in the upstream channel for supporting native protocols which generate variable length data units (e.g., 802.3, IP, IPv6). As shown in figure 5.4, the packet sizes generated by an Ethernet (802.3) card on a PC range from 64 bytes to 1516 bytes (Batch Poisson Traffic Model). This traffic can be transported using VL PDUs or ATM PDUs (with AAL-5 adaptation). We have considered the following assumptions in terms of overhead.

ATM cell PDU

- Payload = 48 bytes;
- MAC + ATM Overhead = 6 bytes
- FEC Parity = Shortened RS (63,53, t=5) Configured to RS (255,245, t=5)
Notation used RS (63,53) <--> RS (255,245), Code-Rate = 0.841
(according to LMDS- specific DAVIC recommendations)
- Preamble = 4 bytes, Guard Band = 1 byte
(according to LMDS- specific DAVIC recommendations)

The total size of 69 bytes becomes 80 bytes by quantizing with minislot of 16 bytes.

Variable Length PDU

For VL packets, the overhead assumptions were taken with the objective to provide almost equal coding protection to each segment. The code rates were chosen approximately as 0.841 (similar to DAVIC specifications for ATM cells). This implies that an adaptive FEC has been selected. The preamble and guardband assumptions were also considered as per DAVIC specifications. In our calculations, we have considered a preamble and guardband for each segment. This overhead can be saved in case a user has an opportunity to transmit more than one segment.

- Maximum Payload = 209 bytes;
- MAC Overhead = 6 bytes
- FEC = RS (255, 215) = 40 bytes
- Preamble = 4 bytes, Guard Band = 1 byte
- Maximum Size = 260 bytes

Figure 5.21 and Table 5.9 to 5.11 shows the relative overhead calculation and comparison for ATM PDUs and VL PDUs. The upstream bandwidth efficiencies for ATM and VL PDUs as a function of packet sizes are compared in Figure 5.22 and calculation is shown in Table 5.12. The average bandwidth efficiencies for ATM and VL PDUs (averaged over the packet size distribution of figure 5.4) are also shown in figure 5.21 (right-most pair of bars)

Table 5.9: Packet Segmentation and Overhead for Variable Length Packets

Packet Size (bytes)	Segmentation	Segment + MAC_OH	Available Standard FEC Coding (shortened if Required)	Code Rate for RS (n,k) k/n	Segment + MAC_OH + FEC_Parity (bytes)
64	64	64 + 6 = 70	RS (84, 70) <--> RS (127, 113)	0.833	84
128	128	128 + 6 = 134	RS(158, 134) <--> RS (255, 231)	0.848	134
256	209 47	209 + 6 = 215 47 + 6 = 53	RS(255, 215) RS (63,53)	0.843 0.841	255 63
512	2 * 209 94	2 * (209 + 6) = 2 * 215 94 + 6 = 100	RS (255, 215) RS(121, 100) <--> RS (127, 106)	0.843 0.826	2 * 255 121
1024	4 * 209 188	4 * (209 + 6) = 4 * 215 188 + 6 = 194	RS (255, 215) RS(234,194) <--> RS (255, 215)	0.843 0.829	4 * 255 234
1516	7 * 209 53	7 * (209 + 6) = 7 * 215 53 + 6 = 59	RS (255, 215) RS(73,59) <--> RS (127, 113)	0.843 0.808	7 * 255 73

Table 5.10: Packet Segmentation and Overhead for VL Packets contd.

Packet Size	Segment + MAC_OH+ FEC_Parity	Segment + MAC_OH+FEC_Parity + Preamble + GuardBand	Quantization by 16 bytes MiniSlot	Total Packet Size (Including Phy OH and Quantization OH)
64	84	$84 + 5 = 89$	96 (6 minislots)	96 (6 minislots)
128	134	$134 + 5 = 139$	176 (11 minislots)	176 (11 minislots)
256	255 63	$255 + 5 = 260$ $63 + 5 = 68$	272 (17 minislots) 80 (5 minislots)	352 (22 minislots)
512	2 * 255 121	2 * (255 + 5) = 2 * 260 $121 + 5 = 126$	2 * 272 (2 * 17 minislots) 128 (8 minislots)	672 (42 minislots)
1024	4 * 255 234	4 * (255 + 5) = 4 * 260 $234 + 5 = 239$	4 * 272 (4 * 17 minislots) 240 (15 minislots)	1328 (83 minislots)
1516	7 * 255 73	7 * (255 + 5) = 260 $73 + 5 = 78$	7 * 272 (7 * 17 minislots) 80 (5 minislots)	1984 (124 minislots)

