ENHANCING THE RELIABILITY OF MEDIUM ACCESS CONTROL LEVEL WIRELESS MULTICAST

by

Vikram Shankar

A Dissertation Presented in Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy

ARIZONA STATE UNIVERSITY

May 2006

ENHANCING THE RELIABILITY OF MEDIUM ACCESS CONTROL LEVEL WIRELESS MULTICAST

by

Vikram Shankar

has been approved

January 2006

APPROVED:

_____, Chair

_____

_____

_____

Supervisory Committee

ACCEPTED:

_____

Department Chair

_____

Dean, Division of Graduate Studies

ABSTRACT

Increasing proliferation of wireless networks and their group-based applications such as multiplayer gaming and smart classroom has motivated research in reliable medium access (MAC) level multicast protocols. Existing Automatic Repeat Request (ARQ) based reliable MAC multicast protocols address different characteristics of wireless communication and differ in their throughput-efficiency and scalability. In order to understand the limits of throughput (delay) efficiency attainable by such protocols, this dissertation presents results of a comprehensive theoretical, analytical, and simulative study which explores the fundamental tradeoff between reliability and throughput (delay) taking into account media characteristics - including its broadcast nature, distance-dependent error, and multicast hidden terminal problem (MHTP). Additionally, reliable multicast protocols must deal with Feedback Implosion Problem (FIP) and the problem of increase in probability of transmission error with group size.

This research first theoretically analyzes common solutions to these problems under the assumption that only collisions cause packet corruption. It is proven that MHTP is prevented throughput-optimally only when each member of a group blocks its one-hop neighbor. A busy tone mechanism to implement optimal-blocking is proposed. Further, it is shown that some throughput-efficient FIP solutions are not reliable. Finally, retransmission error probability is reduced by prohibiting members that receive a packet correctly from successive retransmissions of that packet.

Tradeoffs between reliability, throughput and delay are investigated through analysis and modeling. The error assumption is relaxed to include packet corruption due to channel noise. It is shown that a single protocol configuration is not suitable for both delay- and

loss-tolerant applications. The models allow the prediction of throughput efficiency and delay for a desired level of reliability.

Three protocols to meet various application requirements of throughput and reliability are proposed. The Improved Leader Based Protocol supports high-throughput traffic. The Tone Based Protocol is more reliable; its novel method of using channel state information for feedback makes it scalable. Multi-channel Multicast Feedback Protocol confirms data delivery to individual members.

Results from simulations using an error model that includes distance between stations, interference, capture, and mobility are provided to characterize the performance benefits of the proposed protocols in comparison to existing protocols.

Dedicated to,

Gary and LuAnn Martz.

For your love and support.

ACKNOWLEDGMENTS

I thank my advisor, Dr. Sandeep Gupta, for his continuing support and guidance. He has not only provided useful inputs during the development of this work but he has also been a strong source of motivation and support. His unwavering determination for progress and unconventional ways of thinking have inspired the best in me.

I would like to place on record my appreciation for my committee members, Dr. Arunabha Sen, Dr. Cihan Tepedelenlioglu and Dr. Goran Konjevod. In spite of their busy schedules, they have taken the time to review my research and have provided valuable feedback.

I would like to thank my labmates at the Intelligent Mobile and Pervasive Applications and Communications Technologies (IMPACT) Laboratory. Our group meetings have been intellectually stimulating and so have our numerous cups of coffee together. In particular, I thank Valliappan Annamalai for helping with fixing bugs on several occasions. Georgios Varsamapoulous has helped with my research in many ways.

This work has been partially funded by NSF grants ANI-0073409 and ANI-0086020.

TABLE OF CONTENTS

Page

LIST OF TABLES xii

LIST OF FIGURES xiii

CHAPTER 1 INTRODUCTION 1

1. Motivation 1

2. Problems in MAC-Level Reliable Multicasting 3

2.1. Multicast Hidden Terminal Problem 3

2.2. Feedback Implosion Problem 4

2.3. Problem of Increasing Error Probability 6

3. Applications of this Research 6

3.1. Non Real Time Applications 7

3.2. Real Time Applications 7

4. Desirable Characteristics of a Multicast Protocol 8

4.1. Harness Broadcast Nature of Wireless Medium 8

4.2. Reliability 8

4.3. Fairness 8

4.4. Scalability 9

5. Overview of Results and Contributions 9

5.1. Fundamental Problems 9

5.2. Analysis and Modeling 11

5.3. Proposed Reliable MAC Multicast Protocols 11

Page

5.4. Simulations 12

5.5. Summary of Contributions 12

6. Dissertation Organization 13

CHAPTER 2 PRELIMINARIES 15

1. Network Model 15

2. Multicast Group Model 15

3. Radio Model 19

4. Capture Effect 20

5. Packet Corruption in Wireless Media 21

6. Performance Metrics 21

7. Compatability with IEEE 802.11 23

8. Hidden Terminal Problem in Multicast Communication 23

CHAPTER 3 RELATED WORK 25

1. Channel Reservation and the Hidden Terminal Problem 25

2. Multicast Protocols with Collision Avoidance 27

3. Reliable Multicast Protocols 28

3.1. Sender-Initiated Reliability 28

3.2. Receiver-Initiated Reliability 30

4. Other Reliability Considerations 31

5. Performance Studies 32

6. Multicast Semantics 33

7. A Brief overview of IEEE 802.11 MAC 34

Page

8. Capture Effect 36

CHAPTER 4 DESIGN ANALYSIS 38

1. Solutions to Multicast Hidden Terminal Problem ($\mathcal{MHTP}$) 38

2. Solutions to Feedback Implosion Problem ($\mathcal{FIP}$) 42

2.1. Positive Individual (PI) Feedback 43

2.2. Negative Individual (NI) Feedback 43

2.3. Positive Group (PG) Feedback 44

2.4. Negative Group (NG) Feedback 45

3. Mobility Tolerance 46

4. Mechanisms to Reduce Error Probabilities 46

4.1. Data Sequence Number in RTS Packet 46

4.2. Feedback Channel 48

4.3. Data Accept Policy 48

CHAPTER 5 RELIABLE MAC MULTICAST PROTOCOLS 51

1. Improved Leader Based Protocol (LBP-I) 51

1.1. Reliability Enhancements 52

1.2. Throughput Enhancements 53

2. Tone Based Protocol (TBP) 54

2.1. Protocol Description 54

2.2. Spurious NAK Problem 57

2.3. Operation in the Presence of Mobility 58

2.4. Operation in the Presence of "Pure" IEEE 802.11 Terminals 58

Page

3. Multi-channel Multicast Feedback Protocol (MMFP) 59

3.1. Throughput-Reliability Tradeoff 59

3.2. Physical Layer (PHY) Abstraction 60

3.3. Protocol Description 61

3.4. Feedback Subchannel Assignment 63

CHAPTER 6 PERFORMANCE ANALYSIS 65

1. Assumptions 65

2. Throughput Efficiency 67

2.1. Calculation of P_{SUCC} 69

2.2. Calculation of P_{IDLE} 71

2.3. Calculation of P_{COLL} 71

2.4. Calculation of P_{RERR} and P_{DERR} 72

2.5. Calculation of Numerical Upper Bound on Throughput 73

2.6. Trading Off Reliability for Throughput 73

2.7. Throughput of TBP 74

3. Expected Delay 77

3.1. Expected Delay in TBP 82

4. Reliability 83

CHAPTER 7 SIMULATION RESULTS 86

1. Reliability 89

1.1. Reliability in the Presence of Channel Errors 89

1.2. Reliability in Ad Hoc Environments 89

Page

1.3. Reliability in the Presence of Mobility 90
2. Throughput 92
2.1. Throughput of non-overlapping groups 92
2.2. Throughput of overlapping groups 92
2.3. Throughput in the presence of member mobility 93
2.4. Throughput-Reliability Tradeoff 94
3. Average Delay 96
4. Discussion 98

CHAPTER 8 CONCLUSIONS AND FUTURE WORK 100

REFERENCES 103

APPENDIX A PARAMETER VALUES USED IN SIMULATIONS 110

LIST OF TABLES

Table		Page
1.	Comparison of reliable multicast protocols.	49
2.	Table of Symbols/Notations	66
3.	Channel Holding Time	68
4.	Retransmission probabilities for a group size of 5 and BER of 10^{-6}.	83
5.	Retransmission probabilities for a group size of 25 and BER of 10^{-5}.	83
6.	Channel Holding Times	111
7.	Protocol Parameters	111
8.	Antenna and Channel Parameters	112
9.	Simulation Environment	112

LIST OF FIGURES

Figure Page

1. State diagram of a generic ARQ-based reliable MAC multicast protocol showing the three fundamental problems that must be addressed to obtain throughput-efficient reliability. 3
2. Multicast Hidden Terminal Problem in a wireless ad hoc network. Members of a multicast group are exposed to transmissions from interfering stations. 5
3. Feedback Implosion Problem: (a) Concurrent feedback results in collisions, (b) Consecutive feedback requires time proportional to number of members. 5
4. System Model. 16
5. Protocol Stack showing the services currently provided by different components of the PHY, LINK and NETWORK layers. Also shown are additional services required to implement specific protocols. 17
6. Zones of Signal Reception. 20
7. Channel Model. 22
8. One-hop blocking by the source and a representative is not sufficient to prevent MHTP. One-hop blocking may be implemented using RTS/CTS. . . . 40
9. Two-hop blocking: (a) increases reliability by preventing $\mathcal{M}HTP$, (b) increases packet delay in the network due to blocking of even non-interfering stations. 41
10. One-hop blocking by the source and all members will prevent MHTP. This solution can be implemented using Busy Tones. 42
11. Two hop blocking by source and all members prevents $\mathcal{M}HTP$ but reduces network throughput. 43

Figure Page

12. Feedback failure due to Capture Effect. 44
13. Effect of packet capture on reliability of LBP. 45
14. Reliability of NAK-only protocol compared to LBP. 45
15. Probability of error as a function of retransmission attempt. 47
16. Channel conditions are not the same at all stations. 47
17. Data accept policy. 47
18. Illustration of the Leader Based Protocol 52
19. State diagram of multicast source. 54
20. State diagram of the multicast receiver. 55
21. Illustration of the Tone Based Protocol 57
22. TBP RTS frame format. 57
23. State diagram of multicast source. 62
24. State diagram of the multicast receiver. 62
25. Illustration of the Multi-channel Multicast Feedback Protocol 63
26. Slot diagrams of a generic ARQ-based reliable multicast protocol. Feedback Time (FT) depends on whether protocol employs Individual or Group feedback. 69
27. Throughput analysis flow-diagram. 70
28. Relationship between throughput and number of group members. 74
29. Cost of reliability (parameter k) in terms of throughput for protocols requiring: (a) $O(1)$ feedback time, and (b) $O(N)$ feedback time. 75
30. For constant k, throughput of protocols: (a) with $O(1)$ feedback time increases with decrease in ratio of k to number of members, (b) with $O(N)$ feedback time is primarily determined by group size. 75

Figure Page

31. Markov chain model of retransmission probabilities. 77

32. Delay analysis flow-diagram. 78

33. Delay performance of TBP. 83

34. Number of retransmissions as a function of channel bit error rate (BER). . 84

35. Number of retransmissions as a function of group size. 85

36. Scenarios: (a) Single group (b) Overlapping groups (c) Non-overlapping groups (d) Overlapping groups. 88

37. Reliability in the presence of noise (7 dB SNR). 90

38. Reliability in the presence of noise (5 dB SNR). 90

39. Fairness of reliability in WLANs. 91

40. Reliability in Ad Hoc Networks. Results do not include losses due to queue overflows. 91

41. Saturation throughput per non-overlapping group. 91

42. Effect of station mobility on data reliability. 92

43. Throughput in an ad hoc network. 93

44. Throughput of TBP and LBP as a function of station mobility. 94

45. Throughput-Reliability tradeoff of MMFP. 95

46. Throughput of MMFP as a function of reliability. k is the minimum number of members that must receive the data correctly. 95

47. Fairness of reliability of MMFP as a function of k, the minimum number of members that must receive the data correctly: (a) standard deviation in PDR, (b) standard deviation in the absolute number of packets delivered. . 96

48. Delay increases with a drop in SNR. 97

Figure Page

49. Delay-Reliability tradeoff of MMFP. 98

50. Delay of MMFP as a function of reliability (k). 99

CHAPTER 1

INTRODUCTION

1. Motivation

Reliable multicasting benefits applications that send common information to a group of members. It is a well studied problem at the Application, Transport [11][38][70] and Network [20] layers, and is used in wired networks. Extending it to wireless networks will enable applications such as ad hoc multiplayer gaming and smart classrooms. This research investigates Automatic Repeat Request (ARQ) based reliable multicasting at the Medium Access Control (MAC) layer for shared channel wireless ad hoc networks. MAC level reliable multicasting improves the throughput of multicast applications by taking advantage of the broadcast nature of the wireless medium, increases link quality as perceived by higher layers, and has a cost benefit when compared to end-to-end recovery of the transport layer and reliable routing of the network layer.

Realizing the importance of ARQ-based reliable MAC level multicast, several protocols such as Leader Based Protocol [35], Batch Mode Multicast MAC [39] and Reliable MAC (RMAC) [55] have been proposed in the literature. An ideal reliable MAC protocol must be scalable, throughput-efficient and provide fairness of reliability. Scalability is important because the group size can be large, while fairness ensures that all members receive similar

levels of reliability. For example, a smart classroom may have upward of thirty students and all of those students must be treated fairly. Protocols proposed in the literature do not offer all the qualities of an ideal protocol and differ widely in terms of throughput efficiency, reliability, packet delay, and fairness of reliability. This research attempts to explore through a comprehensive theoretic, analytic, and simulative study the limits of reliability, throughput and delay that may be obtained by such protocols and the underlying principles of reliable multicasting that affect them.

A reliable multicast MAC must overcome three fundamental problems: (1) the Multicast Hidden Terminal Problem ($\mathcal{MHTP}$), (2) Feedback Implosion Problem ($\mathcal{FIP}$), and (3) an increase in the probability of transmission error with group size. We first theoretically investigate solutions to these fundamental problems under a limited error model that allows packet corruption only due to collisions. Through modeling and analysis, we then investigate the cost of reliability in terms of throughput and delay. The error assumptions are relaxed for the analytic study to include packet corruption due to channel noise. We identify underlying principals in the tradeoff between reliability on the one hand, and throughput and delay on the other. These principals will help make educated tradeoffs in the design of future MAC protocols. Based on results from our theoretic and analytic studies, we then propose the Improved Leader Based Protocol (LBP-I), the Tone Based Protocol (TBP), and the Multi-channel Multicast Feedback Protocol (MMFP). Finally, we undertake a simulative study that more accurately models real-world conditions by taking into account distance between stations, interference, capture, channel noise, and station mobility.

Forward Error Correction is not studied but may be applied in tandem with the ARQ-based reliability that is the focus of our research.

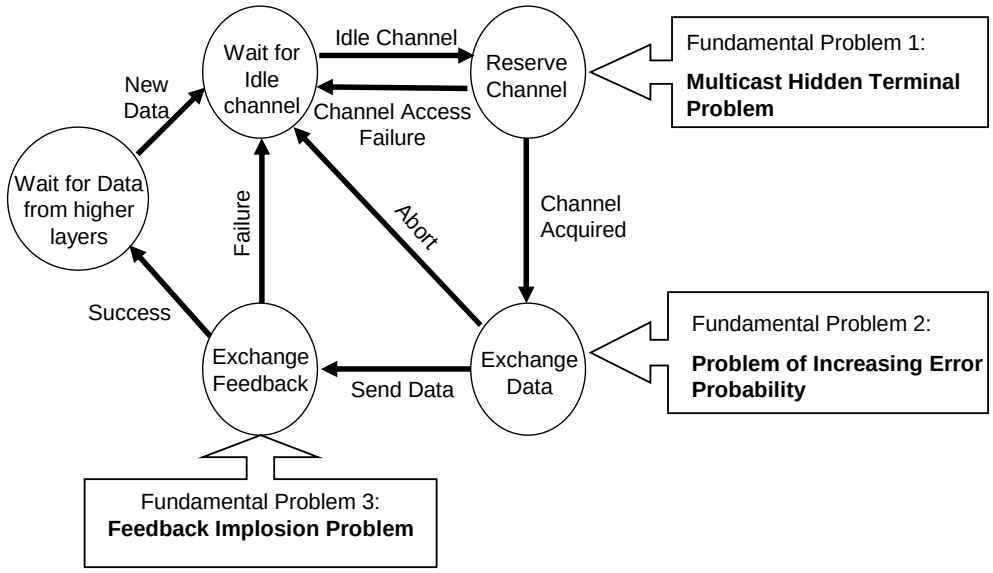

Figure 1. State diagram of a generic ARQ-based reliable MAC multicast protocol showing the three fundamental problems that must be addressed to obtain throughput-efficient reliability.

2. Problems in MAC-Level Reliable Multicasting

Fig. 1 shows a generic ARQ-based reliable multicast MAC and the three fundamental challenges it must address. The first challenge is to ensure that all stations in the vicinity of all multicast members are aware of a multicast exchange. The second challenge is to prevent feedback from multiple group members from colliding at the source. The third challenge is that the probability that at least one member receives the data in error increases with the group size.

2.1. Multicast Hidden Terminal Problem. It is important to ensure that all stations in the vicinity of all multicast group members are aware of a multicast transfer. If this is not done, members may be exposed to the Multicast Hidden Terminal Problem ($\mathcal{MHTP}$) and data will be exposed to collisions. Fig. 2 shows an example ad hoc network

containing a multicast group. Members 1, 2, 3 and 4 are exposed to non-member stations 5, 9, 8, and 7 respectively. These four non-member stations must be prevented from making transmissions during a multicast transfer.

Most protocols proposed in the literature attempt to overcome $\mathcal{M}HTP$ by reserving the channel before data is sent. Reservation is done by exchanging control packets (RTS/CTS) with each and every multicast member (e.g. Batch Mode Multicast MAC (BMMM) [39] and Reliable MAC (RMAC) [55]) or with a subset of members (e.g. Leader Based Protocol (LBP) [35], Broadcast Medium Window (BMW) [34], and [57]). The proposed solutions are either not effective or not scalable with the group size. For instance, protocols that exchange an RTS/CTS with only one representative member will reserve the channel around the source and that member while leaving the other members exposed to $\mathcal{M}HTP$. On the other hand, protocols that exchange an RTS/CTS with each and every member will not be scalable to large group sizes. Finally, protocols may be over-zealous in reserving the channel and prevent even non-interfering stations from making transmissions during a multicast exchange. For example, stations 10 and 11 in the network of Fig. 2 can safely communicate with each other without affecting transmissions by source S. A protocol that prevents those two stations from transmitting will reduce overall network throughput.

In this thesis, we investigate five common solutions to the Multicast Hidden Terminal Problem with respect to their effectiveness in preventing $\mathcal{M}HTP$, and on whether they are reserve the channel optimally.

2.2. Feedback Implosion Problem. ARQ-based reliable protocols use feedback from multicast members to indicate whether a retransmission is necessary. The challenge

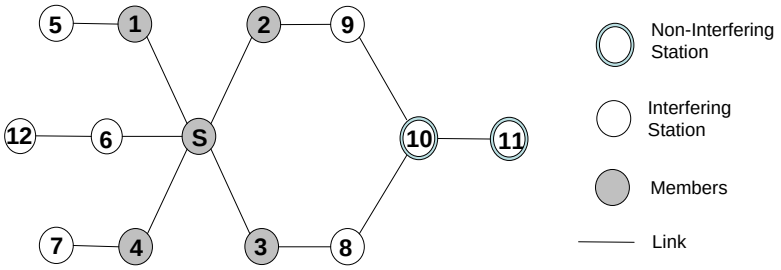

Figure 2. Multicast Hidden Terminal Problem in a wireless ad hoc network. Members of a multicast group are exposed to transmissions from interfering stations.

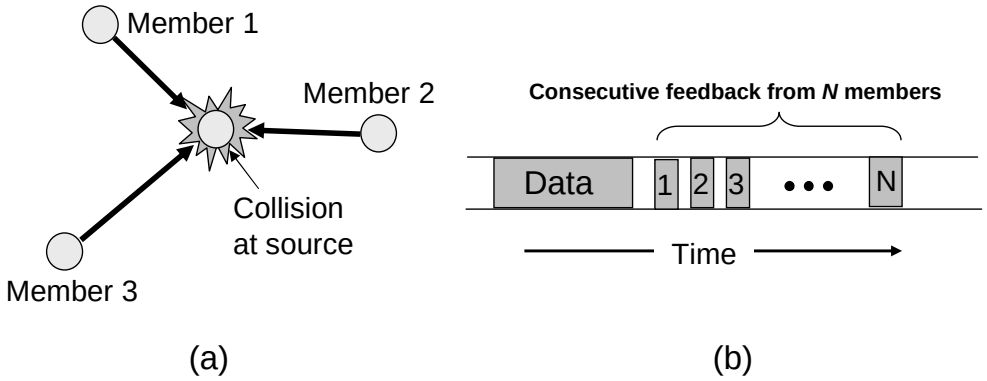

Figure 3. Feedback Implosion Problem: (a) Concurrent feedback results in collisions, (b) Consecutive feedback requires time proportional to number of members.

is to prevent feedback from group members from overwhelming or colliding at the source. This is known as the Feedback Implosion Problem ($\mathcal{FIP}$) and is illustrated in Fig. 3.

Kuri et al. [35] propose the Leader-Based, Delay-Based and Probability-Based protocols. The $\mathcal{FIP}$ avoidance mechanisms in these protocols are based on the idea that one representative station can send feedback on behalf of all other members. We show in this thesis that these protocols are not reliable under certain conditions due to the *capture effect* - the complete masking of a weak signal from a distant station by a stronger signal from a

closer station. In BMMM [39] and RMAC [55], the source individually requests feedback from each multicast member. This solution is not scalable for two reasons: (1) the delay incurred due to control packet exchange increases linearly with the group size, and (2) the probability of retransmission due to a control packet error, rather than data error, increases with the number of control packets.

We classify protocols proposed in the literature into four broad categories based on their feedback mechanisms. The categories differ in whether their recovery is sender-initiated or receiver initiated, and on whether all members send individual feedback or one member responds on behalf of the entire group. Individual feedback may allow fine-grained tracking of all members but is not scalable to large group sizes. Group feedback, on the other hand, maybe be scalable but less reliable.

We investigate the scalability and reliability of the four categories with the purpose of identifying the most suitable feedback mechanism for throughput-efficient MAC level reliable multicasting.

2.3. Problem of Increasing Error Probability. As the number of group members increases, the probability that at least one member receives the data in error increases. This results in an increase in the number of retransmissions required to convey the data correctly to all members. This in turn reduces throughput and increases the average delay faced by individual packets.

3. Applications of this Research

In general, multicast applications can be categorized into Real-Time and Non Real-Time based on their delay constraints. For most applications belonging to both categories,

it is not necessary for the link-level reliability protocol to delivery 100% of the packets. However these applications do require some minimum level of reliability for their proper functioning.

3.1. Non Real Time Applications. Applications such as multicast file transfers (e.g. in a smart classroom) and Group Support Systems have stringent requirements on reliability but can usually accommodate delays that are several orders of magnitude larger than the delay caused by recovery mechanisms. Such applications benefit from link-level recovery in several ways: (1) local-recovery is faster than end-to-end recovery, (2) local recovery consumes less time compared to end-to-end recovery, and (3) local recovery improves link quality as perceived by higher layers (this will affect routing tables).