Table 5.11: Packet Segmentation and Overhead for ATM Cells

Pkt Size	Segmentation	Segment + MAC_OH	FEC	Code Rate	Seg. + MAC_OH + FEC	Seg. + MAC_OH+ FEC+ Preamble + Guard Band	Quantization by 16 bytes MiniSlot	Total Packet Size (Including Phy OH and Quantization OH)
64	2 * 48	2 * (48 + 6)	RS(53,63)	0.841	2 * 64	2 * 69	2 * 80 (2 * 5 minislots)	160 (10 minislots)
128	3 * 48	3 * (48 + 6)	RS(53,63)	0.841	3 * 64	3 * 69	3 * 80 (3 * 5 minislots)	240 (15 minislots)
256	6 * 48	6 * (48 + 6)	RS(53,63)	0.841	6 * 64	6 * 69	6 * 80 (6 * 5 minislots)	480 (30 minislots)
512	11 * 48	11 * (48 + 6)	RS(53,63)	0.841	11 * 64	11 * 69	11 * 80 (11 * 5 minislots)	880 (55 minislots)
1024	22 * 48	22 * (48 + 6)	RS(53,63)	0.841	22 * 64	22 * 69	22 * 80 (22 * 5 minislots)	1760 (110 minislots)
1516	32 * 48	32 * 48	RS(53,63)	0.841	32 * 64	32 * 69	32 * 80 (32 minislots)	2560 (160 minislots)

As we notice from these calculations the Ethernet packets are segmented into smaller ATM PDUs, and the upstream bandwidth efficiency suffers because of the following reasons.

- partial filled ATM cells
- ATM header overhead
- Repeated PHY and MAC overheads for each segment.

The VL mode also requires segmentation for payloads over 209 (255 including overhead) bytes, but this level of segmentation is much less compared to the segmentation required by the ATM mode. For these reasons, the packet overheads carrying Batch Poisson packets in upstream channel is lower for VL PDUs as compared to that for ATM PDUs.

Figure 5.21: Comparison of Upstream Overheads for Carrying Batch Poisson Packets in VL PDUs vs. ATM PDUs.

Table 5.12: Bandwidth Efficiency Calculations for packet sizes for ATM and VL

Offered Packet (Bytes)	Variable PDU Efficiency	ATM PDU Efficiency
64	$64 / 96 = 0.67$	$64 / 160 = 0.4$
128	$128 / 176 = 0.727$	$128 / 240 = 0.533$
256	$256 / 352 = 0.727$	$256 / 480 = 0.533$
512	$512 / 672 = 0.762$	$512 / 880 = 0.58$
1024	$1024 / 1328 = 0.771$	$1024 / 1760 = 0.58$
1516	$1516 / 1984 = 0.764$	$1516 / 2560 = 0.59$
Avg Pkt = 368.04	Avg Efficiency $368.04 / 487.2 = 0.755$	Avg Efficiency $368.04 / 664 = 0.554$

Figure 5.22: Throughput Improvement by Variable Length Data Slots

Table 5.13: Configuration for ATM and VL PDU Experiment

<i>Simulation Parameter</i>	<i>ATM PDU</i>	<i>VL - PDU</i>
Upstream data transmission rate	5.12 Mbps	
Number of Active Stations	200	
Contention Slot Size	16 bytes	
Data Slot Size	80 bytes	Variable (max 260 bytes)
Frame Size	3 ms	
No. of max. CS per frame	120	
No. of max DS per frame	24	Variable
Size of DS: Size of CS	1:5	Variable Ratio
Length of Simulation Run	300 sec	
Length of run prior to gathering statistics	10 % of simulated time	
Contention Slot Allocation	Dynamic Data Slot Priority	Dynamic Data minislot Priority
Traffic Model	Batch Poisson	

The mean access delay plot is shown in figure 5.23. The simulation parameters for this experiment are listed in table 5.13. Clearly, VL PDUs facilitate lower delays as well as higher throughput. With the listed assumption for the upstream channel, the use of Variable length (VL) PDUs for carrying Batch Poisson Traffic results in 20% higher bandwidth efficiency as compared to the use of ATM PDUs (AAL-5). These results point to significant bandwidth efficiency advantage of VL PDUs over ATM PDUs in the upstream direction. We can observe that accommodation of VL PDUs has been accomplished in the upstream by means of minislot granularity.

Figure 5.23: Throughput Improvement by using Variable Length Packets

5.7 Comparative Results of MCNS and 802.14

These experiments were performed to compare the performance of IEEE 802.14 draft and the MCNS MAC protocol under similar assumptions. Table 5.14 lists the simulation assumptions. The simulation is performed for ATM mode of operation in both the protocols for Batch Poisson and Short Random Traffic. A single cell ATM transport is considered in MCNS, i.e. Concatenation feature of the protocol is not implemented. The major difference in the two protocols is the Contention resolution algorithm: the *ternary tree* in the IEEE 802.14 MAC protocol, and the *truncated binary exponential backoff* in the MCNS MAC protocol. Since both the protocols have piggybacking capability, we have implemented piggybacking in both the protocols. Dynamic contention slot allocation is considered in the both cases as well.