It must be noted that even small gains at the link layer can improve overall application performance. At the MAC level, packets are dropped if corrupted. Thus higher level protocols must only deal with missing packets at known locations. Such losses are known as *erasures*. It has been shown that Forward Error Correction (FEC) can recover the entire data when any k of the n data packets are received [44][29]. Therefore, a 100% data delivery can be obtained even though the MAC itself only improves reliability sufficient to obtain a k/n delivery ratio.

3.2. Real Time Applications. Reliability involves a delay overhead and care must be taken to ensure that this overhead does not offset the gains made by a better packet delivery ratio. Nonetheless, increased multicast reliability is beneficial to Collaborative Virtual Environments (CVE) [21] such as virtual reality, distributed multiplayer games and shared whiteboards. Packet loss in such environments will not only reduce the

performance of the application and its user experience, but may also result in undesired side-effects or incorrect operation.

4. Desirable Characteristics of a Multicast Protocol

Several reliable MAC multicast protocols have been proposed in the literature. They differ widely in their objectives and as such no one protocol meets the requirements of all multicast applications. Often protocol designers have focused on reliability and have totally ignored other performance areas such as throughput and delay. In this section we discuss some of the important characteristics that an ideal protocol must have.

4.1. Harness Broadcast Nature of Wireless Medium. Any transmission in the wireless medium can be received by all stations in the vicinity of the source. Though the broadcast medium often leads to packet collisions and interference, it may also be successfully used by multicast protocols to reach several of a group's members with a single transmission. Doing so can significantly improve data throughput.

4.2. Reliability. Multicast reliability is measured as *average packet delivery ratio* (PDR) - it is the average of the number of packets received by members of a group over an observation period. By increasing the PDR, a multicast protocol delivers a greater fraction of transmitted data to the intended destination stations thereby providing a higher quality communication link to the higher level protocols and to the multicast application.

4.3. Fairness. In multicast applications, increasing PDR is important but not sufficient. *Fairness* of reliability must also exist. For example, in a smart classroom, it is

not fair if PDR is improved by increasing packet delivery to some students while ignoring others. All group members must have access to the same reliability.

Fairness is also important when we intend to additionally apply FEC based reliability schemes. An FEC encoder takes k source packets and produces $n > k$ encoded packets. A group member can obtain the entire information if it receives any k of the n transmitted packets. To take advantage of FEC, it is not sufficient to attain an *average* packet deliver ratio of $\frac{k}{n}$; rather, *each* group member must receive at least that ratio of packets.

The benefit of combining FEC with ARQ based recovery was demonstrated in Reliable Multicast Data Distribution Protocol (RMDP) [52] - an application-level protocol.

4.4. Scalability. Scalability is the ability of a multicast protocol to handle a large number of group members. Scalability is usually limited by the number of control packets generated and the delay that a data packet must face as the group size grows.

5. Overview of Results and Contributions

This dissertation has four major components: (1) fundamental problems in MAC level reliable multicasting are investigated through theoretic means, (2) the tradeoffs between reliability, throughput and delay are derived through analytic methods and modeling, (3) new reliable multicast protocols are proposed, and (4) through exhaustive simulations, the proposed protocols are compared against existing protocols.

5.1. Fundamental Problems. Five $\mathcal{MHTP}$ solutions were investigated for their effectiveness in preventing $\mathcal{MHTP}$. The solutions attempt to prevent $\mathcal{MHTP}$ by blocking potentially interfering stations from transmitting. Not blocking all potential interferers may result in $\mathcal{MHTP}$ while inadvertently blocking non-interfering stations will reduce the

network throughput. Of the five solutions investigated, we show that only the solution using one-hop blocking by the source and all members prevents $\mathcal{MHTP}$ throughput-optimally. We then propose a busy-tone mechanism to implement this solution.

The feedback strategies of reliable MAC-level multicast protocols proposed in the literature may be classified into four broad categories based on whether they use positive or negative acknowledgements, and on whether each member provides its individual feedback or one member provides feedback on behalf of the entire group. The four categories are: (1) Positive Individual, (2) Positive Group, (3) Negative Individual, and (4) Negative Group. Our study of the relative merits and demerits of the four categories concludes that Positive Group feedback is throughput-efficient but unreliable under certain conditions while Negative Group is impractical to implement. Of the remaining two, the choice of feedback mechanism depends on application requirements. Positive Individual feedback is suitable only for small group sizes but can provide the higher layers with fine-grained information on exactly which members received the data and which did not. Negative Individual feedback is scalable to large group sizes but does not provide the fine-grained feedback information.

We then consider mechanisms to reduce the probability of a retransmission, and hence improve multicast throughput, without adversely impacting reliability. We note that the probability of communication error (and hence retransmission) depends on both the number of members in the multicast group as well as the channel conditions. We show that it is possible to reduce the probability of a retransmission by excluding members that receive a data packet correctly from subsequent retransmissions for that data packet. The result is that each successive retransmission will have fewer members compared to the previous attempt and the error probability is reduced.

5.2. Analysis and Modeling. Our metrics of interest for comparing protocols are reliability, fairness and packet delay. Multicast reliability is measured as *average packet delivery ratio* (PDR) - it is the average of the number of packets received by members of a group over an observation period. *Fairness* of reliability indicates whether all members have access to a similar level of reliability. It is measured by the standard deviation (σ) in the PDR. Packet delay is the amount of time the protocol takes to complete a reliable data transfer.

We characterize, through analysis and modeling, the trade-off between reliability on the one hand, and throughput and delay on the other. Our analysis shows that one comes at the cost of the other, and hence tradeoffs made by a MAC designer will be dependent on the application - an application requiring high reliability must accept lower throughput and greater delay, while real-time applications must settle for a lower level of reliability. The analysis presented in this work will help protocol designers make the tradeoff.

5.3. Proposed Reliable MAC Multicast Protocols. We propose enhancements to an existing protocol and two new protocols based on our study of solutions to the fundamental problems. The protocols are meant to cover different application requirements. The proposed changes to the Leader Based Protocol (LBP) will enable it to work more efficiently in ad hoc networks. We then propose the Tone Based Protocol (TBP)[1] that handles $\mathcal{MHTP}$ by alerting stations in the vicinity of ALL multicast members by having each member transmit a busy tone in a narrowband subchannel separate from the data subchannel. Non-member stations wanting to make a transmission will first sense the busy tone subchannel, find it active, and back-off - thus preventing a collision. TBP uses negative feedback (NAK); it overcomes $\mathcal{FIP}$ by not depending on NAK packet information

[1] A preliminary version of this protocol, called 802.11MX, appears in [53][27].

but instead assessing channel state (through spectral power density measurements) during a designated feedback period to determine if it is busy, either due to a single NAK or collisions. Finally, we propose the Multi-channel Multicast Feedback Protocol (MMFP) that is designed to take advantage of technologies that efficiently divide a channel into a number of subchannels. Examples of such technologies include Orthogonal Frequency Division Multiplexing (OFDM) and Multi-Input Multi-Output (MIMO). The proposed protocols reduce the probability of retransmissions by prohibiting stations that have already received a packet correctly from subsequent retransmissions for that packet.

5.4. Simulations. We have simulated LBP, LBP-I, TBP, MMFP and Broadcast Medium Window (BMW) [34] using *ns-2* [1]. These are *best effort* protocols and do not guarantee packet delivery. Results indicate the importance of reliability: TBP has a PDR of over 99.9% compared to 60% by IEEE 802.11 multicast under similar channel and traffic conditions. $\mathcal{MHTP}$ must be prevented for two reasons: (1) improvement of reliability, and (2) fairness of reliability. For instance, even though LBP has an average PDR of 90% in ad hoc networks, σ values of up to 27.91% have been observed indicating that some members receive a lot more packets than others. In contrast, BMW and TBP provide fair reliability and have σ of about 0.5%. Results also indicate that the feedback mechanism adopted by LBP and LBP-I fails under certain conditions due to the capture effect. BMW, a positive acknowledgement protocol, faces high packet delays.

5.5. Summary of Contributions. The specific contributions of this work are:

1. A study of solutions to the three challenges facing MAC level reliable multicast. This study will serve as a guideline for future MAC level multicast protocol development.

2. We propose LBP-I, an improved version of LBP, TBP and MMFP. LBP-I is simple to implement while TBP provides better and fair reliability.

3. Through mathematical analysis we characterize the tradeoff between reliability of MAC multicast protocols on the one hand, and throughput efficiency and data latency on the other. To the best of our knowledge, no other such analysis has been done for multicast at the MAC level.

4. Using *ns-2* we simulated TBP, LBP, LBP-I, MMFP and BMW, and studied their performance under various conditions. Our simulation results validate our mathematical analysis.

6. Dissertation Organization

In Chapter 2 we present preliminary information that will help clarify our System Model and assumptions, and provide some basic knowledge that will aid in understanding the rest of this thesis. Related work is discussed in Chapter 3. In this chapter, we briefly describe the currently available MAC-level reliable multicast protocols and their pros and cons. We then discuss related work on mathematical models relevant to our research. We formally define the Multicast Hidden Terminal Problem ($\mathcal{MHTP}$) in Chapter 4 and prove that amongst the solutions studied, only the one that reserves the channel only in the vicinity of the source and all the members is both effective and throughput optimal. We also study solutions to the Feedback Implosion Problem ($\mathcal{FIP}$) in Chapter 4. Unlike $\mathcal{MHTP}$, there is no one single solution for $\mathcal{FIP}$ - rather, the solution depends on application requirements. Improvements to the Leader Based Protocol, and two new multicast protocols are proposed in Chapter 5. Analysis and modeling of the tradeoffs between reliability, and throughput

and delay are presented in Chapter 6. Simulation setup and exhaustive results are provided in Chapter 7. We conclude with a discussion of future work in Chapter 8.

CHAPTER 2

PRELIMINARIES

1. Network Model

Fig. 4 shows our model of multicast groups in a single channel ad hoc network. The transmission area covered by each station is called its *Coverage Area*. The set of stations in the coverage area of a station S is denoted by $\mathcal{C}_S$, and the stations are said to be *one-hop* from S. Due to the dynamic nature of the wireless medium, the coverage area of a station varies over time; we assume that this variation occurs only between transmissions and not during a transmission. We also assume that the transmission powers and receiver sensitivities are homogeneous across the entire network.

The *source* is any station that generates or forwards multicast data. Coverage areas of multiple stations can overlap and since all stations use the *same* channel, transmissions in adjoining or overlapping coverage areas will interfere with each other resulting in the hidden terminal problem [59].

2. Multicast Group Model

A distinction is made between MAC-level Multicast Groups (MMG) and Network-level Multicast Groups (NMG). An NMG may correspond to one or more MMGs at the

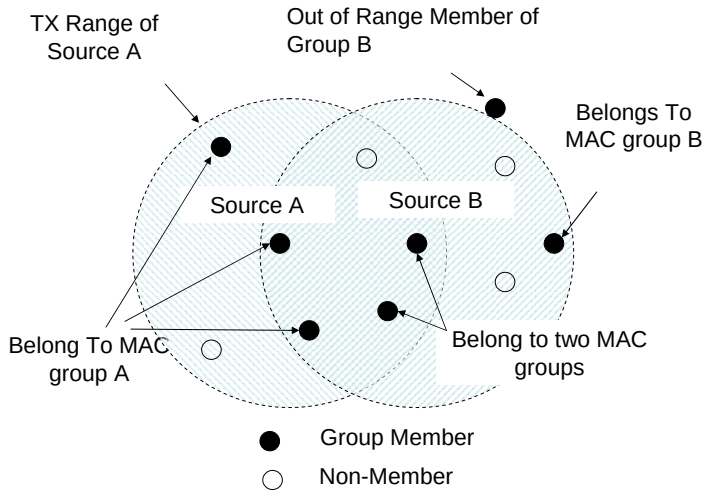

Figure 4. System Model.

MAC level. For example, stations in Fig. 4 may all belong to the same NMG even though they are in two MAC groups. Members of each MMG must be contained within the $\mathcal{C}_S$ of that MMG's source. Multiple MMGs can be co-located and contend for a single channel. Stations may belong to more than one MMG at the same time. Also, as the figure shows, a group member station may have moved out of the $\mathcal{C}_S$ of the multicast source - packet losses due to such movement is not handled at the MAC.

Fig. 5 shows the parts of the protocol stack relevant to multicasting. Group membership information is maintained by the Internet Group Management Protocol (IGMP)[8], an integral part of IP, running on each multicast agent (which is the MAC source in our case). A station joins a specific group by sending a *report* to the agent. Member stations may leave silently at any time. To avoid forwarding data when no members are present, the agent uses a *deadman timer* to periodically broadcast a membership query. To avoid the implosion problem, agents in Internet multicast neither know nor care about the number of members in a group or their identities. The agent multicasts data as long as at least one

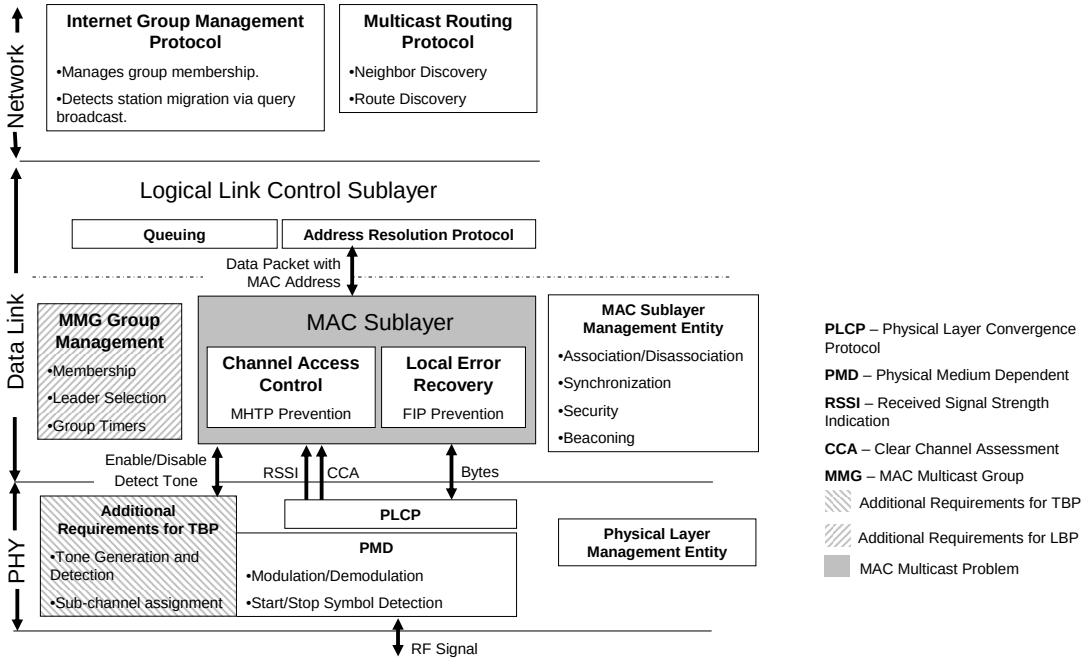

Figure 5. Protocol Stack showing the services currently provided by different components of the PHY, LINK and NETWORK layers. Also shown are additional services required to implement specific protocols.

station is present for that group. The onus of recovering data lost due to station migration rests with the receivers.

The Address Resolution Protocol [48] maps multicast IP addresses to MAC addresses. When a station enters $\mathcal{C}_S$, it broadcasts an ARP-REQUEST with the NMG IP address. Only the MMG source handling data for that NMG will respond with an ARP-RESPONSE that has the MAC multicast address. In the remainder of the paper, *group* refers to an MMG and *member* refers to a station belonging to that MMG.

The MAC Sublayer Management Entity is responsible for neighbor scanning and beaconing. It also manages station migration, association and synchronization and as such these tasks need not be duplicated by the MAC. PLCP is independent of the physical layer

technology and provides the MAC with a uniform interface to exchange an information byte stream and also provides the Clear Channel Assessment (CCA) indicating the status of physical carrier sensing. CCA can have only one of two values: IDLE or BUSY. When used over an RF radio, PLCP also provides Received Signal Strength Indication (RSSI) for each byte received. The PMD implements the actual signaling component and is responsible for modulation and start symbol detection.

When the energy in the channel is above a particular threshold, called *Energy Detect Threshold* (EDT), the PHY indicates it by setting it's PMD_ED primitive to true. When the PHY is able to lock onto a carrier signal (by detecting the psuedorandom number code), the PMD_CS primitive is set to true. PMD_CS and PMD_ED are determined independent of each other. A strong signal is indicated when both primitives are true; a weak signal is indicated when only PMD_CS is true. A noise source (such as an interfering signal) may be sufficiently strong to cause PMD_ED to be true without the PHY being able to lock onto a carrier signal. On the other hand, when both primitives are false, it indicates weak noise. The CCA function of the PHY uses these primitives to make channel state decisions. There are three common decision algorithms (or modes):

1. **CCA Mode 1:** The channel is declared busy when PMD_ED is true.
2. **CCA Mode 2:** The channel is considered busy if PMD_CS is true, irrespective of PMD_ED.
3. **CCA Mode 3:** The channel is declared busy only when both PMD_ED and PMD_CS are true.

The PHY may be switched from one mode to the other at any time.

There are several algorithms to determine whether the received energy is greater than EDT. A simple method is to take a single sample of the input and compare it against EDT. This method is unreliable because of the wide variance in noise; the channel noise may be more than EDT at the sampling moment. Reliability can be improved by taking the mean of several input samples recorded over a sampling period. The mean of a sufficiently large sampling period will iron out variability in the noise and give a clearer indication of channel state. In [49], the authors propose an algorithm that detects and uses outliers to make channel state decisions. Since the energy of noise has a wider variability than that of a digital signal, the presence of outliers almost always indicates the absence of a signal, indicating a free channel.

The multicast MAC is tasked with two responsibilities: controlling channel access to avoid $\mathcal{M}HTP$, and implement local error recovery in such a way to avoid $\mathcal{F}IP$. Fig. 5 also shows the extra resources required by the protocols we discuss in this paper.

The Received Signal Strength Indication (RSSI) is a continuously available value. It may be either expressed as an absolute signal power level, or as a ratio of the received signal power to the noise floor. The second method is preferable since it is adaptive to the environment and compensates for changes in the noise floor.

3. Radio Model

All RF signals (including data packets) are complex combinations of sine waves of different frequencies. A tone is a simple RF signal (a sine wave) modulated to the carrier frequency and has been successfully used in the Advanced Mobile Phone System (AMPS) for out-of-band signaling.

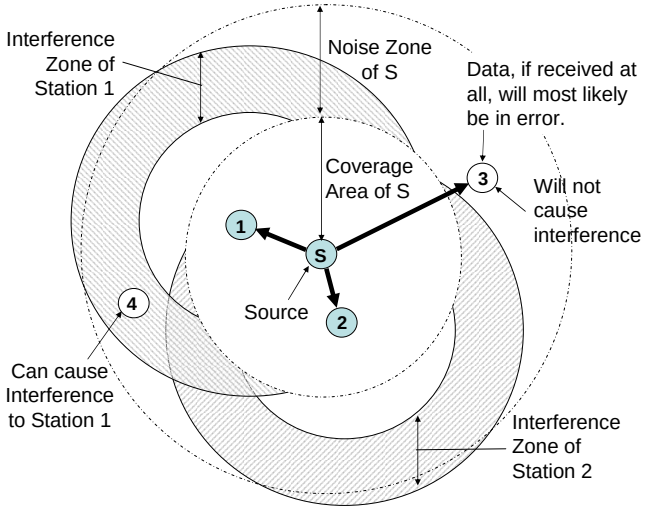

Figure 6. Zones of Signal Reception.

With reference to Fig. 6, a station is said to be in the source's *Coverage Area* and can receive its packets only if the received signal strength (P) is greater than the *Receive Threshold* (RT). Stations just beyond the coverage area may still receive packets, but with errors. Interference is caused when two transmissions occur concurrently. The *Interference Zone* is not fixed but depends on the distance between the source and destination [67]. The Signal-to-Interference Ratio (SIR) rather than the absolute signal strength determines the reliability of a received signal. Beyond the Friis zone, signal fades at $\frac{1}{d^4}$ [51] (where d is the distance of signal propagation) and as such a concurrent transmission should be no farther than 1.78 times the distance between the source and destination to cause interference [67].

4. Capture Effect

A signal is *captured*, and usable, even if it arrives concurrently with a transmission originating from a station beyond the interference zone. This complete masking of a low

power signal from a distant transmitter by a higher power signal from a closer transmitter is called *Capture Effect*. The region beyond the interference zone is called the *noise zone* and signals in this zone are too weak to detect or cause interference. An antenna is tuned to reject signals in this zone by setting the *Carrier Sense Threshold* (CST).

During physical carrier sensing, the *channel state* is considered busy (CCA = BUSY) when $P \geq CST$. Some protocols use a very low CST resulting in a wider area for which channel is busy [67].

5. Packet Corruption in Wireless Media

As shown in Fig. 7, packets are corrupted due to channel noise, signal fading with distance, and collisions. A packet is considered corrupted when one or more bits are received incorrectly as determined by the Cyclic Redundancy Check (CRC)[1]. The probability of correctly receiving a bit depends on $\frac{E_b}{N_0}$, the ratio of energy per bit to the noise per Hertz of the bandwidth. Channel noise and collisions reduce $\frac{E_b}{N_0}$ by increasing N_0, while distance reduces E_b. The MAC is oblivious to the cause of corruption and can detect it only through the CRC check. The CCITT CRC-16 used by IEEE 802.11 can detect all single and double bit errors and ensures detection of 99.998% of all possible errors.

6. Performance Metrics

Reliability: of a multicast session, measured as the *packet delivery ratio* (PDR), is the ratio of the average of the number of data packets delivered successfully to each MMG member (i.e. over a *single hop*) to the total number of data packets generated in

[1]We do not consider forward error correction (FEC) in this research. FEC only improves the bit error rate as perceived by a station and does not affect its ARQ mechanism.

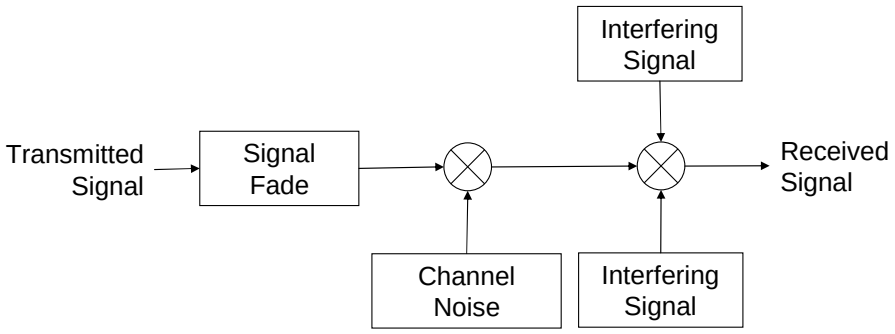

Figure 7. Channel Model.

that multicast session. Therefore PDR = $(\frac{(\sum_{i=0}^{N} D_i)/N}{D})$, where N is the number of members in the MMG, D_i is the number of packets successfully received by member i, and D is the total number of packets generated in the multicast session being observed. Since MAC level reliability does not handle losses due to members leaving, we define the duration of a multicast session as the time during which the MMG membership is constant. Semantically, a new MMG is formed when one or more members leave or join in.

Fairness of reliability is measured by the standard deviation (σ) of the PDR. A small σ implies that the number of packets received by individual members was close to the group average; a large σ signifies that some members received very good reliability while others did not.

Throughput efficiency: is the ratio of the actual sustained throughput to the channel capacity. While measuring actual throughput, only data packets are considered useful. Control packets and back-off time slots are considered overheads. Throughput efficiency allows us to represent performance independent of bandwidth availability.

Average delay: is the expected time that elapses between the moment a data packet is received at the MAC from higher layers to the time the data is successfully transmitted,

or dropped. Delay includes the control packet overhead, random back-off time, and multiple retransmission attempts. This is the per-packet delay as experienced by the source.

7. Compatability with IEEE 802.11

IEEE 802.11 has become the *de facto* standard in North America and many other regions around the world. Any new reliable multicast protocol that is developed can be deployed only if it does not affect operations of the existing IEEE 802.11 Local Area Networks while being on the same channel. In fact, it is best if the new protocol shares the same channel access mechanism; this will ensure that stations have a common understanding on who accesses the channel and for how long, irrespective of whether they have reliable multicast functionality or are "pure" stations.

Compatibility does not necessarily mean that every packet collision is avoided; it has been shown that even unicast is not collision-free [18]. The important thing is to avoid *most* common causes that affect each other's performance. Points of compatibility include: (1) a common channel access mechanism, (2) a means to inform stations that a transmission is in progress, (3) have the ability to work in both infrastructure as well as ad hoc modes. According to the IEEE 802.11 standard all stations should be within range of the *source* (access point) and not necessarily within range of each other.

8. Hidden Terminal Problem in Multicast Communication

A major cause of packet loss in multicast is collisions due to the Multicast Hidden Terminal Problem ($\mathcal{MHTP}$). Consider three stations S_A, S_B and S_C. Suppose that stations S_A and S_C are positioned such that they can each hear S_B but cannot hear each other. If S_A is transmitting to S_B, S_C will not hear it. It will sense the channel as idle and start

a transmission to S_B, resulting in a collision at S_B. In unicast, where we have a source and a single destination, the problem is known as $\mathcal{H}TP$; to avoid it, collision avoidance is performed at both source and destination [22] by exchanging Request-To-Send (RTS) and Clear-To-Send (CTS) between them [31]. Any other station that hears either control packets will refrain from making a transmission. $\mathcal{M}HTP$ involves multiple destination stations and solutions to $\mathcal{H}TP$ are not sufficient.

Let $\mathcal{G}_S$ be the set members belonging to an MMG of interest. To eliminate $\mathcal{M}HTP$ in multicast, the channel must be reserved in the vicinity of each and every member of $\mathcal{G}_S$. Since members of an MMG are always in the coverage area of the MMG's source, we have $\mathcal{G}_S \subseteq C_S$. Let $\mathcal{N}_{G_S}$ be the set of stations that do not belong to $\mathcal{G}_S$ but are in the coverage area of at least one member of $\mathcal{G}_S$. $\mathcal{M}HTP$ in multicast communication arises when stations in $\mathcal{N}_{G_S}$ are not *blocked* from transmitting while multicast data is being exchanged in $\mathcal{G}_S$. Let $\mathcal{B}_X$ be the set of stations blocked from transmitting by a method X (discussed in Section 1) in order to prevent $\mathcal{M}HTP$. Then,

Condition 1 ($\mathcal{M}HTP$ Prevention): $\mathcal{M}HTP$ is averted by method X if and only if every member of $\mathcal{N}_{G_S}$ is a member of $\mathcal{B}_X$. That is, $\mathcal{N}_{G_S} \subseteq \mathcal{B}_X$.

It is necessary and sufficient to block the members of $\mathcal{N}_{G_S}$ to avoid $\mathcal{M}HTP$. When $\mathcal{B}_X$ contains stations that are not members of $\mathcal{N}_{G_S}$, those stations are being unnecessarily blocked and transmissions that could have safely been made are prevented. This results in a reduced network throughput. Therefore,

Condition 2 (Optimality of $\mathcal{M}HTP$ Solution): We say X, a solution to $\mathcal{M}HTP$, is optimal (w.r.t. blocking) if and only if the set of blocked stations $\mathcal{B}_X$ does not include stations other than those in $\mathcal{N}_{G_S}$. That is: $\mathcal{B}_X = \mathcal{N}_{G_S}$.

CHAPTER 3

RELATED WORK

Most research on ad hoc networks have focused on reliable unicasting of data. Of the work that has been done in multicast, a majority of it is targeted solely at the network layer. As a result, these multicast solutions do not take adequate advantage of the broadcast nature of the wireless channel. Further, these protocols do not attempt to provide reliable transfer of multicast data. Examples of ad hoc multicast protocols include [24] [37] [28] [45] [6] [64]. It is our contention that the efficacy of multicast routing protocols in terms of reliability can be improved by providing local error recovery support in the underlying MAC layer.

1. Channel Reservation and the Hidden Terminal Problem

Let there be three stations A, B and C. Suppose that stations A and C are positioned such that they can each hear station B but cannot hear each other. If station A is transmitting to B, C will not hear it. It will sense the channel as idle and start a transmission to B, resulting in a collision at B. This is known as the hidden terminal problem (HTP). Necessary conditions to avoid HTP are studied in [22], and the authors conclude that collision avoidance must be performed at both the sender and the receiver. MACA [31] does this by using the Request-To-Send (RTS) and Clear-To-Send (CTS) control packets.

When station A wants to transmit to B, it sends an RTS. If station B is able to receive the data packet, it responds with a CTS. On receiving the CTS, station A transmits the data packet. Since station C is within range of B, it would have heard the CTS packet and will refrain from transmitting. The RTS/CTS exchange also prevents data packet collision resulting from stations A and C sensing the channel idle at exactly the same time and beginning their transmissions. The RTSes from A and C will collide at B. Since B will not respond with a CTS, A and C will go through a random back-off process and try again later. An alternative method is to use busy tones to indicate that the channel is in use. This idea was originally proposed in the Busy Tone Multiple Access protocol [59]. A receiver-initiated busy-tone multiple access scheme for unicast communication is studied in [63].

Several MAC protocols that have been proposed so far are based on the MACA protocol [31] proposed by P. Karn. However, Deng *et al.* [18] have shown that in spite of using RTS/CTS, the probability of packet collisions in a mobile network can be as high as 60%. Collisions are mainly due to stations that stray into a transmission zone after the RTS/CTS have been exchanged and are therefore unaware of an ongoing transmission. The authors have proposed the use of Dual Busy Tones to alert incoming stations of an ongoing transmission. Examples of MACA-based protocols that use dual busy tones include PAMAS [56] and [65]. These protocols were designed with the objective of conserving energy and better channel utilization and do not attempt reliable multicast.

The HTP prevention methods proposed for unicast cannot be directly applied to multicast communication since preventing HTP at one member does not assure HTP prevention at all other members.

2. Multicast Protocols with Collision Avoidance

Several broadcast and multicast protocols have been proposed that increase packet delivery ratio by reducing the probability of packet collisions. In Robust Broadcast [60], the source exchanges an RTS/CTS pair with one member of the multicast group. A different protocol, Robust Broadcast Reservation Protocol (RBRP) [40] reduces broadcast packet collisions by dividing up channel time into data slots and having each source reserve a data slot during a reservation phase. RBRP avoids the hidden terminal problem but suffers from high data latency under sporadic traffic conditions.

The Early Multicast Collision Detection (EMCD) protocol [43] increases multicast reliability by detecting collisions that occur due to sources starting transmission at exactly the same time. It precedes data transfer with a random length vanguard transmission and a clear channel assessment. A busy channel immediately following the vanguard transmission indicates that some other station is also transmitting. EMCD does not handle MHTP.

Leader Based Priority Ring Reliable Multicast (LPRMP) [19] was developed with the objective of reducing collisions between multicast transmissions of collocated access points. It does not support multicast in the ad hoc mode nor improve the reliability of multicast streams originating in non-AP stations.

The broadcast MAC protocol proposed in [57] requires an RTS/CTS exchange with at least one receiver during a transmission. The source sends the data even if one station responds with a CTS. Since feedback collision is possible, a busy channel during the CTS period is treated as a CTS. Though the protocol increases packet delivery it does not attempt reliable transfer.

3. Reliable Multicast Protocols

The error recovery mechanism in reliable multicast protocols fall into two broad categories: (1) sender-initiated recovery, and (2) receiver-initiated recovery. In sender-initiated recovery, the source is responsible for maintaining information on which members received a data packet correctly and which did not. The source usually gathers this information by receiving positive acknowledgement (ACK) packets from its members. The source will retransmit a data packet if it does not receive a specific number of ACKs. In receiver-initiated recovery, the group members are responsible for reliability and must implement mechanisms to detect packet loss or corruption. A member usually indicates the need for a data packet retransmission by sending a negative acknowledgement (NAK) packet to the source.

Towsley *et al.* [61][68] have shown that for multicast, a NAK based reliability scheme fares better than an ACK based scheme in terms of maximum supportable throughput. Also, since ACK-based schemes maintain state information for each receiver, such protocols are not scalable. Their work was limited to wired networks and does not account for the wireless broadcast advantage. Further, the analysis is for *selective repeat* protocols where several data packets are sent successively and only those in error are retransmitted. In contrast, the IEEE 802.11 standard specifies that the MAC sublayer should not transmit a new data packet addressed to a particular station until the previous packet to that station is successfully transmitted or discarded. Though this can be viewed as a special case of *selective repeat*, it ignores the effect of random back-offs, channel contention, capture effect and protocol specific overhead.

3.1. Sender-Initiated Reliability. In [39], the authors propose two reliable multicast protocols: the Batch Mode Multicast MAC (BMMM) and Location Aware Multicast

MAC (LAMM). In BMMM, the source does an RTS/CTS exchange with each member and sends the data even if one exchange succeeds. It then polls each member for an ACK. In LAMM, location information from a GPS is used for multicast reliability.

In the Broadcast Medium Window (BMW)[34] protocol the source addresses consecutive data packets to different members of the multicast group. Though all members will receive the data, the RTS/CTS and ACK exchange is done only with the selected member. If any other member receives the data in error, it must wait until its turn to report that error. The Round-Robin Acknowledgement and Retransmit (RRAR) [66] protocol is similar to BMW; the primary difference is that it does not use an RTS/CTS exchange. The source chooses members to respond with an ACK in a round robin scheme. Members that do not receive a particular packet must wait for their turn to request a retransmission.

RMAC [55], like the Tone Based Protocol (TBP) proposed in this work, uses Busy Tones for channel reservations. It differs in its feedback mechanism. The source sends a Multicast RTS (MRTS) with the list of group members, followed by the data. Each group member then sends an ACK tone in the order specified in the MRTS. Unlike in our protocol, Virtual Carrier Sensing is totally discarded. In principle, RMAC is very similar to BMMM and faces the same scalability problem.

In Multicast Aware MAC Protocol (MMP)[26] the source first transmits the data without an RTS/CTS exchange. The data carries the Extended Multicast Header (EMH) that specifies the order in which group members must send their ACKs. In case the source does not receive ACKs from all members, it creates and transmits an MRTS packet that lists the those members that did not respond. Each listed member responds with a CTS in the order specified by the MRTS. The data exchange is followed by each specified member sending an ACK in the same order. The MRTS/CTS exchange in the retransmission

attempts serve two purposes: (1) it avoids $\mathcal{M}HTP$, and (2) repeated failure to respond to an MRTS could indicate that the member has moved out of range.

The Broadcast Protocol with Busy Tones (BPBT) [14] uses both busy tones and RTS/CTS exchange to avoid $\mathcal{M}HTP$. The source transmits an RTS with the order in which members must provide feedback. On receiving the RTS, each member immediately starts a busy tone and then transmits a CTS in the order specified in the RTS. The data is sent if one or more CTSes are received. The ACKs are then transmitted in the same order as the CTS and the busy tones are disabled.

3.2. Receiver-Initiated Reliability. Three protocols have been proposed in [35]. The central idea behind all three is to avoid feedback implosion by choosing a representative to provide feedback on behalf of the entire group.

1. *Delay-Based Protocol (DBP):* When a data packet is received, recipient stations start a random timer. When a station's timer expires, it sends an ACK. If a station hears an ACK before its timer expires, it cancels its timer and does nothing. There is a small chance that the timers of two stations expire at the same time and an ACK collision results.

2. *Probability-Based Protocol (PBP):* When a data packet is received, recipient stations send an ACK packet with a certain response probability p. Here again it is possible that two stations send an ACK at the same time and cause a retransmission of the data.

3. *Leader-Based Protocol (LBP):* In this protocol, the base station chooses a leader from among the recipient stations. Only the leader is allowed to send an ACK. The other stations are silent if they received the data packet correctly.

In the above schemes, stations that receive the data packet in error will send a NAK to collide with and destroy the ACK. These protocols suffer from $\mathcal{MHTP}$ and feedback capture.

In [54], the authors discuss broadcast reliability in a multi-hop network; the MAC scheme they propose as part of their solution is very similar to DBP in principle is described. In addition they propose the use of priority queues to give transmission preference to time bound data packets.

Broadcast Support Multiple Access (BSMA) [58] is a simple protocol in which each member sends a CTS when it hears an RTS addressed to its group. The protocol depends on the ability of the source to overcome collisions by capturing the strongest signal (CTS packet). On receiving a CTS, the source transmits the data. Retransmission is done if a NAK is received. Since the ability to capture a signal depends on the relative positions of the transmitting members, this protocol only marginally improves reliability.

Other reliable broadcast schemes are discussed in [46][15][72][10]. Parity-based reliability [44][71] uses erasure codes to identify and recover lost information; this may be applied in tandem with the protocols proposed in this thesis.

4. Other Reliability Considerations

The probability that a data packet is received in error depends on the length of the packet. SmartPackets Inc has proposed the use of a nonlinear recurrent-feedback neural network to predict the size of a packet according to the state of the wireless channel [7]. They have reported a performance improvement of up to 100% in IEEE 802.11 unicast.

In [50], the authors show that the construction of a broadcasting schedule of minimum length in a multihop ad hoc network is NP-complete. They also show that the

problem of finding the maximum number of stations that can transmit simultaneously is also NP-complete. This indicates that reliable MAC multicast protocols that attempt to avoid collisions and the hidden terminal problem through scheduling are either not optimal or not feasible.

The trade-off between reliability and throughput stability is studied in [5]. The object of the proposed Bimodal Multicast is to sustain throughput even in the face of perturbations by a few members. The Lightweight Probabilistic Broadcast [20] trades reliability for scalability. The broadcast is gossip driven - that is each member probabilistically transmits the data to its neighbors. Both these protocols are designed for end-to-end reliability and do not account for the challenges faced by wireless MACs.

5. Performance Studies

In [4], Bianchi does a performance analysis of the IEEE 802.11 distributed coordination function using a bi-dimensional Markov Model for the random back off mechanism. The steady-state analysis provides a non-linear relationship between collision and transmission probabilities at saturation throughput. Packet corruption due to channel errors are not considered. For our *throughput analysis alone*, we develop on Bianchi's work and also account for multiple receivers and channel errors.

Delay analysis for IEEE 802.11 unicast was presented in [9]. An important assumption was that events in successive back off steps are independent; this assumption is not valid in multicast because the number of members reduces with each attempt. Throughput and blocking probabilities of IEEE 802.11 unicast in the presence of hidden terminal problem and station mobility were studied in [32]. Here again, no multicast or broadcast was studied.

The number of retransmissions required to reach M of K multicast members in a wireless LAN was studied in [41]. Two forms of time diversity were considered to improve reliability: (1) the data packet was duplicated L times and sent as one unit (2) several retransmissions of the data packet were performed. It was concluded that performance can be enhanced by keeping track of members that did not receive the data correctly and directing retransmissions only to those members. However the paper does not discuss and specific mechanisms or multicast protocols. Also, this work does not discuss the cost of reliability in terms of overhead due to channel access, feedback and the time diversity.

6. Multicast Semantics

The different multicast semantics are as follows:

1. *Best Effort Multicast:* The objective is to improve the fraction of packets delivered to group members when compared to not having any reliability at all.

2. *Semantically Reliable Multicast:* In certain applications that have read-write semantics such as maintaining cache consistency in distributed file systems, new messages will supercede older messages. This is called message *obsolescence* [47]. Only the latest message is relevant. Hence during periods of congestion, a multicast protocol can drop older messages and deliver only the latest and the communication would still be considered reliable.

3. *Probabilistically Reliable Multicast:* Multicast reliability is improved by probabilistically re-broadcasting the message. Each time a station receives a message, it retransmits that message. The retransmissions stop when the message outlives its time-to-live value. The idea behind such reliability is that a member may receive multiple copies

of a message from different stations around it and at least one of those copies will be without error. Probabilistic reliability is discussed for large-scale networks in [23]; however, concepts may be applied to ad hoc networks as well.

4. *All-or-None Multicast:* This type of multicast is useful for applications such as stock exchanges. It is difficult to implement in high-loss environments such as wireless networks and is limited to tightly couple distributed systems.

7. A Brief overview of IEEE 802.11 MAC

The IEEE 802.11 standard [16] specifies the PHY layer and the MAC sub-layer for wireless LANs. In this section we give a brief overview of the medium access in IEEE 802.11.

The network can operate either in the infrastructure mode or the ad hoc mode. In the infrastructure mode, a *Basic Service Set* (BSS) is a collection of wireless stations that are within transmission range of each other and are serviced by an *access point* (AP). Several BSSes can co-exist, each with its own AP. In the ad hoc mode, all the wireless stations are grouped together in one large *Independent Basic Service Set* (IBSS).

The standard specifies two forms of medium access: a mandatory Distributed Coordination Function (DCF) and an optional Point Coordination Function (PCF). PCF is a contention free polling scheme regulated by the access point and can be used only in infrastructure mode. DCF is a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme and can be used for both modes of operation. Carrier sensing is done both at the physical layer (*physical carrier sensing*) and at the MAC sub-layer (*virtual carrier sensing*). Physical carrier sensing is through listening to the channel for transmissions. Virtual carrier sensing is done by maintaining a *Network Allocation Vector* (NAV) at each

station that tracks the time that has to elapse for the current transmission to end. Every station that receives a packet not addressed to it, updates its NAV.

Channel access priority is regulated by using different inter-packet time periods, called the *Inter-Frame Spacing* (IFS). The shortest IFS is the Short Interframe Spacing (SIFS). High priority packets such as ACKs wait only for a SIFS period before being put on the channel. The PCF-IFS (PIFS) is slightly longer than SIFS and is used for polling. The DCF-IFS (DIFS) is used in the Distributed Coordination Function channel access and is the minimum time the channel has to be free before a station can initiate a communication.

In the basic DCF scheme, a station can transmit if the channel is free for DIFS duration. If the channel is not free or a collision occurs, the station starts a back-off timer. The timer value is randomly chosen from a contention window. The timer decrements while the channel is free and is suspended while the channel is busy. The station can transmit when the timer expires. If a collision occurs again, the station doubles its contention window size and backs off again. This is called *binary exponential back-off.*

IEEE 802.11 unicast addresses the hidden terminal problem by using the Request-To-Send (RTS) and Clear-To-Send (CTS) control packets. However the standard does not have a provision to exchange RTS/CTS packets with multiple receivers and as such its multicast and broadcast operations are susceptible to the hidden terminal problem.

Unicast reliability is provided through Automatic Repeat Request (ARQ). If a sending station does not receive an ACK from the receiver for the last data packet within a timeout period, it will retransmit the data packet. The sender will not transmit a new data packet until the last one was successfully transmitted or the retransmission limit was exceeded and that data packet was dropped from the transmission queue. The standard does not have an error-recovery mechanism for multicast and broadcast transmissions.

8. Capture Effect

In [33], packet sniffing in an IEEE 802.11 WLAN was performed to collect data on actual goodput. Two source stations were placed at different distances from a destination station; all three stations could hear each other. The RTS/CTS mechanism was turned off to avoid virtual carrier sensing. It was observed that even though both source stations transmitted about the same number of data packets, the destination station received about 15% more packets from the closer station. The authors conclude that this was due to the capture effect. Importantly, the capture occurred irrespective of which station started transmission first.

Several theoretical models of capture effect have been proposed to aid in qualitative studies and simulations. The *delay capture model* proposed in [17] captures the first packet to arrive at a station provided no other packets arrive within the capture period (T_c) of the first packet. If the first packet arrives at T_1 and packets i arrive at T_i, the first packet is captured as long as $T_i > T_1 + T_c, \quad \forall i, i \neq 1$. The *power capture model* [2] proposed for Rayleigh fading channels is the most commonly used. The packet with the maximum received power is captured if the sum of the received powers of the interfering packets is less than the capture threshold α. Let P_{max} be the power of the strongest of N signals arriving together at a station and let P_i be the power of signal i. A packet is captured when $P_{max} > \alpha \sum_{i=1, i \neq \max}^{N} P_i$. The underlying assumption is that the received signals have phase terms varying rapidly enough to result in incoherent addition of received powers. The *hybrid capture model* [13] takes into account both delay and power. The first packet arriving at a receiver is captured if $\alpha \sum_{i=2}^{N} P_i[T_1 + T_c - T_i] < T_c P_1$. These three models consider the received signal powers only for a short duration after the start of reception of the first

packet. The message in message model [42] is similar to the power capture model except that it accounts for the possibility that interfering packets may arrive at any time during a packet's reception. During reception of a packet, the PHY will search for a new packet in the signal it is currently receiving if there is a sudden jump in the signal power and the signal-to-interference ratio is greater than a preset threshold. This model has been shown to be a closer representation of the actual IEEE 802.11 capture phenomenon [62]. However, even though the PHY may capture the signal, the MAC may be in a state that does not allow it to receive the new packet. Therefore, signal capture does not necessarily imply packet capture.

We will be using a capture model mostly to study its effect on feedback mechanisms. Since feedback from multiple members will arrive at the source within short durations of each other, the power capture model is sufficient for our work.

CHAPTER 4

DESIGN ANALYSIS

An effective reliable multicast protocol must solve the three fundamental problems discussed. In this chapter, we will study the various solutions to the problems and evaluate their effectiveness. In our evaluation, we use a combination of qualitative arguments and simulation results. Figure 36 shows a sample ad hoc network used in the simulations. Details of the simulations are furnished in Chapter 7.

1. Solutions to Multicast Hidden Terminal Problem ($\mathcal{M}HTP$)

Let M_1 and M_2 be mechanisms that can block stations one and two hops away, respectively, from making any transmission. A station can use these mechanisms to prevent one and two hop neighbors from interfering with its own transmissions. For example in IEEE 802.11 M_1 is implemented using an RTS/CTS exchange. M_2 is implemented in some protocols by using a wide noise zone around a station. When used by a multicast source S, M_1 and M_2 do not have any effect on members of its $\mathcal{G}_S$. Solutions to addressing $\mathcal{M}HTP$ fall into five categories:

1. **Strategy I** - M_1 used only by Source;

2. **Strategy II** - M_1 is used by Source and one other Member (called *representative*);

3. **Strategy III** - M_2 is used only by Source;

4. **Strategy IV** - M_1 is used by Source and all Members; and

5. **Strategy V** - M_2 is used by Source and one or more Members.

We will evaluate these strategies to see if they meet the conditions for preventing $\mathcal{M}HTP$ and achieving optimal blocking.

Theorem 1. *Strategy I does not completely prevent $\mathcal{M}HTP$.*

Proof. All stations that are in the coverage area of the source belong to $\mathcal{C}_S$. Thus on receiving M_1, $\mathcal{C}_S \subseteq \mathcal{B}_{S1}$. Stations in set $\mathcal{N}_{G_S}$ are one-hop from at least one member of $\mathcal{G}_S$ and not necessarily from the source. Stations in $\mathcal{N}_{G_S}$ that are in $\mathcal{C}_S$ will be in $\mathcal{B}_{S1}$. However stations that are in $\mathcal{N}_{G_S}$ but not in $\mathcal{C}_S$ will not belong to $\mathcal{B}_{S1}$. $\mathcal{M}HTP$ will exist under the condition: $\mathcal{N}_{G_S} - \mathcal{C}_S \neq \emptyset$. □

Some existing reliable MAC protocols attempt channel reservation by exchanging an RTS/CTS with ONE representative group member [35].