Table 5.14: Configuration for 802.14 and MCNS MAC

<i>Simulation Parameter</i>	<i>Value for IEEE 802.14 and MCNS</i>
Upstream data transmission rate	5.12 Mbps
Number of Active Stations	200
Contention Slot Size	16 bytes
Data Slot Size	80 bytes
Frame Size	3 ms
No. of max. CS per frame	120
No. of max DS per frame	24
Size of DS: Size of CS	1:5
Length of Simulation Run	300 sec
Length of run prior to gathering statistics	10 % of simulated time
Piggybacking	Yes
Transport Mode	ATM
Contention Slot Allocation	Data Slot Priority

5.7.1 Batch Poisson Traffic

Figure 5.24 shows the Mean Access Delay performance of IEEE 802.14 MAC with MCNS MAC. The mean access delay performance of the two protocols is very close for load conditions less than 0.7, this is mainly due to minislot granularity and piggybacking feature in both the protocols. MCNS shows slightly larger access delay for load greater than 0.7, because ternary tree algorithm is slightly more efficient in resolving collisions than the binary exponential algorithm at higher loads. Although both protocols are able to

maintain mean access delay less than 20 ms for Offered load less than 0.75, there is a difference of 1.2 ms in access delay at load = 0.7, and 4 ms at load = 0.8.

Figure 5.24: MCNS vs. 802.14 Batch Poisson Traffic

5.7.2 Random Short Traffic

Figure 5.25 shows the Mean Access delay performance of IEEE 802.14 MAC and MCNS MAC for Random Short Traffic. This experiment is particularly useful in stressing the contention channel and comparing the performance of Contention Resolution algorithms. We can observe from figure 5.24 that there is no difference in mean access delay for load less than 0.5. After load = 0.5, we start observing the difference in the mean access delay. There is a difference of 0.8 ms at load = 0.5 (at this point load on contention slots is 0.21), 5.4 ms at load = 0.6 (contention load is 0.33) and 11 ms at load = 0.7 (contention load 0.38). The reason again is that the ternary tree algorithm in combination of variable entry persistence, performs better than binary exponential backoff algorithm at higher load conditions. The tree algorithm is able to maintain a stable throughput even after breakdown at 0.39 but the binary exponential algorithm becomes unstable after a load greater than 0.367 on the

contention slots. Figure 5.26 and 5.28 show the distribution of access delay for IEEE 802.14 and MCNS MAC at load = 0.5. Figure 5.27 shows the coefficient of delay variation for the IEEE 802.14 and MCNS MAC protocol. Clearly IEEE 802.14 MAC provides lower delay variation due to the controlled behavior of tree algorithms.

Figure 5.25: MCNS vs. 802.14 Random Short Traffic

Figure 5.26: MCNS vs. 802.14 Random Short Traffic - CDF

Figure 5.27: Coefficient of Delay Variation MCNS vs. 802.14, Random Short Traffic

Figure 5.28: Access Delay MCNS vs. 802.14, Random Short Traffic

5.8 Comparative Results of DAVIC and 802.14

This section has been taken from [45]. The simulation parameters for this experiment are shown in table 5.15.

Table 5.15: Configuration for 802.14 and DAVIC MAC

<i>Simulation Parameter</i>	<i>IEEE 802.14</i>	<i>DAVIC</i>
Upstream data transmission rate	2 Mbps	
Number of Active Stations	200 in Random Short & Batch Poisson Model, varies between 200 - 5000 in WWW Traffic	
Contention Slot Size	16 bytes	68 bytes
Data Slot Size	64 bytes	68 bytes
Frame Size	3.072 ms	3.264 ms
No. of max. CS per frame	48	12
No. of max DS per frame	12	12
Size of DS: Size of CS	4:1	1:1
Transport Mode	ATM	

5.8.1 WWW Traffic Results

Figure 5.29 shows the mean access delay versus user population for the two protocols under dynamic ratio of contention to data slots. It can be observed that the mean-access delay for the two protocols is approximately the same for a user population size of 1500 or less. As the user population increases further, the users of 802.14 experience a lower mean access delay than the DAVIC users. This behavior is due to the smaller size contention slots in IEEE 802.14 than DAVIC (16 bytes in 802.14 and 68 bytes in DAVIC). For each contention slot in DAVIC, there can be approximately 4 contention slots in 802.14. Since

both protocols use the ternary tree CRA which breaks down at 39% load, therefore 802.14 can afford four times more contention load than DAVIC. This increases the request delay in DAVIC and consequently, the mean-access delay. Simulation results show that the offered load for 3000 users is moderate but the delay is large. This is because of the high collision multiplicity. As a result, the length of collision resolution interval increases resulting in higher queuing delays. In case of IEEE 802.14 MAC, the performance improves slightly at higher contention load by using piggybacking

Figure 5.29: Mean Access Delay Comparison under WWW Traffic for DAVIC & IEEE 802.14 (without and with PB and Dynamic CS/DS ratio)

5.8.2 Random Short Traffic Results

Under this scenario, DAVIC operates in immediate data transfer mode whereas 802.14 functions in contention-reservation mode (DAVIC can transmit a single 48 bytes data

packet in a contention slot whereas 802.14 cannot send a 48 bytes packet in 16 byte contention slot). This results in lower mean access delay in case of the DAVIC than 802.14 as illustrated in figure 5.30. The results shown for IEEE 802.14 MAC are for dynamic CS:DS ratio without piggybacking. As explained in 5.4.4, the performance of IEEE 802.14 MAC can be improved slightly by using piggybacking. The higher delays for IEEE 802.14 MAC are due to contention-reservation process, where a user has to send a request and wait for the grant. However, it should be observed that better throughput is achieved in 802.14 than in DAVIC. In a frame of 12 slots, DAVIC can use almost 5 slots (since CRA breaks down at 39% load) efficiently, whereas the IEEE 802.14 can utilize 8 to 9 data slots efficiently with a performance bound of 66.66% and 75% (with Piggybacking) respectively.