Theorem 2. *Strategy II does not completely prevent $\mathcal{M}HTP$.*

Proof. Let $\mathcal{C}_R$ be the set of stations in the coverage area of the representative member (R) and are therefore affected by M_1 used by R. Since the source and R both use M_1, $\mathcal{B}_{S2} = \mathcal{C}_S \cup \mathcal{C}_R$. $\mathcal{M}HTP$ exists in this case when: $\exists x \in \mathcal{N}_{G_S} : x \notin (\mathcal{C}_S \cup \mathcal{C}_R)$. □

Fig. 8 shows that protocols using this strategy will reserve the channel around the source and the representative while exposing other members to $\mathcal{M}HTP$. In an ad hoc network, incomplete channel reservation such as that shown in the figure increases the chances of packet collision.

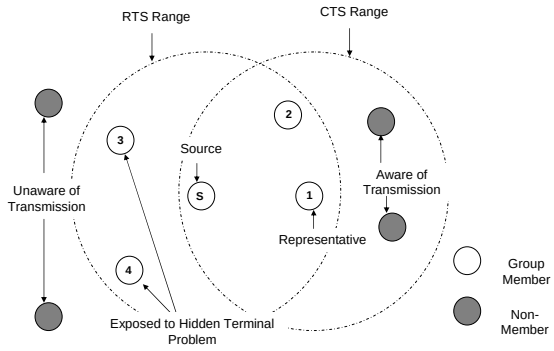

Figure 8. One-hop blocking by the source and a representative is not sufficient to prevent MHTP. One-hop blocking may be implemented using RTS/CTS.

Theorem 3. *Strategy III prevents $\mathcal{M}HTP$ but is not blocking optimal.*

Proof. Let $\mathcal{C}_1$, $\mathcal{C}_2$ ... $\mathcal{C}_m$ be set of stations that are one-hop from multicast group members $1, 2...m$ respectively. Since all group members are one hop from the source and M_2 is two-hop blocking, all stations in $\mathcal{B}_{S3} = \mathcal{C}_1 \cup \mathcal{C}_2 \cup ...\mathcal{C}_m$, will be blocked. Since all stations in $\mathcal{N}_{G_S}$ are within the coverage area of at least one group member, every member of $\mathcal{N}_{G_S}$ will belong to at least one of $\mathcal{C}_1$, $\mathcal{C}_2$ ... $\mathcal{C}_m$. Therefore, $\mathcal{N}_{G_S} \subseteq \mathcal{B}_{S3}$ and condition 1 is satisfied.

This strategy is not optimal since it also blocks non-members that are not in $\mathcal{N}_{G_S}$. Since members and non-members can co-exist in an area, stations in $\mathcal{C}_S$ need not necessarily be in $\mathcal{G}_S$. Let this set be, $\mathcal{G}_S{}' = \mathcal{C}_S - \mathcal{G}_S$. Let $\mathcal{N}_{G_S}{}'$ be the set of stations that are one hop from stations in $\mathcal{G}_S{}'$ but not of stations in $\{\mathcal{G}_S \cup \mathcal{N}_{G_S}\}$. Hence $\mathcal{N}_{G_S}{}'$ represents the set of stations two hops from the source but whose transmission will not affect the multicast communication. Since M_2 from the source is two hop, members of $\mathcal{N}_{G_S}{}'$ will also be blocked. Therefore $\mathcal{N}_{G_S}{}' \neq \emptyset \implies \mathcal{B}_{S3} \neq \mathcal{N}_{G_S}$ and the optimality condition is not satisfied. □

Fig. 9(a) compares the reliability of LBP (Strategy II) with **LBP-I** - a variant that does channel reservation through noise zones (Strategy III). In Strategy II where channel

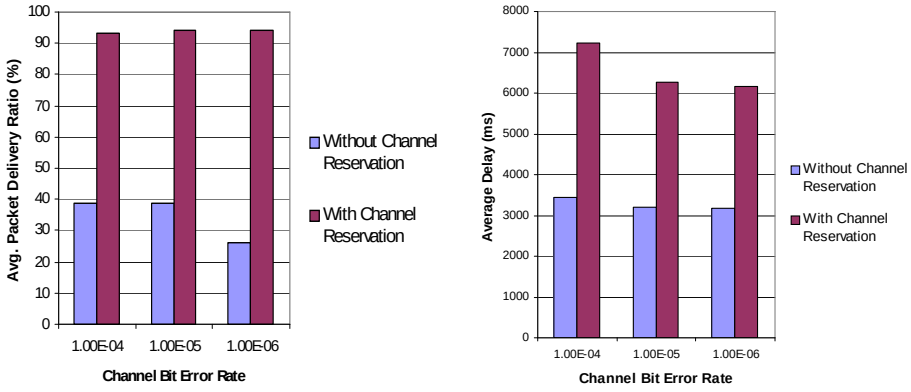

Figure 9. Two-hop blocking: (a) increases reliability by preventing $\mathcal{MHTP}$, (b) increases packet delay in the network due to blocking of even non-interfering stations.

reservation is incomplete, data retransmission is not sufficient to offset the detrimental effect of collisions. Strategy III is more reliable but, as Fig. 9(b) shows, faces increased packet delays.

Theorem 4. *Strategy IV prevents $\mathcal{MHTP}$ and is blocking optimal.*

Proof. Let $\mathcal{C} = \{\mathcal{C}_1 \cup \mathcal{C}_2 \cup ...\mathcal{C}_m\}$. Every member of $\mathcal{N}_{G_S}$ is also one hop from a group member and hence $\mathcal{N}_{G_S} \subseteq \mathcal{C}$. Since every multicast member employs M_1, all stations in $\mathcal{C}$ are blocked. Thus $\mathcal{N}_{G_S} \subseteq \mathcal{B}_{S4}$, and condition 1 is satisfied.

For optimality, stations in $\mathcal{N}_{G_S}{}'$ must not belong to $\mathcal{B}_{S4}$. All one hop neighbors of $\mathcal{C}$ must be in either one of $\mathcal{N}_{G_S}$ or $\mathcal{N}_{G_S}{}'$, but not both. M_1 is used only by members of $\mathcal{G}_S$ and hence only members of $\mathcal{N}_{G_S}$ will be blocked. Since no member of $\mathcal{G}_S{}'$ uses M_1, no member of $\mathcal{N}_{G_S}{}'$ is blocked. Therefore $\mathcal{B}_{S4} = \mathcal{N}_{G_S}$, and condition 2 is satisfied. □

Fig. 10 illustrates the effectiveness of Strategy IV in handling $\mathcal{MHTP}$. In this case, M_1 at each member is implemented as a Busy Tone.

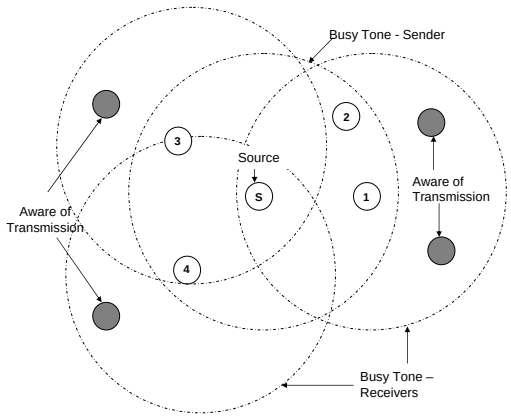

Figure 10. One-hop blocking by the source and all members will prevent MHTP. This solution can be implemented using Busy Tones.

Theorem 5. *Strategy V prevents $\mathcal{M}HTP$ but is not blocking optimal.*

Proof. We know that this strategy addresses $\mathcal{M}HTP$ because it is an extension of Strategy III discussed earlier. It is also not optimal because of the same reasons as Strategy III. □

Fig. 11 illustrates the use of M_2. Here M_2 is implemented as the noise zone generated by any packet transmission. Even though stations two hops away can safely make transmissions, they are disallowed from doing so. For instance, transmissions C2, C3, C4 and C5 are prohibited even though they would not have affected transmission C1.

2. Solutions to Feedback Implosion Problem ($\mathcal{F}IP$)

Feedback such as ACK, NAK, NCTS or CTS (or a combination of them) is needed to inform the source whether a packet transmission was successful. Care must be taken to ensure that feedback control packets do not collide at the source. Feedback strategies fall under four broad categories:

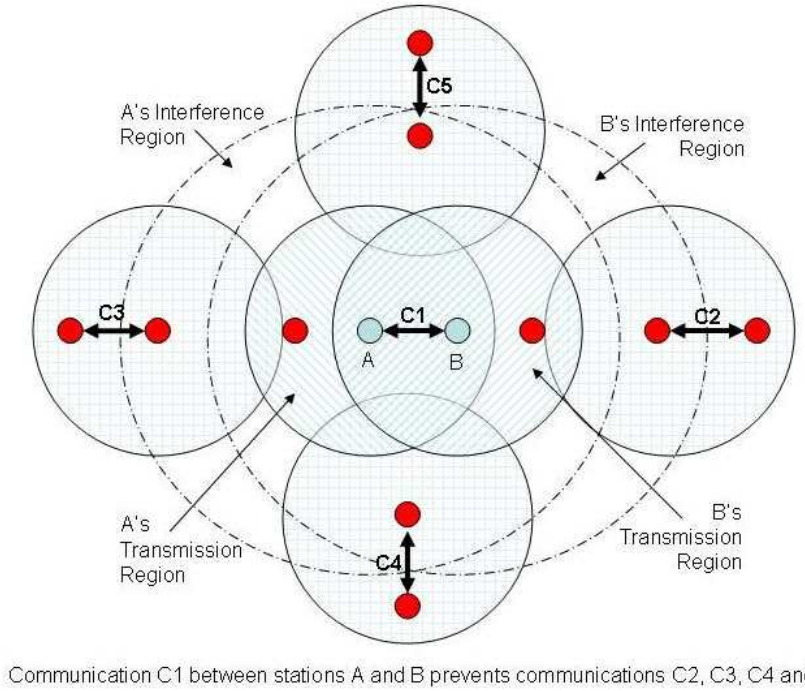

Figure 11. Two hop blocking by source and all members prevents $\mathcal{MHTP}$ but reduces network throughput.

2.1. Positive Individual (PI) Feedback. A time slot is assigned to each member during which it can send its feedback. This approach has two problems: (1) slots must be assigned to a potentially changing set of receivers (due to mobility); (2) the approach is not scalable - if there are N group members, N RTS/CTS/ACK exchanges will take place resulting in an O(N) overhead. Further, the susceptibility to retransmission increases because corruption of any of the O(N) control packets will result in a retransmission. Examples of protocols using this method are BMMM [39] and RMAC [55].

2.2. Negative Individual (NI) Feedback. There are two possibilities here. First, like in the Positive Individual method, members can each respond through polling by the source. As earlier, this method is not scalable. In the second scheme, members concurrently transmit their NAKs. A NAK or any garbled packet (due to collisions) will

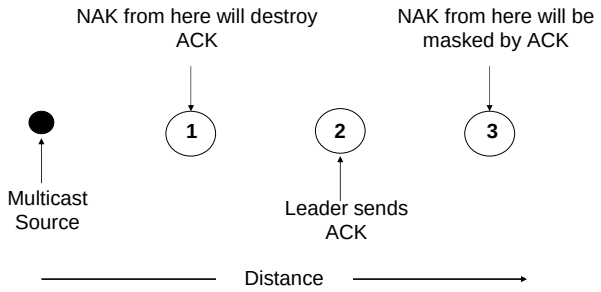

Figure 12. Feedback failure due to Capture Effect.

trigger a retransmission. We will call a protocol that uses this method as NAK-Only Protocol (**NOP**). Since, irrespective of the group size, all members respond in the same slot, the overhead is always O(1) making it a scalable solution.

2.3. Positive Group (PG) Feedback. One member of the group is chosen to provide the positive feedback on behalf of the entire group. When any other member receives the data in error, it will try to destroy the ACK by concurrently transmitting a NAK. LBP, DBP and PBP [35] are based on this method. In LBP, the source explicitly chooses the representative (called *Leader*). In DBP each member chooses a random delay and the one with the shortest delay is the representative. In PBP, members independently decide to become a representative based on a certain probability.

There are several problems with this approach. First, it is necessary for all members to be within transmission range of each other in order to coordinate feedback. This places restrictions on the maximum coverage area of a cell. Second, feedback collision is still possible. Finally, *capture effect*, can adversely affect reliability. This is a major problem in wireless networks because the received signal strength falls off *at least* as an inverse square function of the distance between transmitter and receiver. In Fig. 12, Station 2 (Leader) is

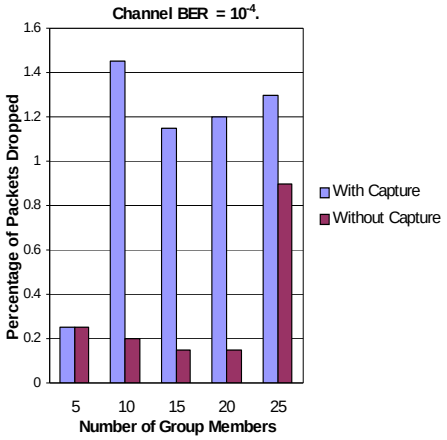

Figure 13. Effect of packet capture on reliability of LBP.

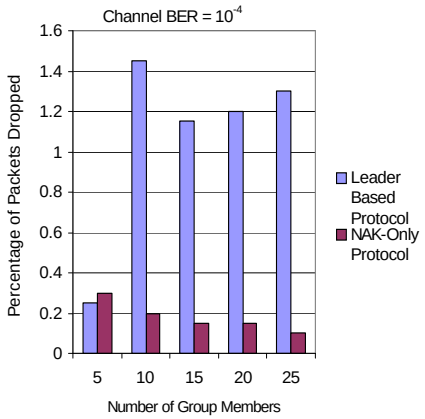

Figure 14. Reliability of NAK-only protocol compared to LBP.

responsible for sending the ACK. Though a NAK from Station 1 will corrupt the ACK, that from Station 3 might be completely overpowered by the ACK. The result is that the source erroneously concludes that the data transmission was successful. Fig. 13 shows simulation results illustrating the effect of capture on multicast reliability of LBP. The results are compared with an imaginary version of LBP in which capture effect is ignored. In reality, capture effect cannot be ignored.

2.4. Negative Group (NG) Feedback. Only one of the members that received the data in error will transmit a negative feedback. Here again, we can follow either a leader-based, probability-based or delay-based approach. We will refer to these protocols as **NLBP, NPBP** and **NDBP** respectively. Since we do not know *a priori* which or how many members will receive a data packet in error, it is difficult to select a representative. In case of NPBP, if none of the members transmit a NAK, the source will mistake it to mean a successful transmission. This also occurs when NAKs collide in NDBP or when the leader does not belong to the set of members with erroneous data in NLBP.

3. Mobility Tolerance

There are two types of mobility of interest: (1) members moving in and out of the source's coverage area, and (2) non-members moving in and out of the vicinity of multicast members.

In the first case, the throughput of a protocol that requires feedback from each individual member will be affected. In such protocols, the source will continue to perform retransmissions until it realizes that the member has left or until retransmission limit is reached.

In the second case, if incoming non-members are not alerted to a multicast exchange in progress, packet collisions will occur. These collisions will affect both reliability as well as throughput.

4. Mechanisms to Reduce Error Probabilities

Communication error probability depends on both channel conditions as well as number of stations involved in the data exchange. More the number of stations, higher is the likelihood that at least one station receives the data in error. Fig. 15 shows that when all stations are involved in all retransmission attempts of a data packet, the error probability is constantly high. On the other hand, if we prohibit stations that have already received the data correctly from subsequent retransmissions of that packet, the error probability significantly reduces with each attempt. A detailed analysis is presented in Chapter 6.

4.1. Data Sequence Number in RTS Packet. In order to implement the above, a station must know whether the packet it is receiving is a new packet or one that it already

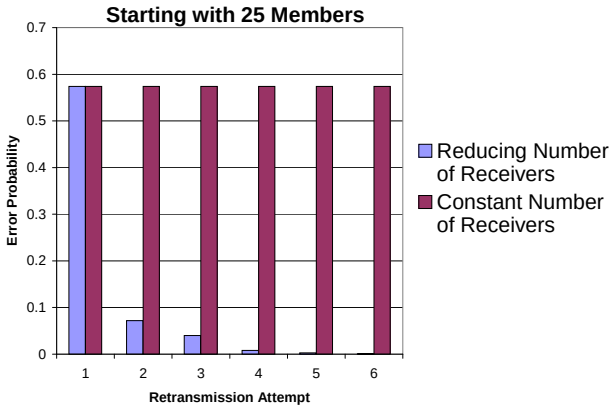

Figure 15. Probability of error as a function of retransmission attempt.

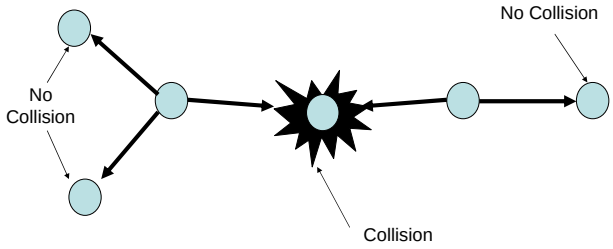

Figure 16. Channel conditions are not the same at all stations.

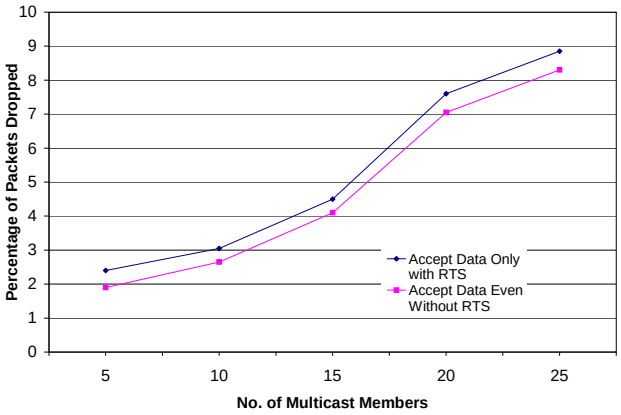

Figure 17. Data accept policy.

has - this information cannot be obtained from the packet if it is corrupted. A solution is to carry the data sequence number in the RTS packet. Due to its small size, the probability of an RTS being received in error is low. When the RTS carries the data sequence number, stations can determine whether they already have that data or not. They participate in the ensuing retransmission only if they don't already have that data.

4.2. Feedback Channel. In spite of channel reservation, collisions can still occur; for instance, when two stations sense the channel free at the same time and begin transmitting. An important observation, shown in Fig. 16, is that because of the spatial distribution of multicast receivers, *packet collisions may occur at some receivers but not at others*. When packets collide at a station, sending a NAK on the data channel will affect the reception of the data at other stations. To avoid this, feedback (such as NAK) must be sent in a subchannel separate from the data subchannel. Sending feedback on a separate channel improves throughput by reducing retransmission probability since stations that receive the data correctly will be eliminated from subsequent retransmissions. This performance enhancement does not come at extra cost as technologies such as OFDM which efficiently divide a channel into subchannels are already part of the IEEE 802.11 *a* and *g* standards.

4.3. Data Accept Policy. Fig. 17 shows that performance can be improved if protocols accept data even if they didn't receive the RTS correctly. Accepting data even without the RTS increases the number of members that received the data correctly and in turn will reduce the probability of retransmission.

Table 1 summarizes various protocols in terms of the strategies discussed in this section.

Table 1. Comparison of reliable multicast protocols.

Protocol	$\mathcal{MHTP}$ Solutions					$\mathcal{FIP}$ Solutions				Scalable	Reliability Affected By	Throughput Affected By
	I	II	III	IV	V	PI	NI	PG	NG			
BMMM [39]				•		•				O(N)	–	MM
MMP [26]				•		•				O(N)	–	MM
BMW [34]		•				•	•			O(N)	NM	NM
RBRP [40]		•				•	•			O(N)	NM	NM
LBP [35]		•						•		O(1)	NM, CE	MM, CE
DBP [35]		•						•		O(1)	NM, CE	CE
PBP [35]		•						•		O(1)	NM, CE	CE
LBP-I [page 41]					•			•		O(1)	CE	MM, CE
RMAC [55]				•		•				O(N)	–	MM
BPBT [14]				•		•				O(N)	–	MM
TBP [Sec. 2]				•			•			O(1)	–	–
Tang *et al*[57]		•						•		O(1)	NM	NM
Sheu *et al*[54]		•						•		O(1)	NM, CE	CE
NLBP [page 45]	•								•	O(1)	NM	MM
NDBP [page 45]	•								•	O(1)	NM	–
NPBP [page 45]	•								•	O(1)	NM	–
MM - Member Mobility NM - Non Member (Static or Mobile) CE - Capture Effect												

We developed a protocol based on the design principals outlined. To avoid $\mathcal{MHTP}$, we implemented Strategy IV using busy tones. This alerts only stations in the vicinity of multicast traffic, thereby reducing collisions without unnecessarily reducing network throughput. We chose Negative Individual feedback. This method is scalable and its performance is not affected by capture effect. The NAK-Only Protocol that uses NI provides better packet delivery ratio compared to LBP that uses positive group feedback (Fig. 14). We also incorporated the reliability enhancing mechanisms discussed in this section.

CHAPTER 5

RELIABLE MAC MULTICAST PROTOCOLS

In this chapter we first propose improvements to the Leader Based Protocol (LBP) [35] that will improve its throughput in wireless local area networks and reliability in ad hoc networks. The improved protocol, LBP-I, is simple to implement and provides higher reliability compared to IEEE 802.11 multicast. We then propose the Tone Based Protocol (TBP) and Mult-channel Multicast Feedback Protocol (MMFP). TBP is reliable and scalable to large group sizes. MMFP is suitable for multicast applications that have small groups but require finegrained information on exactly which members received the data correctly and which did not.

1. Improved Leader Based Protocol (LBP-I)

LBP attempts to avoid $\mathcal{M}HTP$ by using one-hop blocking by the source and a representative method (called *Leader*). The source transmits an RTS and the Leader responds with a CTS. This solution was shown to be ineffective in Chapter 4. Also, in order to avoid $\mathcal{F}IP$, only the Leader responds with an ACK on behalf of the entire group. This is the Positive Group Feedback discussed in Chapter 4 and was shown to suffer feedback unreliability due to the capture effect.

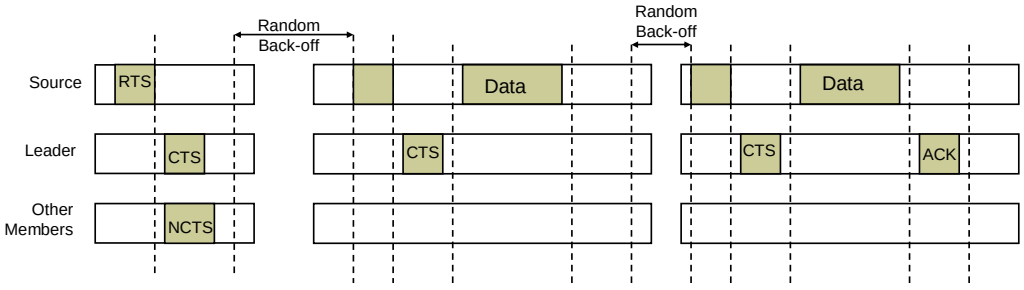

Figure 18. Illustration of the Leader Based Protocol

Fig. 18 illustrates the functioning of the protocol. The source sends an RTS to probe if the group is ready to receive a multicast data packet. The leader sends a CTS, while the other members stay quiet if they are ready. If not ready, a member sends an NCTS to destroy the CTS from the leader. If the leader is not ready, it abstains from sending the CTS. The source will transmit the data only if it receives the CTS. The leader sends an ACK while the other members do nothing if they received the data correctly. If any member, including the source, receives the data in error, it sends a NAK.

The reliability and throughput of LBP can be improved in several ways. Its main weaknesses are exposure to $\mathcal{M}HTP$, unreliable feedback due to capture effect, inefficient group management, and intolerance of group management control packet loss.

1.1. Reliability Enhancements. Exposure to $\mathcal{M}HTP$ can be reduced, without major changes to LBP, by using Strategy V discussed in Chapter 4. Mechanism M_2 is implemented by setting Carrier Sense Threshold in all receivers such that the noise region is exactly twice the coverage area of a station, and by interpreting the channel busy if a signal is above CST. This in effect results in wo-hop blocking by the source and a representative

member. It was shown in Chapter 4 that this solution prevents $\mathcal{M}HTP$ but is not blocking-optimal.

1.2. Throughput Enhancements. In LBP, the first station to join a group is chosen as leader. The leader will attempt to send a *Leave* message before quitting the group, but that message may not reach the source. When several attempts of sending an RTS fail, the source assumes that the leader has left quietly and stops forwarding multicast data to that group. When other members do not receive any data for a specified period, they re-subscribe to the group by sending *Join* messages. Any multicast data arriving between the time the source stops forwarding and the time the stations re-subscribe will be dropped. There are two drawbacks of this method: (1) control packets such as Join, Leave and the leader selection packet are themselves prone to corruption, making group management unreliable, (2) in ad hoc networks, RTS packets may face repeated collisions due to which the leader may not respond and the source erroneously concludes that the leader has left the group. As a result, throughput is wasted in forming and re-forming groups.

We propose a modification that will make LBP more robust to control packet loss. Unlike an NMG, the MMG group is not expected to be very large. The source can store the list of stations that sent it a Join message for a given group. This list need not accurately reflect the membership of the group as some Join messages may be lost. The source arbitrarily selects one station from this list to be leader and instead of informing that station alone, it piggybacks the leader information in every RTS packet. When the specified leader is non-responsive, the source picks a different member from its list to be leader and continues forwarding the data.

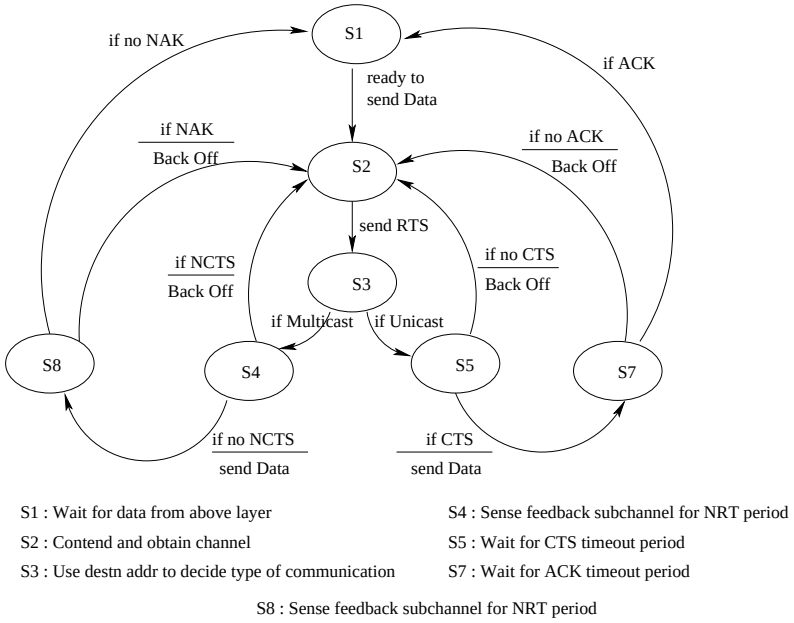

Figure 19. State diagram of multicast source.

2. Tone Based Protocol (TBP)

We developed the TBP protocol based on the lessons learnt from Chapter 4. TBP prevents $\mathcal{MHTP}$ by reserving the channel in the vicinity of all stations using Dual Busy Tones to indicate that a transmission is in progress. The protocol uses only negative feedback and overcomes the $\mathcal{FIP}$ by using the state of the channel rather than the NAK packets to indicate a negative acknowledgement. It uses sequence numbers in the RTS to allow multicast receivers to decide whether they will take part in a retransmission or will abstain if they already have the data.

2.1. Protocol Description. A channel is divided into three subchannels - the data, feedback, and signaling (busy tone) subchannels. Unicast operation is the same as IEEE 802.11 MAC. Multicast is as follows:

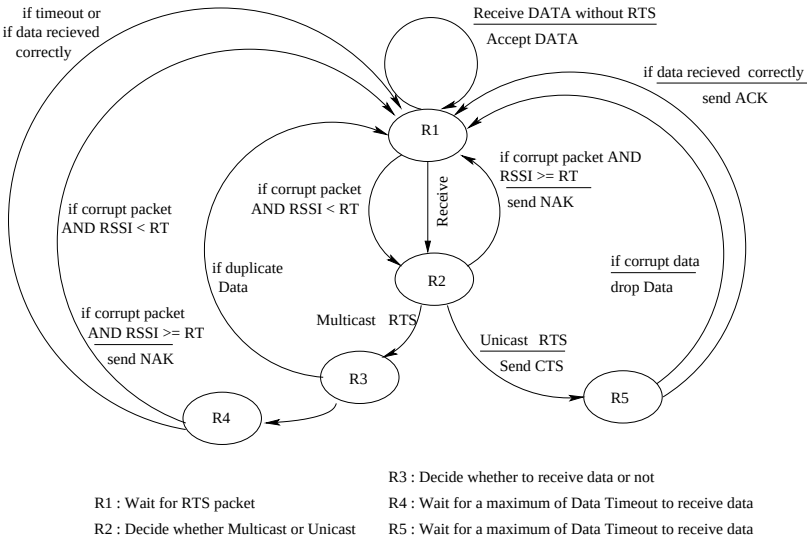

Figure 20. State diagram of the multicast receiver.

2.1.1. *Multicast Source.* When a source is ready to transmit (state S1 of Fig. 19), it senses the channel (all three subchannels) for a random time. If the channel is free it sends an RTS and listens on the feedback subchannel for an NCTS (states $S2 \rightarrow S3 \rightarrow S4$). The purpose of the RTS is to probe the members on the safety of transmitting. An RTS collision is less expensive than data collision. If no NCTS is sensed, the source begins simultaneously transmitting the data packet and busy tone in their respective subchannels (states $S4 \rightarrow S8$). At the end of the packet transmission, the source senses the feedback subchannel for a NAK. If no NAK is sensed, the sender assumes that the data transmission was successful(states $S8 \rightarrow S1$). In case of a NAK, the source must re-acquire the channel by sending an RTS. In either case it disables the busy tone.

2.1.2. *Multicast Receivers.* A received packet can be of four types: (1) multicast RTS, (2) unicast RTS, (3) some other packet, and (4) undeterminable erroneous packet. On receiving a multicast RTS, a member ignores the data if the data sequence number in the RTS indicates a packet it has (states $R2 \rightarrow R3 \rightarrow R1$ of Fig. 20). Otherwise, it prepares

to receive the data packet by transmitting a busy tone in the busy-tone subchannel (state $R2 \rightarrow R3 \rightarrow R4$), and starts a *Receive Timer*. If the data packet is not received before the Timer expires, it means that some other member has sent an NCTS; busy tone transmission is terminated. If the data packet is received without error, the packet is sent to the higher layers ($R5 \rightarrow R1$). If the packet is received with errors, the packet is dropped and a NAK is transmitted(states $R4 \rightarrow R1$). In either case, the busy tone is terminated at the end of the data reception. The working of the protocol is illustrated in Fig. 21.

If a member receives a packet and cannot determine it to be an RTS, it sends an NCTS in the feedback subchannel ($R1 \rightarrow R2 \rightarrow R1$). Since the feedback is on a separate subchannel, it does not affect the packet's reception by other stations. A unicast sender will ignore the feedback channel while a multicast sender will retransmit the packet.

2.1.3. *Overcoming Feedback Implosion Problem.* We have evaluated two methods of overcoming the feedback implosion problem. The first involves examining the Clear Channel Assessment during the NRT period shown in Fig. 21. If CCA is BUSY, it indicates the presence of negative feedback. Thus it is not the contents of the NAK packet that carries the feedback information but rather the increased power spectral density that causes the feedback channel to appear busy. Our second method involves using NAK tones instead of NAK packets. Here again, the tones do not carry any information but rather they cause the feedback channel to become busy, thus indicating the need for retransmission. Simulation results suggest that both methods have the same effect.

2.1.4. *Improving Reliability.* Apart from effectively avoiding $\mathcal{M}HTP$ with busy tones, our protocol also improves reliability by applying the methods discussed in Sec-

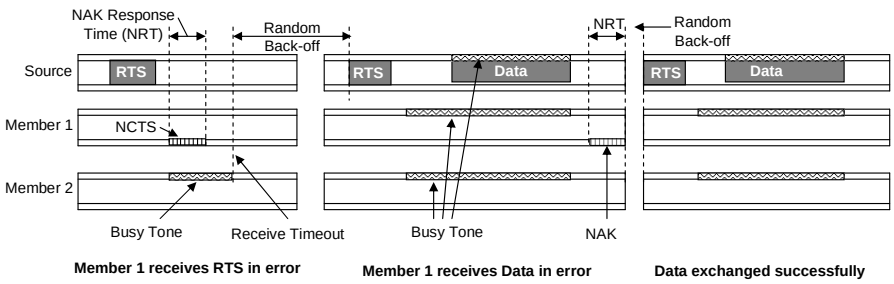

Figure 21. Illustration of the Tone Based Protocol

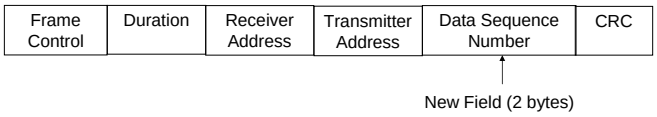

Figure 22. TBP RTS frame format.

tion 4. We add the data sequence number to the RTS packet (Fig. 22) so that members do not take part in the exchange if they already have that data packet.

2.2. Spurious NAK Problem. As illustrated in Fig. 6, stations in the interference range of the source may still be able to receive packets, but incorrectly. In a NAK based protocol, such stations will send a NAK even though the packet was not addressed to them. To handle this, each station in TBP uses a boolean function reflecting whether the average Received Signal Strength Indication (RSSI) value obtained during a packet reception is above a predetermined threshold; an erroneous packet will be NAKed only if the RSSI value is above the threshold.

Another problem is that feedback in TBP does not carry source or destination information. When a member is mobile, a NAK meant for one MMG may affect a different

MMG. This can happen only if: (1) the NRT periods of the two MMGs are perfectly aligned (their sources must sense the channel idle at the same time AND choose the same back-off time), and (2) the mobile station travels at least two hops during a transmission (groups one hop away will not have active transmissions since they will interfere with the multicast exchange.) The probability of both conditions being met is very low. When it does happen though, a retransmission occurs.

Therefore, as our simulation results will show, TBP significantly increases reliability but does not guarantee 100% packet delivery.

2.3. Operation in the Presence of Mobility. In TBP, channel is reserved using both RTS as well as Busy Tones. A new station may enter the coverage area of the members any time during the multicast exchange and is alerted to an ongoing transmission by the Busy Tones. The only weak point is on the side of the source, between the time the RTS was transmitted and until the Busy Tone is started at the end of the NRT period. This exposure occurs only if all the multicast destination stations are to one side of the source. This probability is very low. However, collisions do occur resulting in retransmissions.

2.4. Operation in the Presence of "Pure" IEEE 802.11 Terminals. TBP was designed to work with the ubiquitous IEEE 802.11 protocol. The channel access mechanism is the same and will work with each other. The channel reservation scheme, however, is different. TBP checks for three conditions before it makes a transmission: (1) no prior RTS/CTS exchange in the vicinity indicate the channel is in use; (2) the physical channel must be free (i.e. all subchannels must be idle); (3) the Busy Tone subchannel must be free. Conditions 1 and 2 are common with the IEEE 802.11 standard and are sufficient to prevent a TBP station from interrupting communication between "pure" stations. Condition 3 is

really a subset of condition 2. Hence a pure station will abstain from transmission when it senses the channel busy due to the Busy Tones in the busy tone subchannel.

3. Multi-channel Multicast Feedback Protocol (MMFP)

A major drawback of negative acknowledgement based protocols such as LBP, LBP-I and TBP is that the multicast source is not certain if and how many of its members received a data packet correctly. In Internet multicast applications using the Internet Group Management Protocol (IGMP), this lack of fine-grained knowledge is acceptable. However, other applications such as file transfers in smart classrooms and electronic stock exchanges may require that knowledge. It is expected that applications that require such knowledge will be run on a small scale (e.g. smart classrooms may have 30 to 100 students), or will be scaled up to thousands of members by distributing multicasting load between several smaller groups (e.g. stock exchanges using several access points (AP) with each AP servicing only a handful of brokers.) As such, there exists an opportunity for us to trade-off some scalability to acquire fine-grained feedback information. In this section we propose the Multi-channel Multicast Feedback Protocol that takes advantage of multi-channel technologies to obtain feedback from individual members and conduct retransmissions if necessary. Technologies may include TDMA, OFDM and MIMO. Therefore, MMFP represents a class of protocols that obtain information from individual members; for instance, Batch Mode Multicast MAC (BMMM) [39] and Reliable MAC (RMAC) [55] are special cases of MMFP in which subchannels are obtained through Time Division Multiplexing.

3.1. Throughput-Reliability Tradeoff. A multicast policy that requires all members to be ready before transmitting a data packet will render the multicast proto-

col unstable [12] i.e. the throughput of the protocol decreases as the number of members (N) increase inpite of the wireless broadcast advantage. In general, the throughput stability of a multicast protocol is related to a threshold k that defines the minimum number of members that must receive the data correctly in order for the transmission to be considered successful. When k is large, the probability that at least k members receive the data correctly decreases, resulting in more frequent retransmissions and hence a reduced throughput. Reduced throughput in turn increases packet delays and packet drops due to queue overflow. A small k on the other hand implies a lower level of reliability as the protocol will not schedule a retransmission even if up to $N - k$ members do not receive the data correctly. This tradeoff between reliability and throughput is studied in more detail in Chapter 6.

Since a multicast MAC cannot provide both high throughput and high reliability, an application must be willing to tradeoff some level of reliability for improved throughput. Since different applications have various requirements, MMFP is designed to let the application control the level of throughput it requires provided it is willing to accept a level of reliability commensurate with that throughput.

3.2. Physical Layer (PHY) Abstraction. MMFP is meant to be a generic protocol independent of the underlying PHY technologies that efficiently divide a channel into subchannels. The number of sub-channels available depends on the technology, regulatory constraints and standards. Let the available channel C be divided into s non-overlapping and non-interfering sub-channels $c_1, c_2, ..., c_s$. All subchannels are assumed to be always available. It is also assumed that subchannels allocated to different stations may be used at the same time and that stations have the ability to receive all of them concurrently.

3.3. Protocol Description.

3.3.1. *Multicast Source.* When a source is ready to transmit (state S1 of Fig. 23), it senses the channel C for a random time. If the channel is free it sends an RTS ($S2 \rightarrow S3$) addressed to the multicast group. It then listens for feedback on each of the subchannels that have been assigned to its members (state S4). If the number of CTSes received exceeds an application controlled threshold k, the source will transmit the data packet ($S4 \rightarrow S5$) on channel C; else the source will back off and try again ($S4 \rightarrow S2$). After transmitting the data, the source listens for an ACK on each of the subchannels assigned to its members. If an ACK was received from at least k members, the data exchange was successful and the source goes back to the waiting state ($S5 \rightarrow S1$); else the source backs off and tries to send the data again (state $S2$).

3.3.2. *Multicast Receiver.* When a station receives a multicast RTS addressed to its group, it first determines if it already has the data by examining the data sequence number in the RTS (states $R1 \rightarrow R2 \rightarrow R3$ in Fig. 24). In either case it sends a CTS on the subchannel that has been assigned to it ($R3 \rightarrow R4$ and $R3 \rightarrow R5$). The station then starts a timer and waits to receive the data packet. If the data packet is received without error, the timer is canceled and the station will send an ACK on its subchannel ($R5 \rightarrow R1$). If the data packet is not received at all or if the packet is dropped due to errors, the timer will expire and the station will return to state $R1$ without sending an ACK. For duplicate data packets (as determined by the sequence number in the RTS), the station sends an ACK even if the packet was received in error.

Fig. 25 illustrates the working of the protocol with two members and $k = 2$. During the first round, only one CTS is received and hence the source backs off without sending the

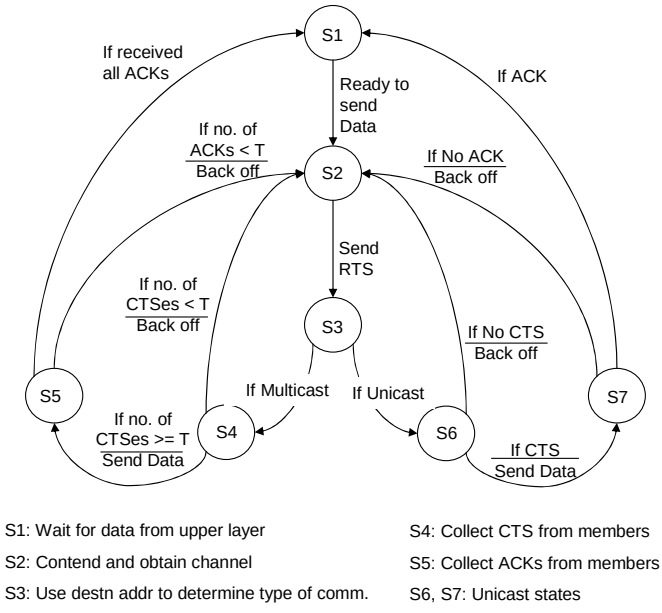

Figure 23. State diagram of multicast source.

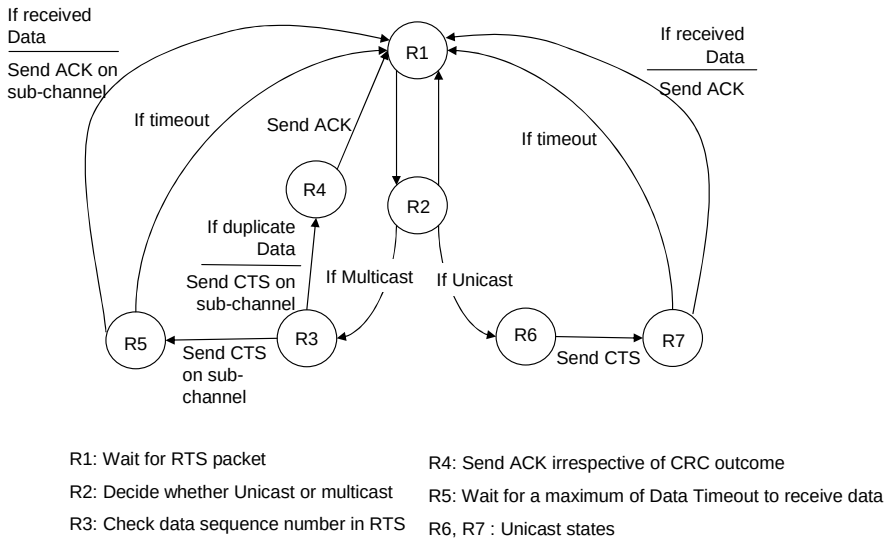

Figure 24. State diagram of the multicast receiver.

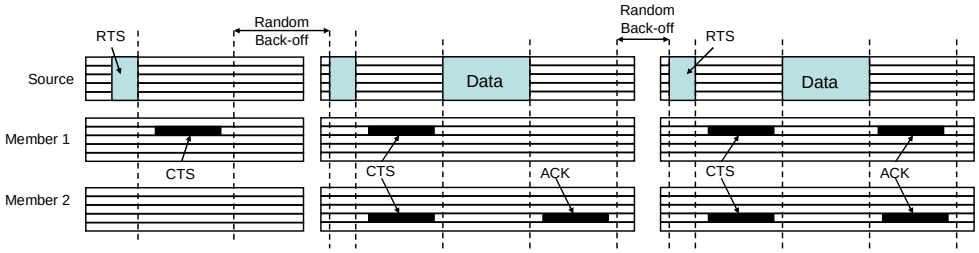

Figure 25. Illustration of the Multi-channel Multicast Feedback Protocol

data. In the second round, the source sends the data after receiving both CTSes. However, only one ACK is received and the the data must be retransmitted after a back off period. The data is successfully transmitted in the third round of the example.

3.4. Feedback Subchannel Assignment.

3.4.1. *Group Size smaller than Available Subchannels.* This is the simple case in which the source simply assigns feedback subchannels numbered according to the order in which stations join the group membership. Therefore the first member will be assigned subchannel 1, the second is assigned subchannel 2 and so on. When a member leaves, its subchannel is returned to the pool of unused subchannels and may be reassigned to a new member.

3.4.2. *Group Size greater than Available Subchannels.* Since a primary objective of the MMFP protocol is to obtain feedback information from individual members, it is important that our feedback strategy accommodate all members. In order to do this, we propose dividing the set of members into several batches. Each batch must be no larger than the number of subchannels available. When the group membership increases beyond the size that can be accommodated by the current number of batches, a new batch is created and the joining members are assigned to it.

The batches are numbered from 1 through the number of batches. On receiving a multicast data packet, feedback is provided one batch at a time sequentially. On retransmission attempts, the source may choose to merge batches if there are no conflicting subchannel assignments among the remaining members. Merging strategies are not studied in this thesis.

CHAPTER 6

PERFORMANCE ANALYSIS

Reliability comes at the cost of throughput and delay. In this section we mathematically analyze the throughput and delay of CSMA/CA based reliable multicast protocols. Values of variables used in the analysis and simulations are shown in Table 2.

1. Assumptions

This analysis is limited to protocols that completely avoid $\mathcal{M}HTP$. We assume that the network is operating at saturation throughput with at least one source having data to send at any given moment.

Let S_t be the power with which the source transmits; let G_t and G_r be the antenna gains at the transmitter and receiver respectively; let λ be the wavelength of the signal and L be the system loss (*i.e.* losses in the transceiver circuitry, cables etc.). To calculate the received signal power S_r at a station distance d from the transmitter, we assume that in the near field of the transmitting antenna the signal power attenuates according to Friis' propagation model given by $S_r = \frac{S_t G_t G_r \lambda^2}{(4\pi d)^2 L}$ and in the far field we assume the Two-Ray Ground propagation model given by $S_r = \frac{S_t G_t G_r (h_t h_r)^2}{d^4 L}$, where h_t and h_r are the heights (in

Table 2. Table of Symbols/Notations

Symbol	**Definition**	**Value**
C	Total Channel capacity(bits/sec)	2 Mbps
γ	Feedback subchannel. as fraction of C	2%
ρ	Tone subchannel as fraction of C	2%
C_D	Data Channel Capacity	$C(1-\gamma-\rho)$
η	Throughput Efficiency	Eq. 6.3
S	Effective Throughput	ηC
L_{RTS}	RTS packet length (bits)	44 bytes
L_{DATA}	Data packet length (bits)	512 bytes
n	No. of stations in system	
N	No. of members in multicast group	
W_{min}	Min. contention window size	32
m	Max. no. of backoff stages	5
τ	Retransmission Limit	7
X_j	No. of members participating in attempt j	
ξ_{total}	Total channel time	Eq. 6.2
$t_{s(i)}$	Time due to successful transmission in i^{th} attempt	
$t_{BO(i)}$	Time wasted due to back-off at start of i^{th} attempt	
$t_{de(i)}$	Time wasted due to data error in i^{th} attempt	
$t_{re(i)}$	Time wasted due to RTS error in i^{th} attempt	
$P_{RS(i)}$, $P_{DS(i)}$	Prob. of being in states R_i and D_{ij} respectively	
$P_{SUCC(i)}$	Prob. of success only due to i^{th} attempt	
P_{S0}, P_{SS}, P_{SF}	Prob. of being in states S_0, S_S and S_F	
Δ_i	Total delay experienced at end of i^{th} attempt	
$\mathcal{P}$	Probability Transition Matrix	
$\mathcal{V}$	State Probability Vector.	
P_b	Channel Bit Error Rate	
$P_{tr}[x\|n]$	Prob. x of n stations transmit.	
P_T	Prob. a station transmits	
P_C	Prob. of RTS collision	
P_R	Prob. a station receives RTS in error	
P_D	Prob. a station receives data in error	
P_{RE}	Prob. of RTS exchange failure	
P_{DE}	Prob. of data exchange error	
P_{FE}	Prob. of feedback error.	
P_{FAIL}	Prob. a transmission fails	
P_{SUCC}	Prob. a transmission succeeds	

meters) of the transmitter and receiver antennas respectively. The cross-over point from Friis' to Two-Ray propagation model is at $\frac{4\pi h_r h_t}{\lambda}$.