Figure 5.30: Mean Access Delay Comparison under Random Short Traffic for DAVIC & IEEE 802.14

5.8.3 Batch-Poisson Traffic

In this scenario, we allocate a fixed capacity to data and contention channel. Two third of the frame duration was allocated to the reservation channel, and one third of the frame

duration was allocated to the contention channel (therefore max. achievable capacity will be 66.66% in both cases). With this allocation, 8 data slots were assigned in a frame for both 802.14 and DAVIC. But the number of contention slots becomes 16 in the 802.14 frame and 4 in the DAVIC frame. The reason is due to the difference in the size of contention slots; 16 bytes in IEEE 802.14 and 68 bytes in DAVIC. The rationale behind this scenario was to see the effect of excess contention slots in the 802.14 protocol. Clearly, we can observe from figure 5.31 that for this type of traffic the excess contention slots do not affect the mean access delay performance. The reason for this behavior is due to the large average request size (8.3 ATM cells) and low contention traffic load. With this scenario, the ratio of CS:DS becomes 4:8 (1:2) and 16:8 (2:1) for 802.14. As described in section 5.4.5 (figure 5.12), this type of traffic has the CS:DS ratio of 1:3 and performs optimally. So a ratio of 3:9 (1:3) in DAVIC and 4:11 (~1:3) is an optimum choice. With this ratio the maximum achievable capacity for DAVIC becomes 75% (9/12) and 91.66% (11/12) for IEEE 802.14.

Figure 5.31: Mean Access Delay Comparison under Batch Poisson Traffic for DAVIC & IEEE 802.14

Conclusions and Recommendations

6.1 Summary Of the Dissertation

This dissertation presents the performance evaluation of medium access control (MAC) protocols originally designed for Hybrid Fiber Coaxial (HFC) Systems but could also find applications to LMDS networks. The work presented here was a joint study project on three candidate MAC protocols: the evolving IEEE 802.14 MAC protocol (HFC-specific), MCNS-DOCSIS MAC protocol (HFC-specific), and the DAVIC 1.2 MAC (LMDS-specific) protocol. The focus of this thesis is to implement IEEE 802.14 MAC and MCNS MAC protocols, and provide a fair comparison under similar assumptions.

The IEEE 802.14 MAC convergence draft is a simple contention-reservation based protocol with a centralized control. The simulated protocol is the frame-based implementation of the IEEE 802.14 MAC. The upstream and downstream channels are segmented into fixed size frames of constant duration. An upstream frame consists of contention minislots and data slots. The bandwidth requests are transmitted in the small contention minislots while actual data is transmitted in the larger reserved data-slots. The allocation of data slots is entirely controlled by the LMDS hub (headend) according to a scheduling discipline. The LMDS hub (headend) provides feedback and grant for the request transmitted in the *frame i* in the *frame i+1*. The boundaries between the contention and reservation

regions are dynamically changed and thus the number of contention minislots and data slots are adjusted to the workload need.

The user protocol is very simple. Whenever a user has data to transmit, he transmits a bandwidth request in one of the contention minislots specifically allocated by LMDS hub for newcomers with entry probability p . Since there is no prior arbitration in the request transmissions, the request suffers from a collision. When a collision occurs, a ternary tree contention resolution algorithm is used to resolve the contention. The LMDS hub (headend) allocates a unique Request Queue identification number (RQ) to the set of collided users in a contention minislot, and allocates three contention minislots with the same RQ number some time in the future. The user then transmits randomly in one of the three allocated contention minislots. The ternary splitting process repeats itself until all the collided users are able to transmit successfully. While the users are involved in the contention resolution process the bandwidth request size is updated each time they retransmit. The new arrivals are blocked until the current contention is resolved. Finally, when a user is able to successfully transmit his request, he waits for the bandwidth allocation. When transmitting data in the reserved slots, a user can also piggyback the bandwidth request. Figure 6.1 shows the control architecture of the implemented frame based IEEE 802.14 MAC model. The three main components at the LMDS hub (headend) arbitrate access to the channel: CSA, CRA and scheduler. The CSA specifies the boundaries between the contention and reservation regions for each frame. The scheduler deals with the allocation of reserved data slots to different users. The CRA operates on the contention minislots and defines contention-minislots for new arrivals and already collided users. CRA also provides the transmission (entry) probability for new arrivals and the unique RQ number to the set of collided users in a contention-minislot. The value of transmission probability (entry probability) for new-arrivals is calculated by estimating a new arrival population and the availability of contention minislots for the new arrivals. Both the