The probability of bit error P_b depends on the energy E_b with which the bit was received and the noise E_n that it must overcome. $E_b = \frac{S_r}{R}$ where R is the bit rate. For Binary Phase Shift Keying (BPSK) modulation $P_b = Q(\sqrt{\frac{2E_b}{E_n}})$, where Q is the Marcum q-function. Under the assumption that both interference from neighboring transmissions and channel noise is Gaussian, $E_n = N_0 + \sum_{i=0}^{N} I_i$ where $\frac{N_0}{2}$ is the variance of the channel noise with zero mean, and I_i is the interfering power from station i. The probability of Data and RTS packet errors are $P_D = 1 - (1 - P_b)^{L_{DATA}}$ and $P_R = 1 - (1 - P_b)^{L_{RTS}}$ respectively. For simplicity, we will assume that all members are equidistant from the source.

2. Throughput Efficiency

The throughput efficiency η of a multicast protocol gives us a measure of how well the protocol uses the channel capacity C. The throughput S delivered by the protocol is given by,

$$S = \eta C. \tag{6.1}$$

For multicast transmissions, the time wasted on collisions, the time wasted due to control and data packets errors, and the time spent on back-offs can be modeled as discrete random variables. When the channel is observed for a time window sufficient to cover several multicast transmissions, the channel will be occupied (or idle) for various amounts of time for various reasons including successful data transmission, collisions, packet errors and back-offs. According to the law of large numbers, the long-term average of a random variable when sampling repeatedly will tend toward the expected value of that random variable. The expected values quantify the average total amount of time taken for a multicast data

Table 3. Channel Holding Time

T_{SLOT}	Slot time (20 μs)
T_{SIFS}	Short Inter-frame Spacing (10 μs)
T_{DIFS}	DCF Inter-frame Spacing (50 μs)
T_{RTS}	RTS packet transmission time ($= L_{RTS}/C_D$)
T_{DATA}	Data transmission time ($= L_{DATA}/C_D$)
T_{FT}	Feedback Time (protocol dependent)
T_{COLL}	$T_{DIFS} + T_{RTS} + T_{SIFS} + T_{FT}$
T_{RERR}	$T_{DIFS} + T_{RTS} + T_{SIFS} + T_{FT}$
T_{DERR}	$T_{DIFS} + T_{RTS} + T_{SIFS} + T_{FT} + T_{SIFS} + T_{DATA} + T_{SIFS} + T_{FT}$
T_{SUCC}	$T_{DIFS} + T_{RTS} + T_{SIFS} + T_{FT} + T_{SIFS} + T_{DATA} + T_{SIFS} + T_{FT}$
See Fig. 26 for illustrations.	

transfer, and the fractions of that time spent on retransmissions, successful transfer and the time the channel was left idle because all transmitters were backing off. Theoretically, the expected value of a random variable ξ can be calculated as $E[\xi] = \sum_{i=0}^{\infty}(p_i\xi_i)$, where p_i is the probability distribution function of the specific values ξ_i that ξ can take.

In the case of multicast transmission, ξ can take the values T_{SUCC}, T_{IDLE}, T_{COLL}, T_{RERR} and T_{DERR} representing channel holding times for successful transmissions, idle channel, packet collisions, transmission failures due RTS errors and data errors respectively. The expected total channel time required for a multicast exchange is then given by $E[\xi_{total}] = \sum_i(P_iT_i)$ where $i = \{SUCC,\ IDLE,\ COLL,\ RERR,\ DERR\}$. That is,

$$E[\xi_{total}] = P_{SUCC}T_{SUCC} + P_{IDLE}T_{IDLE} + P_{COLL}T_{COLL} + P_{RERR}T_{RERR} + P_{DERR}T_{DERR} \tag{6.2}$$

η is the ratio of the time the channel was used for successful data packet transmissions to the total time for which the channel was held during that window. That is,

$$\eta = \frac{P_{SUCC}T_{DATA}}{E[\xi_{total}]} \tag{6.3}$$

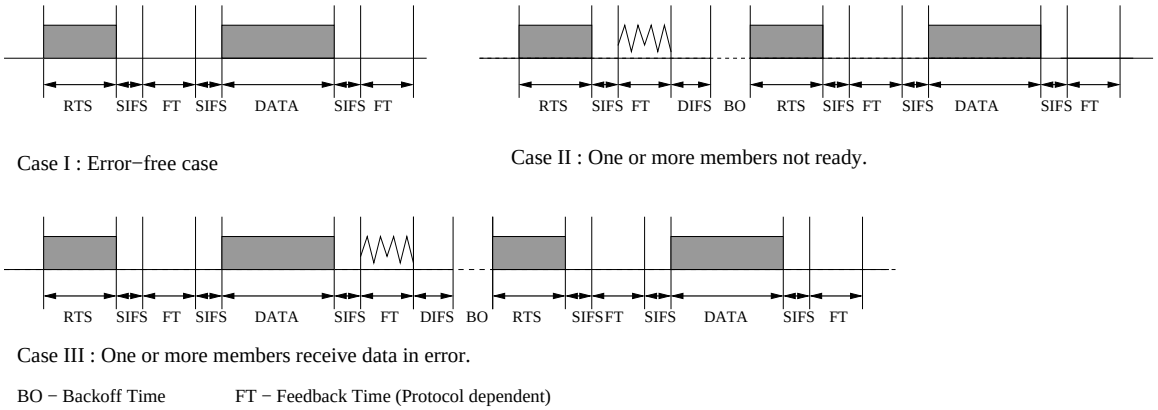

Figure 26. Slot diagrams of a generic ARQ-based reliable multicast protocol. Feedback Time (FT) depends on whether protocol employs Individual or Group feedback.

Since the channel holding times T_i are constants (see Table 3 and Fig. 26), to calculate η it is sufficient to find the probability distribution function P_i, $i = \{$*SUCC, IDLE, COLL, RERR, DERR*$\}$. Fig. 27 is a graphic representation of the steps we follow in deriving the throughput of multicast protocols and will aid in understanding the following analysis.

2.1. Calculation of P_{SUCC}. A multicast exchange can be successful only if the transmission actually occurs and it is not aborted due to an RTS collision, error in channel reservation control packets, or data exchange error. Let P_1 be the probability that at least one source transmits, and let P_C, P_{RE} and P_{DE} be the probability that the multicast exchange is aborted due to collision, RTS/CTS/NCTS packet error and data exchange error respectively. Thus,

$$P_{SUCC} = P_1(1 - P_C)(1 - P_{RE})(1 - P_{DE}) \tag{6.4}$$

Let P_T be the probability that a given station will transmit. Then the probability that at least one of n stations will transmit is, $P_1 = 1-(1-P_T)^n$. The probability P_C that a collision affects that transmission is the probability that at least one of the remaining $n-1$ stations start a concurrent transmission. This probability is, $P_C = 1-(1-P_T)^{n-1}$.

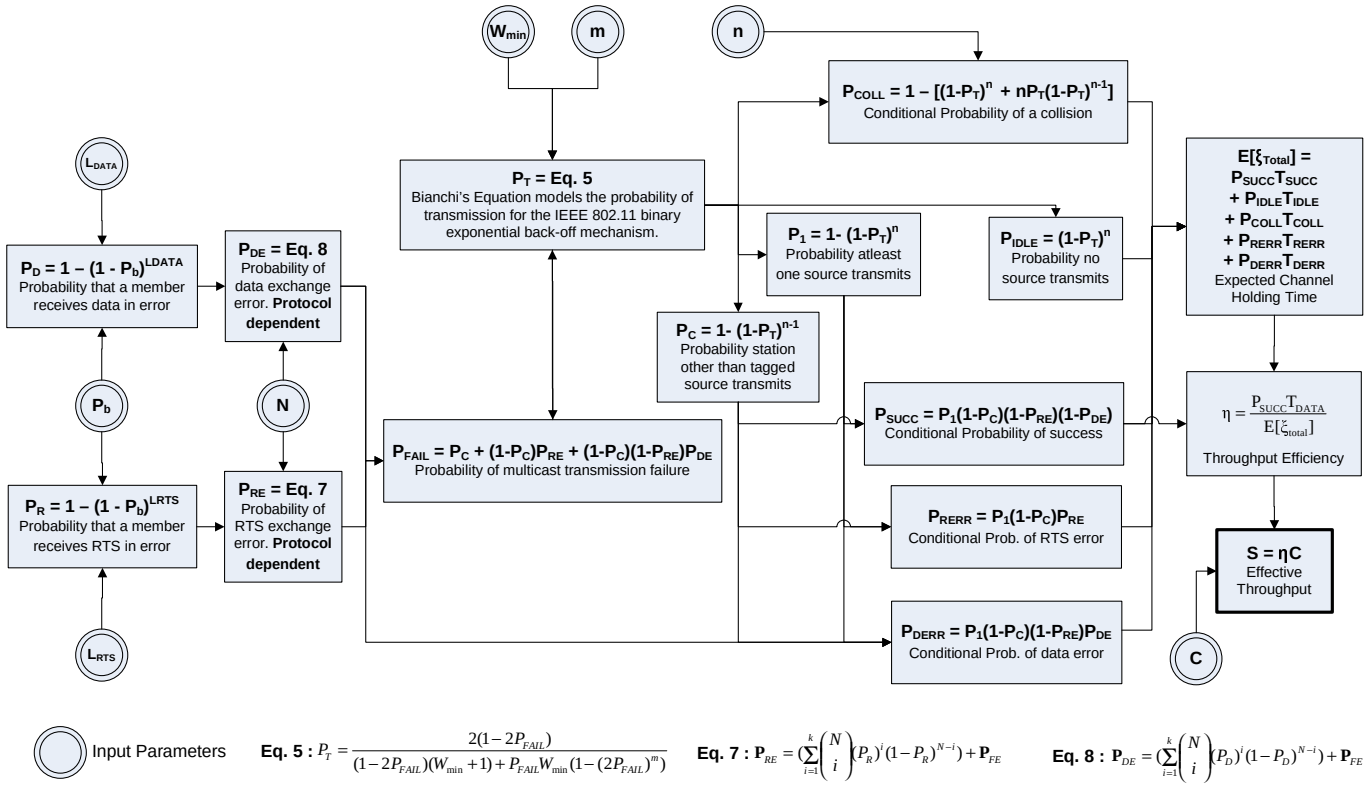

Figure 27. Throughput analysis flow-diagram.

P_T depends on the back-off scheme that is used. The exponential random back-off scheme used by IEEE 802.11 was fairly accurately modeled by Bianchi [4] as a bi-dimensional process with discrete-time Markov chains with the simplifying approximation that packets transmitted by each source collide with a constant probability independent of the number of retransmissions the packets have already suffered. As per the model proposed in [4], the probability that a station will transmit is,

$$P_T = \frac{2(1-2P_{FAIL})}{(1-2P_{FAIL})(W_{min}+1) + P_{FAIL}W_{min}(1-(2P_{FAIL})^m)} \qquad 0 \leq P_{FAIL} \leq 1 \quad (6.5)$$

where, W_{min} is the minimum back-off window size, m is the number of back-off stages and P_{FAIL} is the probability of failure. Eq. 6.5 indicates that the probability that a station will transmit (or retransmit) depends on the probability that a transmission will fail and on the contention window size. The smaller the contention window size, greater is the probability that a station will transmit in a given slot. Analysis in [4] is limited to failures caused by collisions. We extend this work to multicast traffic by including failures due to RTS and data exchange errors by redefining the probability of failure as,

$$P_{FAIL} = P_1(P_C + (1-P_C)P_{RE} + (1-P_C)(1-P_{RE})P_{DE}) \quad (6.6)$$

2.2. Calculation of P_{IDLE}. P_{IDLE} is the probability that a time slot is wasted because all stations that have a packet to transmit are in the back-off state and the channel is left idle. CSMA/CA protocols use binary exponential random back-offs to avoid lockstep collisions between two or more transmitting stations. Since P_T is the probability that a station will transmit, the probability that the channel is idle is $P_{IDLE} = (1-P_T)^n$.

2.3. Calculation of P_{COLL}. The probability that a collision will occur is the probability that two or more of the n stations transmit. Let x be the number of sta-

tions that transmit. Then $P_{COLL} = P[x >= 2] = 1 - P[x < 2]$. Therefore, $P_{COLL} = 1 - [(1 - P_T)^n + nP_T(1 - P_T)^{n-1}]$.

2.4. Calculation of P_{RERR} and P_{DERR}. In wireless networks, the channel conditions are not guaranteed to be the same at all stations and as such the errors that occur during packet reception at members will be different from each other. Let P_R be the probability that a member receives the RTS packet in error and let P_D be the probability that it receives a data packet in error. Let P_{FE} be the probability of feedback error; P_{FE} depends on the actual feedback mechanism and is thus protocol dependent. It must include control packet errors (e.g. ACK, NAK, CTS, NCTS), feedback collisions, false NAKs and any other feedback errors that cause a retransmission.

The probability P_{RE} that an RTS exchange fails due to errors depends on the semantics of the multicast protocol. In some protocols the exchange fails if even one member receives the RTS in error while in others the exchange will continue if one or more stations received and responded to the RTS correctly. If k is the minimum number of members that must receive the RTS correctly for the exchange to continue, then the probability of RTS exchange error is given by:

$$P_{RE} = (\sum_{i=0}^{k-1} \binom{N}{i} (1 - P_R)^i (P_R)^{N-i}) + P_{FE}. \tag{6.7}$$

An RTS exchange error can occur only if our source did transmit an RTS and that RTS did not suffer a collision. Therefore, $P_{RERR} = P_1(1 - P_C)P_{RE}$.

Similarly for a data packet error, $P_{DERR} = P_1(1 - P_C)(1 - P_{RE})P_{DE}$. where,

$$P_{DE} = (\sum_{i=0}^{k-1} \binom{N}{i} (1 - P_D)^i (P_D)^{N-i}) + P_{FE}. \tag{6.8}$$

Note that a data packet error is possible only if an RTS was transmitted without collision or exchange error.

We now have the probability distribution function P_i, $i = \{SUCC, IDLE, COLL, RERR, DERR\}$, that can be substituted in Eq. 6.3 to obtain the throughput efficiency η.

2.5. Calculation of Numerical Upper Bound on Throughput. In order to find the numerical upper bound on the throughput of the multicast protocol in a wireless LAN, we must study multicast operation in isolation. To do this, we eliminate all unicast traffic and have only one multicast group with a single source (n=1). This will ensure that back-off delays are only due to error recovery and not due to channel contention. Since no RTS collisions can occur, Eq. 6.6 now becomes: $P_{FAIL} = P_T P_{RE} + P_T(1 - P_{RE})P_{DE}$. All equations are now linear in P_T and P_{FAIL}, and η can be found by direct substitution when P_{RE}, P_{DE} and the number of group members N are known.

Fig. 28 shows the throughput of a protocol with semantics that require all members to receive data correctly (i.e. $k = N$ in Eqns. 6.7 and 6.8). The throughput, calculated using Eq. 6.1 for a 2Mbps channel, decreases with increasing group size for all values of SNR. The throughput of such protocols are severely affected at low SNR (e.g. 7dB).

2.6. Trading Off Reliability for Throughput. Multicast throughput can be significantly improved if the protocol semantics require only a subset of members to receive the data correctly. The number of members that must receive the data correctly for a transmission to be considered successful is modulated by parameter k of Eqns. 6.7 and 6.8. However, it is useful to express k as a fraction of the group size N since applications may want to determine what fraction of their members are expected to receive data correctly for a desired throughput. Fig. 29(a) and Fig. 29(b) show the tradeoff between reliability and throughput at 7 dB for protocols with $O(1)$ and $O(N)$ feedback times respectively.

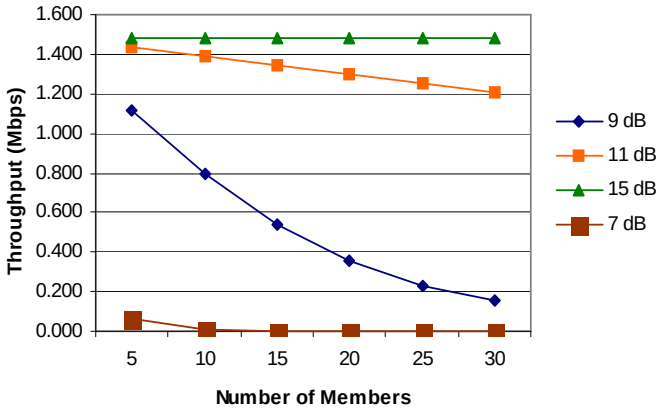

Figure 28. Relationship between throughput and number of group members.

Applications that require higher throughput but can accommodate greater packet loss can use low values of k relative to their group size.

Fig. 30(a) and Fig. 30(b) show throughput as a function of constant values of k for protocols with $O(1)$ and $O(N)$ feedback times respectively. These results determine the throughput of multicast applications that use a quorum to determine success i.e. a minimum number of members must receive the data correctly irrespective of group size. In general, throughput is better for smaller values of k. However, for small group sizes, throughput is dominated by the ratio k/N. This is because the probability that at least k members receive data correctly reduces with the group size i.e. the larger the group, higher is the probability that at least k members receive the data correctly. Finally, as seen in Fig. 30(b), irrespective of k, throughput is limited by the group size for protocols with $O(N)$ feedback time.

2.7. Throughput of TBP. TBP divides the available channel into three subchannels. Let γ and ρ be the fractions of the available channel C allocated to the feedback and busy tone subchannels respectively. The remainder of the available channel is allocated to

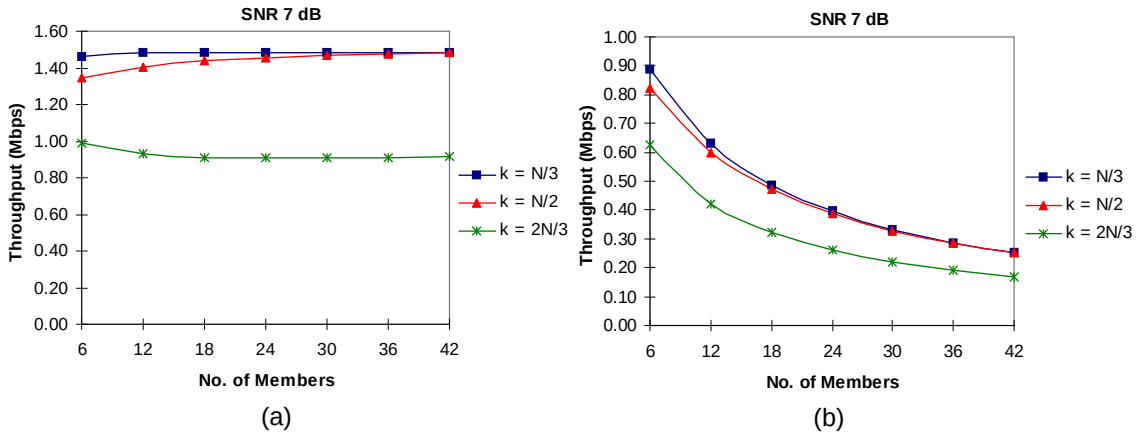

Figure 29. Cost of reliability (parameter k) in terms of throughput for protocols requiring: (a) $O(1)$ feedback time, and (b) $O(N)$ feedback time.

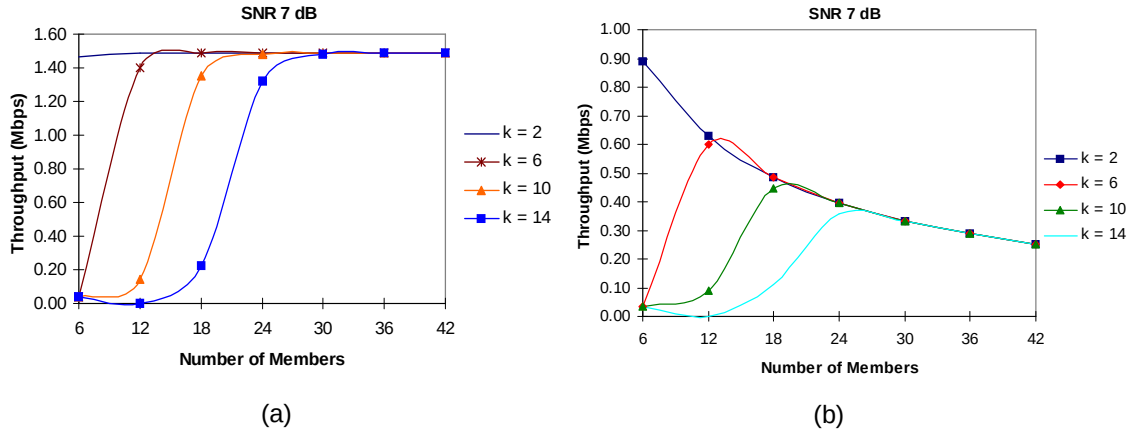

Figure 30. For constant k, throughput of protocols: (a) with $O(1)$ feedback time increases with decrease in ratio of k to number of members, (b) with $O(N)$ feedback time is primarily determined by group size.

the data subchannel. Its capacity is then given by $cc = C(1-\gamma-\rho)$. Analysis done in [59] show that the optimal size of a tone subchannel should be in the range $[\frac{1}{100}, \frac{1}{50}]^{th}$ of the size of the total channel. Therefore the use of the tone and feedback subchannels reduces overall capacity by only about 2 to 4%.

Note that unlike in IEEE 802.11 unicast and LBP, the channel becomes free immediately after a successful data transmission in TBP as well as IEEE 802.11 multicast.

In TBP, there are two types of feedback errors: (1) the channel noise is above the EDT resulting in false NAK detection, (2) the received power of a NAK tone falls below EDT resulting in the source not detecting the NAK. The first type of error affects throughput while the other affects reliability.