Scheduler and CRA compete for bandwidth in their individual sub-channel to clear up the “queues” faster. The CSA arbitrates between them by balancing the capacity so that maximum overall system performance is obtained. Similarly, CRA can also handle priorities by temporarily blocking low-priority users and serving contention minislots to the high priority users. The implementation of the MCNS protocol is also done on a frame basis by using a simple ethernet type “*truncated binary exponential backoff*” contention resolution algorithm. The implementation of this CRA is much simpler than the tree based algorithm because of its distributed nature and memoryless properties. In the frame-based MCNS protocol, the LMDS hub (headend) provides a bandwidth allocation MAP in the downstream, which specifies the reservation and contention regions. The bandwidth MAP also provides initial and maximum backoff windows to users. The CRA operates on a pure acknowledgment basis. The protocols works in the following manner. Whenever a user has data to transmit, he can either transmit data in the contention mode by concatenating minislots or he can send a bandwidth request (number of minislots required) after randomly deferring (based upon the initial backoff widow) in a contention minislot and wait for the acknowledgment. In the case where there is an absence of acknowledgment or grant in the next bandwidth MAP, a collision is assumed and the users backoff exponentially, and retransmit again after deferring. If the backoff window has been updated by the LMDS hub (headend), the user defer according to the new window. Finally, when a user is able to transmit a successful request, he waits for the bandwidth grant. The bandwidth allocation is entirely controlled by the LMDS hub (headend) Scheduler. When transmitting data, a user can piggyback bandwidth request with the data transmission. In our MAC model, the bandwidth MAP is defined by CSA and Scheduler (central) components of the LMDS hub (headend). The implementation of CRA is rather distributed. We have considered a fixed window in every bandwidth MAP with the initial backoff window of 16 (2^4) and Maximum backoff window of 256 (2^8). The implemented CRA can be made centralized by updating the window size in every MAP by estimating active population based

upon collisions and successful transmissions in the previous frame. This can provide a persistence stabilization capability to the CRA.

Figure 6.1: Control Architecture of the IEEE 802.14 MAC Model

6.2 Thesis conclusions

With the above simulated MAC models, it has been found that the minislot granularity becomes one of the key factors in the performance improvement of IEEE 802.14 MAC and MCNS MAC over the DAVIC MAC. The DAVIC MAC however relies on much larger contention slots. The additional performance improvement has been observed by the use of piggybacking in both the IEEE 802.14 and the MCNS MAC. Within IEEE 802.14 MAC and MCNS MAC, the major factor that affects the performance is the Contention Resolution Algorithm. The Ternary tree with variable entry persistence in the IEEE 802.14 and truncated binary exponential backoff in the case of the MCNS. Under similar assumptions, the mean access delay performance of ternary tree algorithm is slightly better than the truncated binary exponential backoff at higher contention load conditions. Moreover the controlled behavior of tree algorithm provides lower delay variation than binary exponential backoff algorithm for moderate to high contention load conditions. However the MCNS MAC does support the more centralized approach to contention resolution algorithm by frequently updating the backoff window, which can improve its performance. The concatenation feature of MCNS MAC can provide higher bandwidth efficiency by saving MAC and PHY overhead. Both the IEEE 802.14 MAC and MCNS MAC support variable length packets, which increases the bandwidth efficiency in the upstream. The presence of channel errors can degrade the performance of IEEE 802.14 MAC protocol, by increasing the access delay/delay variation and decreasing throughput. However strong forward error correction and time-out recovery mechanisms can prevent errors to occur and propagate respectively.

6.3 Contributions of the Dissertation

Contention Resolution Algorithms

- Implementation of unblocking “*ternary-tree*” Contention Resolution Algorithm in the context of IEEE 802.14 MAC protocol. The implemented algorithm works efficiently with the contention channel of variable capacity.
- Implementation of variable entry persistence algorithm for new arrivals by estimating newcomer population, in the context of IEEE 802.14 MAC protocol.
- Implementation of “*truncated binary exponential back-off*” algorithm in the context of MCNS MAC protocol.

Contention Slot Allocation Algorithm

- Designed a self-regulated CSA mechanism that is extremely simple to implement with the near optimum performance.
- Showed that there is an optimal ratio between Contention slots and Data slots.
- Studied the impact of piggybacking in certain traffic scenarios.

Bandwidth Scheduling

- Implementation of bandwidth scheduling and priorities for homogeneous and heterogeneous traffic conditions.

Channel Errors

- Studied the performance of protocol under noisy channel. Showed the effect of channel errors on data slots and contention slots. Implemented selective ARQ scheme for data packets in error and simulated error recovery mechanisms for the various cases of errors in the contention slots.

Variable Length Packet Support

- Studied the support of variable length packet in the both protocols and showed its performance improvement.

Traffic Models

- Implementation of various traffic models.

Framework

- Designed a framework for the access link of LMDS networks which can be used for further research.

6.4 Recommendations for further study

We have studied the implementation and performance of a medium access control protocol originally designed for HFC networks but also applicable to LMDS networks. There are many open issues which need further investigation.

Contention Slot Allocation

We have conducted preliminary study on contention slot allocations. As evident from our study, a dynamic mix of contention slots with data slots provides performance improvement. More detailed study is required to optimize these algorithms for a range of traffic conditions.

QoS Mechanisms

1) Service Priority

Various scheduling disciplines have been proposed in the context of ATM switching networks to support different Quality of service (QoS) requirements. Some of these scheduling discipline can be applicable to LMDS networks [58, 21, 50, 25]. The major difference between a switching network (mesh topology) and access network (uni-directional bus topology), is the network topology. Switching networks are based on point-to-point communication links whereas LMDS networks is a broadcast and shared medium in downstream and upstream respectively. The suitable MAC protocols for LMDS networks are based on a centralized scheme with the LMDS hub as an intelligent node with complete control over the user population.

Figure 6.2: Bandwidth Scheduling at LMDS Hub based on knowledge of requests

The three basic steps of a demand assignment based protocol are request, grant, and transmission. The responsibility of a scheduler is to provide bandwidth grant and to regulate the traffic (CAC and traffic policing). The initial request made by users is on the contention basis (piggybacking can be used in IEEE 802.14 and MCNS MAC protocols after receiving the grant). Therefore a random delay (within certain bounds) is caused between the request generation and request arrival. The uncertainty is caused by contention resolution algorithms. Therefore, the scheduler generates grants based upon the delayed and partial knowledge of the buffer status of the user. Figure 6.2 [25] shows the operation of bandwidth scheduler at LMDS hub.

2) Access Priority

The IEEE 802.14 WG is investigating and proposing the access priority mechanisms to provide priority in allocating contention slots to the high priority users. This mechanism is being proposed by using Contention Resolution Priority Profile (CRPP) in the RMS Grant Information Element. This binary vector is designed to provide various priority levels to different users (or Logical Queues) to access contention slots [28 , 46, 8].

The study of the combined effect of service, and access priority to fulfil the QoS requirements will be a natural extension to our work.

Collision Resolution Algorithm

- Generalized N-Tree Algorithm with FIFO capabilities (see section 3.1.2.9.5) for IEEE 802.14 [28].
- Optimization of splitting factor (N) in the case of packet capture for 802.14 MAC [10].
- Optimization of Backoff window in MCNS-MAC Protocol [47, 57]
- Implementation of Adaptive Polling in DAVIC MAC [43, 41]

Delay Bounds

- Calculation of the end to end delay bounds for network topology, in which communication takes place from a user in one LMDS cell to a user in other LMDS cell. This will include the access delay calculation of a LMDS uplink, with network delay caused by the backbone network and egress delay caused by downlink in a LMDS cell. This study will be based on the analytically derivation of the performance bounds for various algorithms used in the protocol.(e.g CRA, Scheduling at hub and backbone network, access priority algorithms etc.)

Downstream Modeling

- Modeling of congestion in the grant channel.
- Study of flow control mechanisms for ABR traffic when the flow of traffic is in downstream direction. This implies that feedback resource management cells will be transmitted in the upstream direction by using contention-reservation process. This situation may cause variation in the Allowable Cell Rate (ACR) parameter of the ABR source, which can cause cell loss for finite buffer sources.
- Modeling both the upstream and downstream will provide a more clear picture of the protocol behavior.

BIBLIOGRAPHY

- [1] A.A. Azzam, "High-Speed Cable Modems Including IEEE 802.14 Standards" McGraw-Hill, 1997.
- [2] ATM Forum, "Traffic Management Specification Version 4.0," ATM Forum-TM 95-0013 R8, August 1995.
- [3] B. McNeil, C. Bisdikian and D. Lin "Implementation Overview of Tree-Based Algorithm" contribution to the IEEE802.14 WG, No. 80214-96/246, November 1996.
- [4] B.T. Doshi, S. Dravida, P.D. Magil, C.A. Siller and K. Sriram "A Broadband Multiple Access Protocol for STM, ATM, and Variable Length Data Services on Hybrid Fiber-Coax Networks", Bell Labs Technical Journal, pp. 36-65, Summer 1996.
- [5] B.T. Doshi, S. Dravida, P.D. Magil, C.A. Siller and K. Sriram, "Advantages and Methods of Supporting Variable Length PDUs in Upstream and Downstream HFC Channels" contribution to the IEEE802.14 WG, No. 802.14-96/221, Sept. 17, 1996.

-
- [6] C. Bisdikian "Cable Access Beyond the Hype: On Residential Broadband Data Services over HFC Networks" IEEE communications Magazine, November 1996.
 - [7] C. Bisdikian "Performance Analysis of the Multi-slot n-ary Stack Random Access Algorithm (msSTART)" contribution to the IEEE802.14 WG, No. 802.14-96/117, May 1996.
 - [8] C. Bisdikian, "Enhancing the Collision Resolution Algorithm" contribution to the IEEE802.14 WG, No. 80214-97/121, November 1997.
 - [9] C. Bisdikian, "A Review of Random Access Algorithms" contribution to the IEEE802.14 WG, No. 80214-96/019, January 1996.
 - [10] C. Bisdikian, "The n-ary stack algorithm for the wireless random access channel" Mobile Networks and Applications, 1997, pp.89-99.
 - [11] C. Bisdikian, B.Neil, R. Norman and R. Zeisz "MLAP: A MAC Level Access Protocol for the HFC 802.14 Network" IEEE communications Magazine, pp. 114-21, March 1996.
 - [12] C. Siller, K. Sriram, "Enhanced Throughput Efficiency By Use of Dynamically Variable Request Mini-Slots in MAC Protocols for HFC Access Networks", contribution to the IEEE802.14 WG, No. 802.14-97/092, July 7, 1997.
 - [13] C.A. Siller and K. Sriram, "Implicit Frame Types for CRA implementation in the Presence of Round-Trip Propagation Delay" contribution to the IEEE802.14 WG, No. 802.14-97/035, March 10, 1997.
 - [14] C.C. Lee, R. Citta and D. Lin, "Performance Analysis of Adaptive Random Access Protocol, Part I" contribution to the IEEE802.14 WG, No. 802.14-96/070, March 1996.