Let P_s be the probability the noise energy E_n is greater than the Energy Detect Threshold. That is:

$$P_s = P[E_n > EDT] = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{EDT}^{\infty} e^{-(\frac{(u-\mu)^2}{2\sigma^2})} du \tag{6.9}$$

For a Gaussian noise with zero mean and a variance of $\frac{N_0}{2}$, we have

$$P_s = \frac{1}{\sqrt{\pi N_0}} \int_{EDT}^{\infty} e^{-(\frac{u^2}{N_0})} du = \frac{\sqrt{2N_0}}{\sqrt{\pi N_0}} \int_{\frac{EDT}{\sqrt{2N_0}}}^{\infty} e^{-(\frac{x^2}{2})} dx \tag{6.10}$$

$$= 2Q(\sqrt{\frac{EDT}{\sqrt{2N_0}}}) \tag{6.11}$$

Let the sampling rate be B MHz and let the sampling time be T_s. The number of samples obtained will be $Y = BT_s$. The PHY considers the channel busy if y of the Y samples are above EDT. Therefore the probability P_{FE} of feedback error is:

$$P_{FE} = \binom{Y}{y} P_s^y (1-P_s)^{Y-y} \tag{6.12}$$

P_{FE} can be used to calculate P_{RERR} and P_{DERR}. P_{IDLE} and P_{COLL} are as discussed earlier in this section. We thus know the probability distribution function for the total time

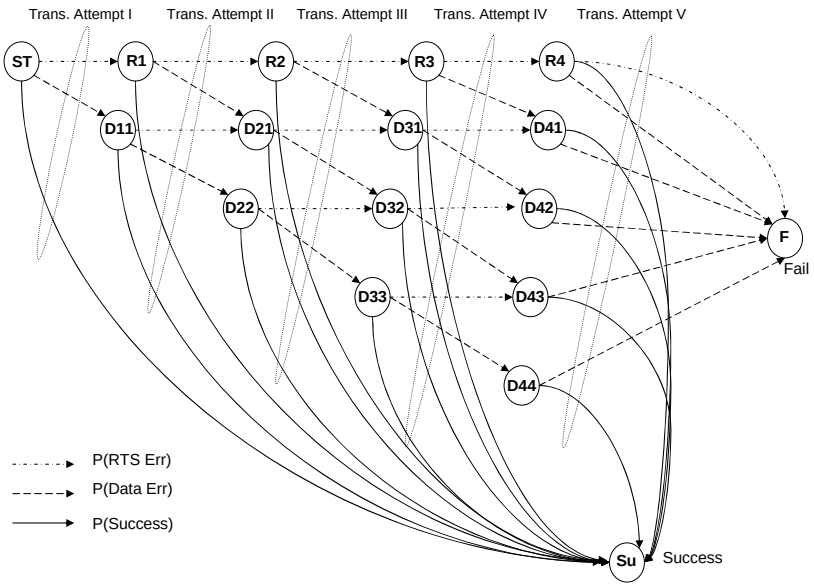

Figure 31. Markov chain model of retransmission probabilities.

required to make a successful transmission; from this we can calculate η according to Eq. 6.3.

3. Expected Delay

In this section, we study the delay caused in reliable multicast protocols due to retransmissions resulting from errors and collisions. Fig. 32 shows the steps involved in our analysis. Each data transfer may have zero or more retransmission attempts. Total delay is the sum of the delays of all the retransmission attempts.

Each retransmission attempt has three possible outcomes: (1) the RTS exchange fails, (2) RTS exchange is successful but the data exchange fails, and (3) both RTS and data exchange are successful. Failures may be due to packet errors, packet collisions, or

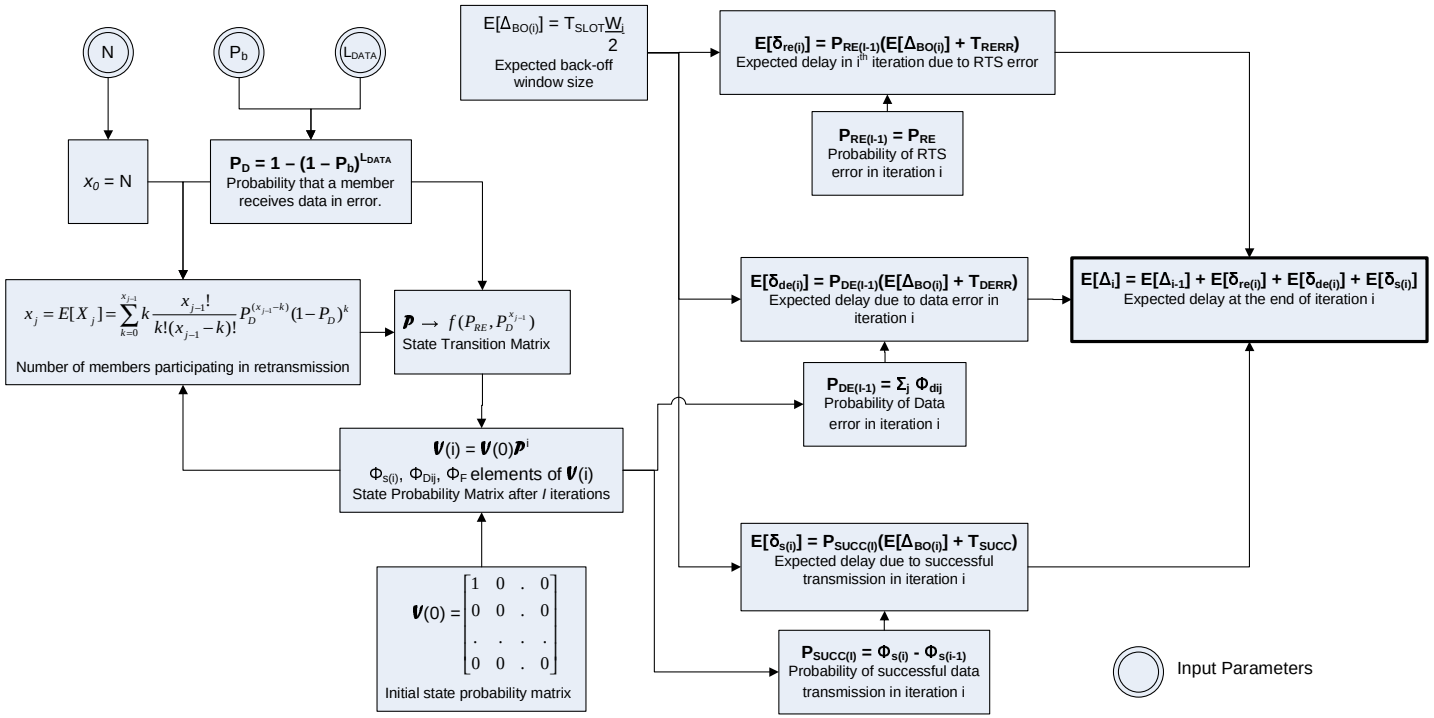

Figure 32. Delay analysis flow-diagram.

error in the feedback. Each retransmission attempt results in a delay that is the sum of a random back-off value plus the channel holding time associated with the outcome of the retransmission. The expected delay at the end of the i^{th} retransmission attempt is the sum of the delays of the previous attempts plus the expected contributions of each outcome of the i^{th} retransmission attempt. Thus,

$$E[\Delta_i] = E[\Delta_{i-1}] + E[\delta_{re(i)}] + E[\delta_{de(i)}] + E[\delta_{s(i)}] \tag{6.13}$$

where $E[\delta_{re(i)}]$, $E[\delta_{de(i)}]$, $E[\delta_{s(i)}]$ and $E[\Delta_{i-1}]$ are the expected contributions of RTS and data exchange failures[1], successful transmission, and the cumulative delay from the previous retransmission attempts. To calculate these expected values, we must know the probability of entering retransmission attempt i and the probability of occurrence of each of the three outcomes. Let the probability of entering retransmission attempt i **and** having an RTS exchange failure there be $P_{RE(i-1)}$, and that of having a data exchange failure be $P_{DE(i-1)}$. Let the data transmission be successfully concluded in attempt i with probability $P_{SUCC(i)}$. We then have, $E[\delta_{re(i)}] = P_{RE(i-1)}(E[\delta_{BO(i)}] + T_{RERR})$, $E[\delta_{de(i)}] = P_{DE(i-1)}(E[\delta_{BO(i)}] + T_{DERR})$, and $E[\delta_{s(i)}] = P_{SUCC(i)}(E[\delta_{BO(i)}] + T_{SUCC})$. $E[t_{BO(i)}]$ is the back-off time in retransmission attempt i. T_{RERR}, T_{DERR} and T_{SUCC} are the channel holding times described in Table 3. We can thus calculate the average delay if we know the probabilities $P_{RE(i)}$, $P_{DE(i)}$ and $P_{SUCC(i)}$ for all i.

An important observation is that *even though a multicast data transfer is unsuccessful, there may be a subset of members that received the data correctly.* These members need not take part in subsequent retransmissions of the same data and hence $P_{DE(i)}$ decreases

[1] $E[\delta_{re(i)}]$ and $E[\delta_{de(i)}]$ also include the contribution of feedback errors e.g. CTS, ACK and false NAK errors

with the number of retransmissions of the data packet. Members that have not yet received the data are called *active* members.

The probability P_D that a member receives the data in error is independent of other members, as a result of which the number of active members in the i^{th} retransmission step is a binomial random variable X_j with expected value $x_j = \mathrm{E}[X_j]$. x_j's value depends on the number of active members in the previous retransmission attempt which in turn depends on the number of times, j, the data packet was actually retransmitted. Intuitively, more the active members we have, greater will be the number of them that receive a packet in error. Hence, expected number of members that will take part in the next retransmission is,

$$x_j = E[X_j] = \sum_{k=0}^{x_{j-1}} k \frac{x_{j-1}!}{k!(x_{j-1}-k)!} P_D^{(x_{j-1}-k)} (1-P_D)^k \tag{6.14}$$

It must be noted that x_j depends on the number of times the data was transmitted (j) and not the retransmission count i; specifically, $j \leq i$ because the data packet is not transmitted when an RTS error occurs. We must therefore track both i and j. We model this by the Markov chain shown in Fig. 31. As mentioned earlier, each retransmission can have three outcomes. Since all stations listen for an RTS, P_{RE}, the probability of RTS error is a constant. The probabilities of data error and of transmission success depend on x_j.

With reference to Fig. 31, we enter state R_i if we haven't yet transmitted the data packet and an RTS error occurs in the $(i-1)^{th}$ retransmission attempt. States D_{ij} are entered when the data packet is transmitted at least once. The i in R_i and D_{ij} counts the total number of retransmission attempts while j counts the number of attempts in which the data packet was transmitted. State S_S is entered when all members have received the data packet correctly. State S_F represents failure and is entered when the retransmission limit τ is exceeded.

We calculate the state transition probability matrix $\mathcal{P}$ as follows. Since all stations receive an RTS in all retransmission attempts, the transition probability from any state $R_{(i-1)} \rightarrow R_i$ or $D_{(i-1)j} \rightarrow D_{ij}$ is always P_{RE}. Let x_{j-1} be the number of remaining active members in state $D_{(i-1)(j-1)}$ according to Eq. 6.14. The probability that the RTS exchange is successful but the data exchange fails in retransmission attempt i is represented by the transition probability from $D_{(i-1)(j-1)} \rightarrow D_{ij}$, where $i < \tau$. This probability is $(1 - P_{RE})P_D^{x_{j-1}}$. If $i = \tau$, then an RTS or data error will result in the data being dropped. Thus the transition probability from any state D_{ij} or R_i, $i = \tau$, to state F is $P_{RE} + (1 - P_{RE})(1 - P_D^{x_{j-1}})$. The transition probability from state $D_{(i-1)(j-1)}$ to state S_S is $(1 - P_{RE})P_D^{x_{j-1}}$. All other transition probabilities are 0.

Let Φ_{Ri} and Φ_{Dij} represent the probabilities of being in states R_i and D_{ij} respectively. Let Φ_0, Φ_S and Φ_F represent the probabilities of being in state S_0, S_S and S_F respectively. For a fresh data transmission, the initial state probability row vector $\mathcal{V}(0)$ will have $\Phi_0 = 1$ with all other entries being 0.

Given $\mathcal{P}$ and the initial state probability vector $\mathcal{V}(0)$, we can calculate the probability of being in any state after i transitions (retransmissions) using the Chapman-Kolmogorov equations [69] as follows:

$$\mathcal{V}(i) = \mathcal{V}(0)\mathcal{P}^i \qquad 0 \leq i \leq \tau \tag{6.15}$$

Each value of i in Eq. 6.15 corresponds to a retransmission attempt. Therefore the value of Φ_S in row vector $\mathcal{V}(i)$ is the probability that all members have received the data correctly by the end of attempt i i.e. data exchange is successful. The contribution of the i^{th} retransmission attempt to the increase in the probability that data exchange is successful is denoted by $P_{SUCC(i)}$ and is the difference in the values of Φ_S in $\mathcal{V}(i)$ and $\mathcal{V}(i-1)$. The

probability of RTS exchange error in attempt i is $P_{RE(i)}$ in $\mathcal{V}(i)$ while the probability of data exchange error is the sum $P_{DE(i)} = \sum_{j=0}^{i-1} \Phi_{Dij}$.

After each RTS or data error, the transmitting station backs off for a random number of time slots. The random number of slots for retransmission attempt i is chosen from the contention window given by $W_i = \begin{cases} 2^i W_{min}, & 0 \le i \le m \\ 2^m W_{min}, & m \le i \end{cases}$ [16], where m is the maximum number of back-off stages and W_{min} is the minimum window size. The average back-off delay in retransmission attempt i, $E[\delta_{BO(i)}]$, will be $T_{SLOT}\frac{W_i}{2}$, where T_{SLOT} is the slot time.

We can now express Eq. 6.13 in terms of $P_{RE(i)}$, $P_{DE(i)}$, $P_{SUCC(i)}$ and $E[\delta_{BO(i)}]$ as follows,

$$\begin{aligned} \Delta_i = {} & P_{RE(i-1)}P_{DE(i-1)}\Delta_{i-1} + P_{RE(i-1)}(E[\delta_{BO(i)}] + T_{RERR}) \\ & + P_{DE(i-1)}(E[\delta_{BO(i)}] + T_{DERR}) \\ & + P_{SUCC(i)}(E[\delta_{BO(i)}] + T_{SUCC}) \qquad 0 < i < \tau \end{aligned} \tag{6.16}$$

where T_{RERR}, T_{DERR} and T_{SUCC} are defined in Table 3.

3.1. Expected Delay in TBP. In TBP, an RTS exchange fails even if one member receives the packet in error. If we assume that the probability of detecting a false NAK is insignificant ($P_{FE} = 0$), then $P_{RE(i)} = P_{RE} = 1-(1-P_R)^N$ and $P_{DE(i)}$ is calculated based on $P_D = 1-(1-P_b)^{L_{DATA}}$. Fig. 33 compares the delay of TBP obtained through analysis with simulation results.

Fig. 33 shows effect of group size on the average delay that will be experienced by the source in completing a successful data transfer. The average delay increases with increasing group size as well as with the bit error rate. As is seen, the delays computed from our mathematical analysis closely matches our simulation results. This is true for retransmission probabilities as well. At a low bit error rate of 10^{-6} and 5 nodes, the

Table 4. Retransmission probabilities for a group size of 5 and BER of 10^{-6}.

Retransmissions	0	1	2	3	4	5	6
Simulation	0.9756	0.0243	0.0001	0.0000	0.0000	0.0000	0.0000
Analysis	0.9767	0.0231	0.0002	0.0000	0.0000	0.0000	0.0000

Table 5. Retransmission probabilities for a group size of 25 and BER of 10^{-5}.

Retransmissions	0	1	2	3	4	5	6
Simulation	0.2967	0.5526	0.0974	0.0393	0.0100	0.0033	0.0007
Analysis	0.3018	0.5741	0.0721	0.0399	0.0079	0.0030	0.0007

difference between retransmission probabilities from simulation and analysis differ only by ± 0.001 (Table 4) while for a BER of 10^{-5} and 25 nodes this difference goes up to around ± 0.02 (Table 5). In general this difference is within ± 0.005. In the analysis, we ignored the overhead due to the PHY and MAC headers for both control as well as data packets.

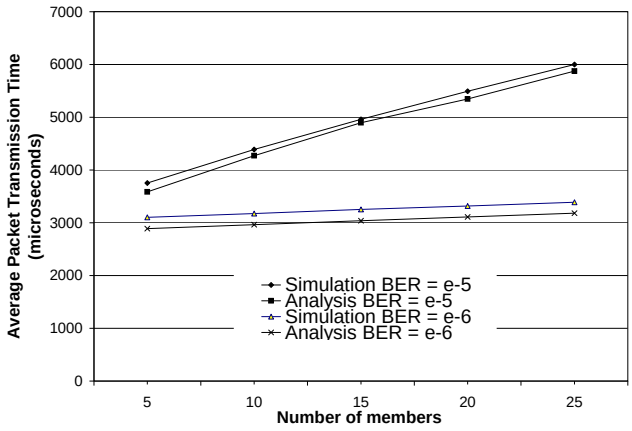

Figure 33. Delay performance of TBP.

4. Reliability

The analysis in this section is based on the assumption that members that receive a data packet correctly do not participate in successive retransmisions for that packet. The

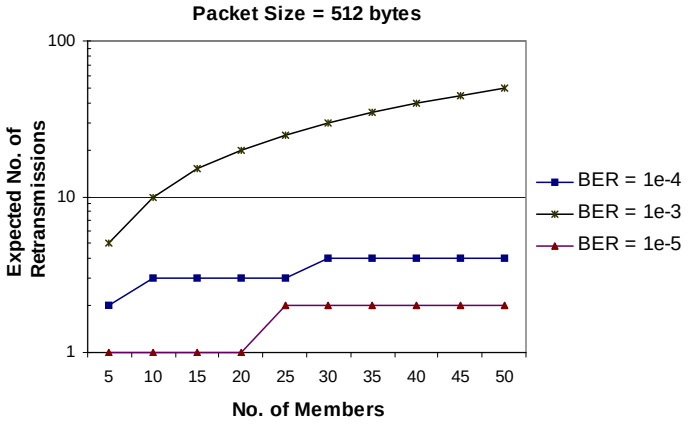

Figure 34. Number of retransmissions as a function of channel bit error rate (BER).

number of members that do not have the data at the end of retransmision attempt i is then given by Eq. 6.14.

Fig. 34 shows for different channel bit error rates, the number of retransmissions necessary to reliably transfer data to every member of a group. Though the number of retransmissions required is a function of the group size, the relationship is less than linear for bit error rates as high as 10^{-4}. The important observation is that the retransmission limit of 7 adopted by IEEE 802.11 unicast is also sufficient for multicast communication as long as the packet size not larger than 1024 bytes.

Fig. 35 shows that packet size plays a bigger role than the group size in determining the required number of retransmissions. In general, and in keeping with observations of unicast traffic, larger packets are more prone to errors in wireless networks.

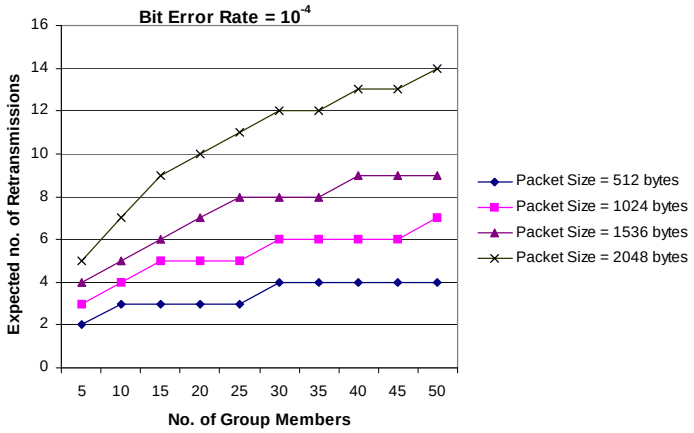

Figure 35. Number of retransmissions as a function of group size.

CHAPTER 7

SIMULATION RESULTS

We used *ns-2* [1] to simulate five protocols: TBP, BMW, LBP, LBP-I and MMFP. With reference to Section 1, LBP and LBP-I implement Strategies II and V respectively while TBP and MMFP implement Strategy III . LBP, LBP-I and MMFP use PI feedback while TBP uses NG. BMW addresses successive data packets to different members in a round-robin fashion. RTS/CTS/ACK are exchange only with the members being addressed while the other members snoop on the communication. Simulation results are an average of several runs. For the throughput study, each run was 500 seconds and for the reliability and delay studies, each run was the time taken for the multicast source to send 10,000 data packets. The position of nodes and/or the random seed was changed between runs. The number of runs was determined by the desired confidence intervals for a 95% confidence.

Simulations were conducted and results reported according to suggestions in [36] and [25]. Output from simulations were recorded only after the initialization period in order to avoid bias. Simulator environment and protocol configurations that were explicitly changed are listed in Appendix A; all other parameters are as reported in *ns-default.tcl* of *ns-2*. Network topologies and traffic model used in our simulations were chosen to demonstrate protocol performance under specific usage scenarios and are not meant to reflect performance for all possible topologies. In our simulations, a station is one-hop

from another if the packets it receives from that station have a signal-to-noise ratio (SNR) above a *Receive Threshold*. Therefore, whether a station is a one-hop neighbor is not only determined by distance but also by the level of channel noise and interference at the time a packet is received.

To be fair to all protocols, all stations had the same MAC. The simulation set-up is as follows:

- *Data Traffic Model*: Since data files are typically several orders of size larger than the 512-bytes data payload size of the MAC, data from the upper layers will appear as a continuous stream of packets. Hence, we used constant bit rate (CBR) traffic for all simulations.

- *Mobility Model*: We used the Random Waypoint Mobility Model [30] . When using this model for long simulation runs, stations tend to aggregate at the center of the mobility area [3] and skew results. To avoid this, we took the average of several shorter simulation runs for each speed.

- *Channel and Error Model*: The relationship between received signal power and the distance between transmitter and receiver was modeled using the Friis' propagation model for distances in the near region of the transmitter, and the two-ray ground propagation model for distances in the far region [51]. For a received Signal-to-Noise ratio, the probability of error was consistent with the BPSK modulation scheme (see Chap. 6 - Sec. 1). The noise included an Additive White Gaussian Noise (AWGN) channel noise component, and interference from other concurrent transmissions. The SNR indicated on graphs in this section are for points at the edge of the coverage area of a transmitter.

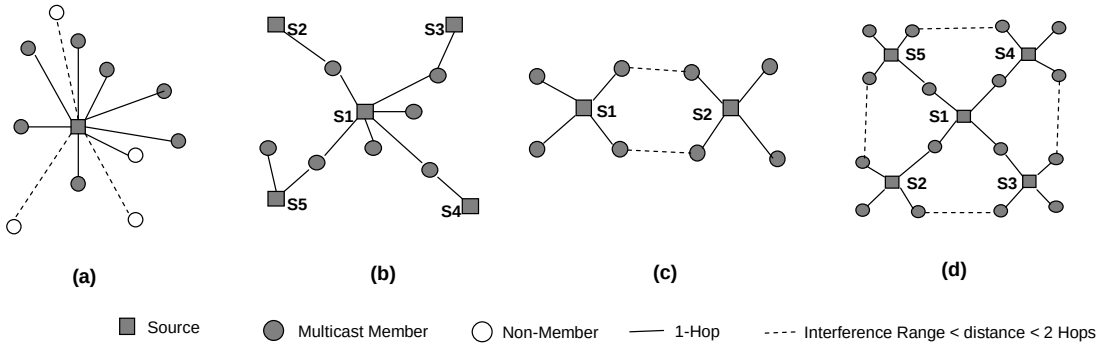

Figure 36. Scenarios: (a) Single group (b) Overlapping groups (c) Non-overlapping groups (d) Overlapping groups.

- *Packet Capture Model*: The capture mechanism in *ns-2* allows the capture of only the first arriving packet if its power is greater than that of all other concurrently arriving packets. We re-implemented the capture mechanism so that the highest powered packet is captured irrespective of the order in which packets begin to arrive. Also, the signal strength of the strongest powered packet is compared against the sum of interference of all the other concurrent packets to determine capture. Once captured, the packet is subject to the bit error probability that conforms to the observed signal-to-interference ratio for BPSK modulation.