-
- [15] C.C. Lee, R. Citta and D. Lin, "Performance Analysis of Adaptive Random Access Protocol, Part II" contribution to the IEEE802.14 WG, No. 802.14-96/071, March 1996.
 - [16] C-S. Wu, Po-N. Chen, and Gin-K. Ma, "An Efficient Collision Resolution Scheme for Wireless Multiple Access" P04-8, VTC '98.
 - [17] D. Bertsekas and R. Gallager "Data Networks", Prentice Hall 1992.
 - [18] D. Gingold, "Integrated Digital Services for Cable Networks" Master of Science Thesis at the MIT, Sept. 1996.
 - [19] D. Gray, "A Broadband Access System at 28 GHz", Technical paper FM96-1.1, presented at Wireless Cable Europe, London, Feb 28, 1996.
 - [20] D. Sala and J.O. Limb, "Comparison of Contention Resolution Algorithms for a Cable Modem MAC Protocol." International Zurich Seminar on Broadband Communications, pp. 83-90, Zurich (Switzerland), February 17-19, 1998
 - [21] D. Sala, and J.O. Limb, "Scheduling Disciplines for HFC Systems: What can we learn from ATM scheduling?" Proc. 1996 3rd International Workshop on Community Networking, pp. 13-18, Antwerpen (Belgium), May 23-24, 1996.
 - [22] D. Sala, J.O. Limb and S. Khaunte, "Adaptive Control Mechanism for Cable Modem MAC Protocols" in Proc. INFOCOM'98, San Francisco, March 1998.
 - [23] D.D.Falconer, G.Stamatelos, "LMCS System Architecture and Associated Research Issues", CITR Internal report, November 1996.

-
- [24] D. Goodman, R. Valenzuela, K. Gayliard, B. Ramamurthi, "Packet Reservation Multiple Access for Local Wireless Communications", *IEEE Transactions on Communications*, pp 885-889, vol 37, No. 8, August 1989
 - [25] G.M. Stamatelos, "LMDS/LMCS hub interconnection alternatives and multiple access issues", *CCBR' 98*, Ottawa, ON, June 1998.
 - [26] Gene Robinson, "Radio Transmission Options: Hub to Customer", *Wireless Broadband Conference*, 15 July, 1996.
 - [27] IEEE 802.14 WG, "Cable TV Access Method and Physical Layer Specification" Specification for HFC CATV Networks" IEEE 802.14 Cable TV Protocol Working Group, Draft 2, Revision 2, July 31, 1997 (<http://www.walkingdog.com>).
 - [28] IEEE 802.14 WG, "Cable TV Access Method and Physical Layer Specification" Specification for HFC CATV Networks" IEEE 802.14 Cable TV Protocol Working Group, Draft 3, Revision 1, April 13, 1998. [unofficial draft under revision]
 - [29] IEEE 802.14 WG, "Cable TV MAC/PHY Protocol Working Group Functional Requirements", October 19, 1994. contribution to the IEEE802.14 WG, No. 80214-94/002 (<http://www.walkingdog.com>).
 - [30] IEEE 802.14 WG, "Formal MAC Proposals" IEEE802.14 Cable TV Protocol Working Group, November 1995.
 - [31] IEEE 802.14 WG, "Media Access Control Specification for HFC CATV Networks" IEEE802.14 Cable TV Protocol Working Group, MAC Standard Draft V1.1, February 1997.
 - [32] J. Banks, J. S. Carson, B. L. Nelson, "Discrete-Event System Simulation", Prentice Hall 1996.

-
- [33] J.Bibb Cain, Dennis .N. McGregor, "A Recommended Error Control Architecture for ATM Networks with Wireless Links", pp 16-28, IEEE JSAC, Vol. 15, No.1, January 1997.
 - [34] J. Dail, M. Dajer, C-C Li, P, Magill, C. Siller, K. Sriram and N. Whitaker, "Adaptive Digital Access Protocol: A MAC Protocol for Multiservice Broadband Access Networks", IEEE Communications Magazine, pp. 104-112, March 1996.
 - [35] J. I. Capetanakis, "Tree Algorithm for Packet Broadcasting Channel" IEEE Trans. Inform. Theory, Vol. IT-25, pp. 505-515, September 1979.
 - [36] J.L. Massey, "Collision-Resolution Algorithms and Random-Access Communications", Multi-User Communications, G. Longo (ed.) [CISM Courses and Lectures Series, No. 265, pp. 73-137, Springer-Verlag, 1981].
 - [37] J.O. Limb et. al., "Performance Evaluation Process for MAC Protocols" contribution to the IEEE802.14 WG, No. 802.14-96/083R2, March 1996.
 - [38] K. Sriram, "HFC MAC Protocols with Dynamically Variable vs. Fixed Number of Request Mini-Slots: Performance and Capacity Comparisons", contribution to the IEEE802.14 WG, No. 802.14-96/120, May 1, 1996.
 - [39] L. G. Roberts, "Dynamic Allocation of Satellite Capacity Through Packet Reservation", AFIPS Conference Proc., National Computer Conference, 42, pp. 711-716, June 1973.
 - [40] L.F. Merakos and C. Bisdikian, "Delay Analysis of the n-Ary Stack Random-Access Algorithm" IEEE Trans. on Information Theory, Vol. 34, No. 5, pp. 931-942, September 1988.

-
- [41] LMDS Baseline Document no. 17, Revision 3.3, <http://www.davic.org>, December 1996.
 - [42] Local Multipoint Communication Systems (LMCS) in the 28 GHz range: policy, authorization procedures and evaluation criteria", Industry Canada, Ottawa, Canada, February 1997.
 - [43] M. Momona, "Framed Pipeline Polling for Cable TV Networks [Rev. 2.0]" IEEE802.14-95/100, September 6, 1995.
 - [44] M.T.Ali, "MAC Alternatives For LMCS/LMDS Networks", M.Eng Thesis, Carleton University, August 98.
 - [45] M.T.Ali, R.Grover, G.Stamatelos, D.Falconer "MAC Alternatives For LMCS/LMDS Networks", Submitted to ICC, 99.
 - [46] Mark D. Corner, N. Golmie, D.H. Su "A Priority Scheme for the IEEE 802.14 MAC Protocol for Hybrid Fiber-Coax Networks", Proc. INFOCOM'98, San Francisco, March 1998
 - [47] MCNS Data over cable interface specifications (DOCSIS)", SP-RFI-I02-971008, <http://www.cablemodem.com>".
 - [48] N. Golmie, D.H. Su "First Transmission Rule: Unblocking Techniques" contribution to the IEEE802.14 WG, No. 802.14/96-245, November 7, 1996.
 - [49] N. Golmie, G. Pieris, D.H. Su, S. Masson "Performance Evaluation of Contention Resolution Algorithm: Ternary-tree vs p-Persistence" contribution to the IEEE802.14 WG, No. 802.14/96-241, October 4, 1996.

-
- [50] N. Golmie, Mark D. Corner, J. Liebeherr, D.H. Su "Improving the Effectiveness of ATM Traffic Control over Hybrid Fiber-Coax Networks", Proc. GLOBECOM '97, Phoenix, November 1997.
 - [51] N. Golmie, S. Masson, G. Pieris, D.H. Su "Performance evaluation of MAC protocol components for HFC networks" Broadband Access Systems Proc. SPIE 2917, Boston, Massachusetts, 1996, pp 120-130.
 - [52] N. Golmie, Y. Saintillan, D.H. Su "Continuous vs Cluster Mode: Simulation Results" contribution to the IEEE802.14 WG, No. 802.14/97-076, May 19, 1997.
 - [53] P. Jacquet, P. Muhlethaler, and P. Robert, "Framing Protocols on Upstream Channel in CATV Networks: Asymptotic Average delay" contribution to the IEEE802.14 WG, No. IEEE 802.14-96/247, December 1996.
 - [54] Proakis, J.G., "Digital Communications", McGraw-Hill 1995.
 - [55] R. Citta, and D.Lin "Formal MAC Layer Protocol Proposal: Adaptive Random Access protocol for CATV Networks" contribution to the IEEE802.14 WG, No. 802.14/95-144, October 23, 1995.
 - [56] R. Citta, and J. Xia "Adaptive P-Persistence Algorithm with Soft Blocking for Contention Resolution" contribution to the IEEE802.14 WG, No. 802.14/97-037, March 7, 1997.
 - [57] Ronald L.Rivest, "Network Control by Bayesian Broadcast", Technical Report MIT/LCS/TM-287, MIT Lab for Computer Science, Cambridge, Massachusetts, 1985.

-
- [58] S. Golestani, "A Self-Clocked Fair Queueing Scheme for Broad-band Applications," Proceedings of the IEEE Infocom 1994, Vol. 2, Toronto, Canada, April 1994, pp. 636-646.
 - [59] S.Deng, "Empirical Model of WWW Document Arrival at Access Link", Proc. ICC'96, Dallas, Texas, USA, pp. 1797-1802.
 - [60] Scott Y. Seidel and Hamilton W. Arnold, "28 GHz Local Multipoint Distribution Service (LMDS): Strengths and Challenges", Virginia Technology Symposium, June 1995.
 - [61] Third notice of proposed rule making and supplemental tentative decision", Federal Communications Communication, FCC 95-287, Washington, D.C. July 28, 1995.
 - [62] Tim Kwok, "A Vision for Residential Broadband Services: ATM-to-the-Home", IEEE Network pp 14-28, September/October 1995.
 - [63] W.S. Ciciora, "Cable Television in United States", An Overview, Cable Television Laboratories Inc., 1995.

IMAGE EVALUATION TEST TARGET (QA-3)

APPLIED IMAGE, Inc
1653 East Main Street
Rochester, NY 14609 USA
Phone: 716/482-0300
Fax: 716/288-5989

© 1993, Applied Image, Inc., All Rights Reserved