  Analysis in [59] indicate that tones require 1% of total available bandwidth for reliable detection within 5 μs; we allocated 2% of the 2 Mbps total bandwidth to each of the feedback and tone subchannels. Since we only need to detect a tone and not decode it, we assumed the tone and feedback were always reliable.

1. Reliability

1.1. Reliability in the Presence of Channel Errors. We used the scenario in Fig. 36(a). For an SNR of 7dB, Fig. 37 shows that TBP and BMW deliver 100% packets to all members. However the performance of BMW is affected as the group size increases. The average PDR of LBP and LBP-I varies between 96% and 99%; the reduced reliability is because the ACKs from the Leader overpowers NAKs from distant members (capture effect, see Sec. 2). The confidence intervals are indicated around the mean.

Fig. 39 shows the standard deviation σ of the PDR obtained during several simulation runs. LBP shows a wide range of values for σ indicating that its fairness of reliability is erratic - sometimes it is very fair ($\sigma = 0$) while during other runs it was not ($\sigma = 11\%$). In cases where the Leader is closest to the source, no retransmissions were done irrespective of how many other members received the data in error. TBP and BMW on the other hand have consistently low σ values indicating that the protocols are fair to all members during all simulation runs.

Fig. 38 shows that the reliability of TBP and BMW drops under high error probabilities (SNR 5dB); packets were being dropped after reaching their retransmission limits and the increased delay caused due to error-recovery resulted in queue overflows. The queue length was set to 20 packets for the results shown in Fig. 38.

1.2. Reliability in Ad Hoc Environments. Fig. 36(b) shows a sample scenario used. Each of the five sources (of different MMGs) generated 10 packets per second for 200 seconds for a total of 10,000 packets. Results in Fig. 40 indicate that TBP is the most reliable with a PDR of 100% followed by LBP-I with a PDR of 99%; LBP performed the worst due to $\mathcal{MHTP}$. It also had σ of up to 27.91% indicating that members had widely

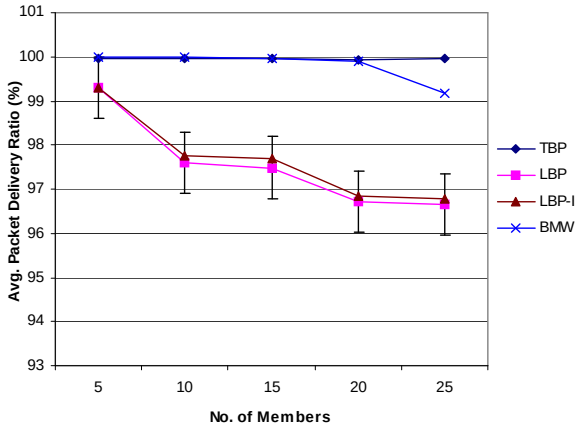

Figure 37. Reliability in the presence of noise (7 dB SNR).

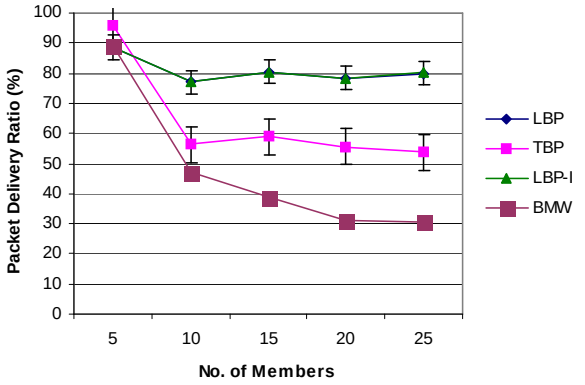

Figure 38. Reliability in the presence of noise (5 dB SNR).

varying PDR. In fact, some members did not receive any multicast packets. LBP-I had σ of only up to 1.77% indicating that its members faced less variability in PDR compared to LBP but far more than TBP ($\sigma = 0$).

1.3. Reliability in the Presence of Mobility. We used the scenario in Fig. 36(b). Except sources, all stations were mobile. Care was taken to ensure that each group always had at least one member. The purpose was to study the effect of mobile stations straying into an area of data exchange[18]. Importantly, this study captures the spurious

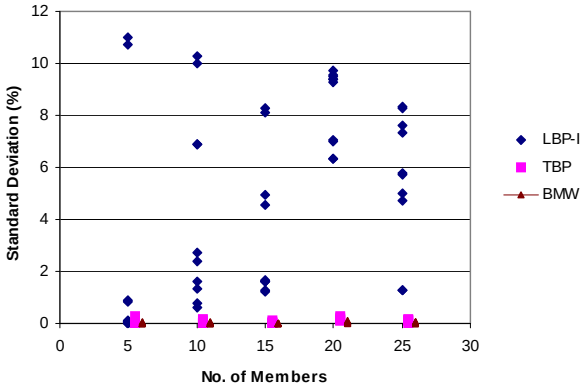

Figure 39. Fairness of reliability in WLANs.

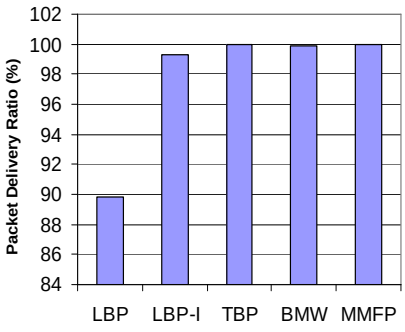

Figure 40. Reliability in Ad Hoc Networks. Results do not include losses due to queue overflows.

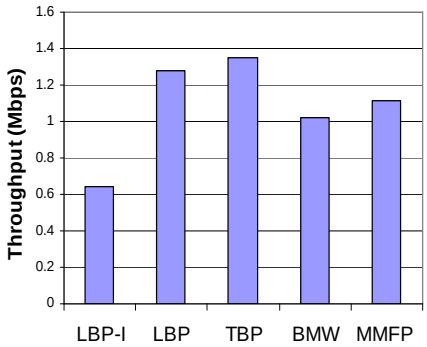

Figure 41. Saturation throughput per non-overlapping group.

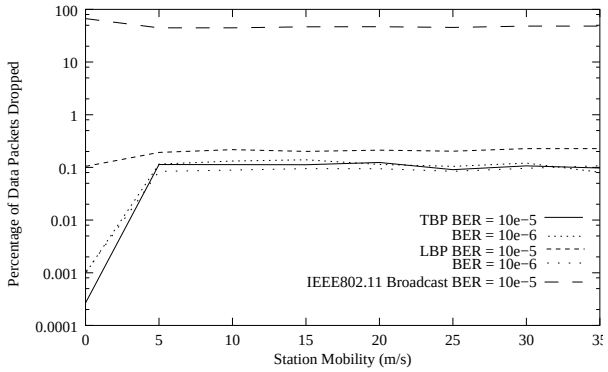

Figure 42. Effect of station mobility on data reliability.

NAK problem of TBP. We ignored the capture effect to see the contribution of mobility on reliability. This was done by setting a high capture threshold in the simulations (in reality, this cannot be done).

Fig. 42 shows that none of the protocols studied provide 100% reliability. TBP drops 0.1% of the packets while the modified LBP-I drops 0.2%. This difference is statistically insignificant. For comparison, the IEEE 802.11 multicast drops as much as 40% of the data packets under similar traffic conditions.

2. Throughput

2.1. Throughput of non-overlapping groups. Fig. 36(c) shows two non-overlapping groups whose transmissions do not affect each other. Fig. 41 shows that throughput per group of LBP-I is about half that of TBP because of its conservative $\mathcal{M}HTP$ prevention - only one group can transmit at a time. With TBP and the regular LBP, both groups can use the channel simultaneously.

2.2. Throughput of overlapping groups. Fig. 36(d) shows overlapping groups that affect one or more of the others. Results are shown in Fig. 43. LBP and LBP-I have

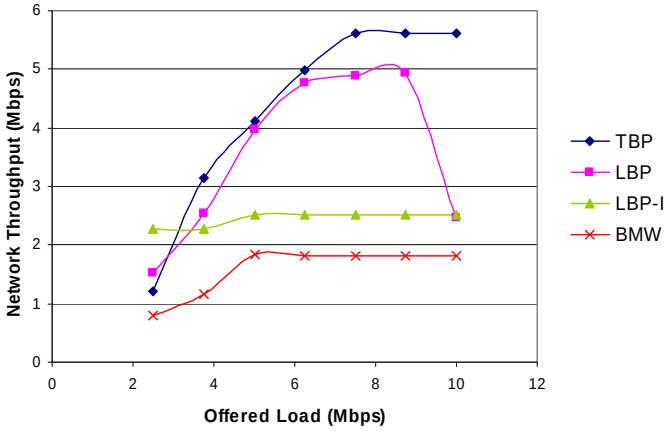

Figure 43. Throughput in an ad hoc network.

lower throughput than TBP due to different reasons: LBP due to collisions; LBP-I due to non-optimal blocking. Throughput of LBP initially rises with offered load but as the load is increased the probability of packet collision also increases and LBP's throughput falls. On the other hand, TBP and LBP-I reach and stay at their respective saturation throughputs since they avoid collisions. The throughput of TBP is much higher than that of LBP-I due to its optimal $\mathcal{MHTP}$ solution. BMW has the least throughput due to two reasons: (1) it does not take advantage of the wireless broadcast medium for the given topology (since members belonging to multiple groups can be active only in one group at any given time), (2) members that belong to multiple groups become bottlenecks (a multicast session is blocked until the member being polled becomes available).

2.3. Throughput in the presence of member mobility. The objective was to study the effect of membership change on throughput and is a good indication of throughput available to the higher layers in a mobile ad hoc network. Fig. 36(a) shows the scenario.

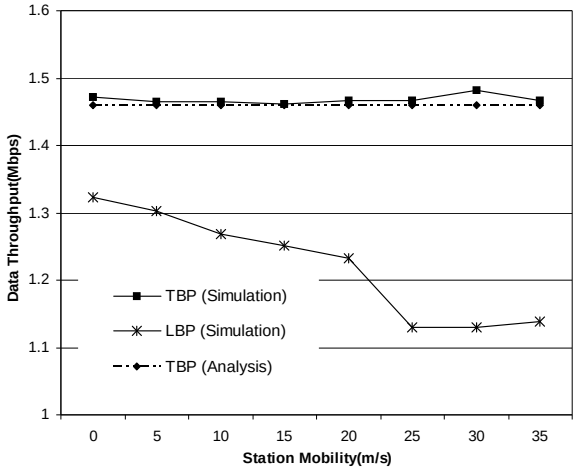

Figure 44. Throughput of TBP and LBP as a function of station mobility.

Fig. 44 shows the relative throughputs of TBP and LBP in a 2Mbps channel. We ensured that at least one member was in the source's coverage area so that LBP always has a leader. LBP's throughput gradually decreases with increasing station mobility because the probability of needing a leader re-selections increases. For each re-selection, time is taken to detect that the leader is missing, and select a new leader. TBP's performance is also better in a static network primarily because of the more efficient error recovery mechanism discussed in Section 4. Also, the NAK is transmitted only when the data packet has errors and the channel becomes free immediately after a successful data transmission. In LBP, on the other hand, the channel becomes free only at the end of the ACK transmission. Finally, control packets in LBP, such as CTS and ACK, are themselves prone to error resulting in unnecessary retransmissions. The figure also shows that the TBP throughput obtained through simulation agrees very well with that obtained through mathematical analysis.

2.4. Throughput-Reliability Tradeoff. Fig. 45 shows the tradeoff between reliability and throughput for MMFP with 15 members. k is the minimum number of members

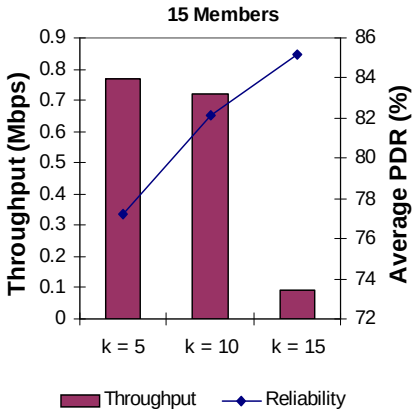

Figure 45. Throughput-Reliability tradeoff of MMFP.

that must receive the data correctly for the transmission to be considered successful. As k is increased, the average PDR increases making the multicast more reliable, but the throughput decreases.

Fig. 46 shows the throughput-reliability tradeoff for varying numbers of members. The relationship is as was predicted in Chapter 6 (see Fig. 30(b)). As explained earlier, the throughput initially falls and then rises because the probability that at least k members out of N receive the data correctly increases with N.

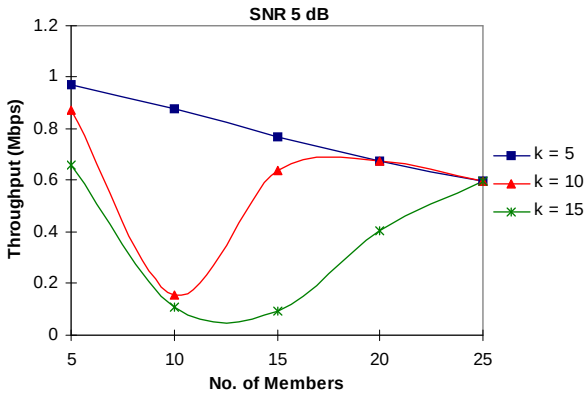

Figure 46. Throughput of MMFP as a function of reliability. k is the minimum number of members that must receive the data correctly.

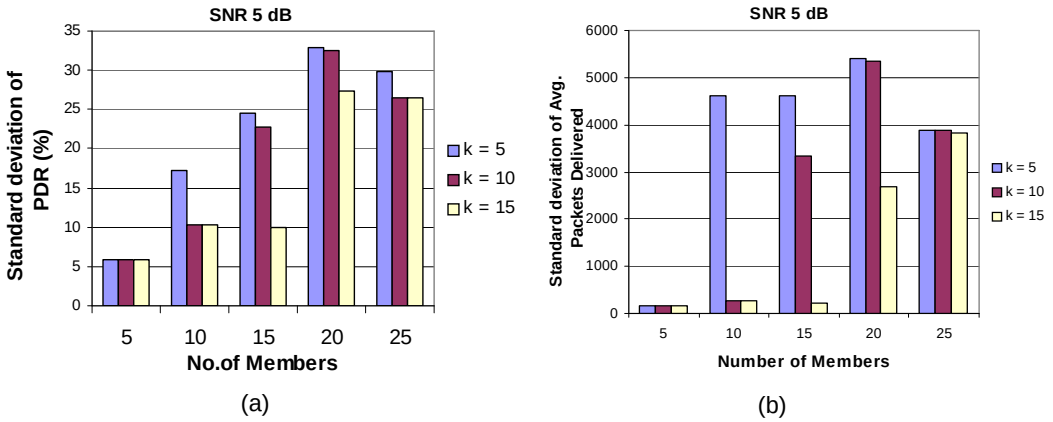

Figure 47. Fairness of reliability of MMFP as a function of k, the minimum number of members that must receive the data correctly: (a) standard deviation in PDR, (b) standard deviation in the absolute number of packets delivered.

Fig. 47 shows the cost of throughput in terms of fairness of reliability. Part (a) of the figure shows that the standard deviation in the average PDR decreases with increasing k. This implies that when throughput is increased by reducing k, there is a direct cost paid in terms of fairness of reliability. Fig. 47(b) shows the same results as a standard deviation of the number of packets received by members rather than the PDR (recall that the PDR normalizes the average number of packets delivered by the number of packets transmitted). Importantly, since a higher throughput means a larger number of data packets are transmitted, Fig. 47(b) shows that a greater number of packets are lost.

3. Average Delay

Since we are specifically interested in the delay overhead of multicast protocols due to reliability, our simulations avoid delays due to channel contention between stations and collisions. This is done by studying delay for a static network with only one active transmission. Fig. 36(a) shows the scenario we used for studying the delay.

Fig. 48 shows the delay performance of the protocols. For SNR of 7 dB or better, LBP and LBP-I have slightly less packet delay compared to TBP while BMW suffers the worst delay. In poor channel conditions such as 5 dB, LBP and LBP-I outperform TBP primarily because of unfairness - the protocols deliver packets to members that are closer to the source and ignore far off members. At 5 dB, BMW dropped most packets due to queue overflow and hence its average delay was not plotted for that SNR value.

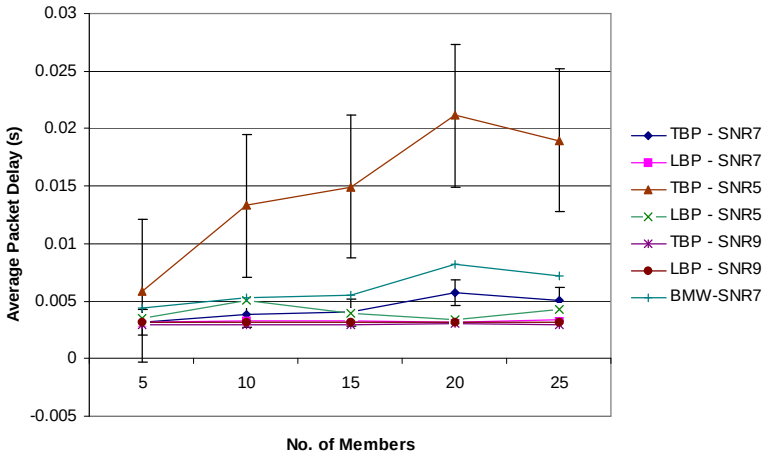

Figure 48. Delay increases with a drop in SNR.

Fig. 49 shows the reliability-delay tradeoff for MMFP with 15 members. k is the minimum number of members that must receive the data correctly for a transmission to be considered successful. As k is increased, the average PDR increases and so does the delay. Therefore an application must be willing to accept high delays in order to obtain better reliability.

Fig. 50 shows the relationship multicast group size and delay of MMFP for various values of k. It seen that delay depends both on k and on the ratio $\frac{k}{N}$, where N is the number of group members. The delay is higher when the ratio $\frac{k}{N}$ is small. This is because the probability that at least k members receive the data correctly increases as N increases.

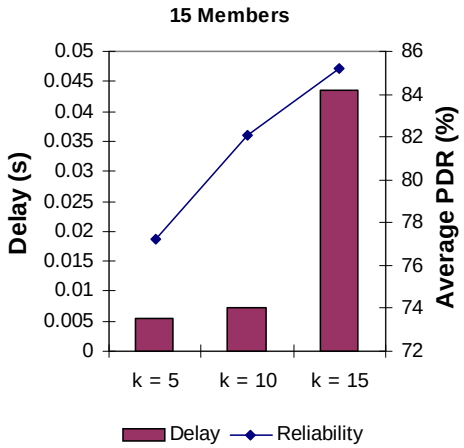

Figure 49. Delay-Reliability tradeoff of MMFP.

4. Discussion

Simulations results have established the importance of preventing $\mathcal{MHTP}$ throughput-optimally; reliability is affected when $\mathcal{MHTP}$ is not prevented (e.g. LBP), while throughput is affected when the $\mathcal{MHTP}$ solution is not optimal (e.g. LBP-I). TBP performed the best on both counts due to its optimal solution.

The results also show that a single feedback mechanism is not suitable for all application requirements of throughput, reliability and delay. While BMW and TBP provide high levels of reliability, their throughput is severely affected under poor channel conditions. Less reliable protocols such as LBP and LBP-I performed better under poor channel conditions (5 dB SNR). In general, negative feedback protocols (LBP, LBP-I and TBP) had a better throughput compared to positive feedback protocols (BMW and MMFP). Also, even though BMW and MMFP are reliable MACs, the delay incurred due to recovery made them more prone to packet loss resulting from queue overflows at the Link Layer.

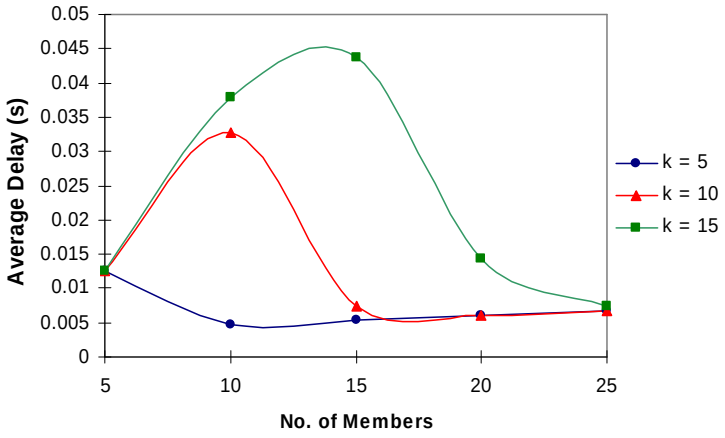

Figure 50. Delay of MMFP as a function of reliability (k).

The relationship between reliability and throughput (via parameter k) predicted in our analytic study was confirmed in our simulation results. Further, it was observed that the fairness of reliability is also affected by changes in the value of k.

CHAPTER 8

CONCLUSIONS AND FUTURE WORK

The objective of this research was to understand and develop throughput-optimal reliable multicasting at the MAC level. A comprehensive theoretic, analytic, and simulative study was undertaken to identify underlying principles of reliable multicasting.

The theoretic study identified and investigated three major challenges affecting MAC level multicast reliability: $\mathcal{M}HTP$, $\mathcal{F}IP$ and the increase of transmission error probability with group size. In order to focus attention on the challenges, the assumed error model in the theoretic study restricted packet loss only to collisions. We showed that one-hop blocking by all members provides a throughput-optimal solution to $\mathcal{M}HTP$. Of the feedback mechanisms studied Positive Group suffered from capture effect, Positive Individual was not scalable, and Negative Group was not feasible. In general, negative feedback does not have the ability to provide fine-grained information on which members received the data correctly and which did not. Since no one feedback mechanism was both scalable, reliable and fine-grained, we concluded that the choice of feedback mechanism depended on application requirements. Finally, we showed that the probability of retransmission can be reduce by prohibiting members that received the data correctly from successive retransmissions for that data packet.

Our analytic study relaxed the error model to include packet corruption due to channel errors. We showed through analysis and modeling that a tradeoff exists between reliability on one hand, and throughput and delay on the other. Importantly, a single protocol configuration is not suitable for all types of applications. An application requiring high reliability must be willing to accept lower throughput and higher delay while real-time applications must accept lower levels of reliability. Since multiple applications may run on the same station, it is important to develop a MAC that allows applications to dynamically configure service parameters such as reliability, throughput and delay. The underlying principals of the tradeoff presented in this work will aid protocol designers in making MAC design and configuration choices most suited for a desired application.

Based on our analysis, we proposed improvements to an existing protocol and two new protocols. The protocols are meant to address different application requirements of throughput and reliability. LBP-I improves on LBP and enables it to work in ad hoc networks. It is simple to implement, is suitable for high-throughput applications in WLANs, and provides stable throughput even under poor channel conditions. Its reliability improvement comes at the cost of throughput in ad hoc networks. TBP provides greater reliability in both WLANs as well as ad hoc networks and is scalable to large group sizes. It optimally prevents $\mathcal{MHTP}$ using busy tones and overcomes $\mathcal{FIP}$ using information of channel state during a specific period for feedback. Its main drawback is that its throughput is unstable under poor channel conditions. Finally, MMFP was proposed for applications that require fine-grained information on which members received multicast data correctly. Also, MMFP allows control of the tradeoff between reliability and throughput. However it is not scalable to large group sizes.

We performed simulations with an error model that accounts for distance between stations, interference, capture, channel noise and station mobility. Five protocols were simulated using *ns-2*. Results of reliability, throughput and delay agree with the analysis of the fundamental problems, and are in accordance with the tradeoffs predicted by the throughput and delay models. TBP is suitable for delay-tolerant applications requiring high reliability. LBP-I on the other hand may be used for real-time applications that do not require high levels of reliability or fairness.

This work demonstrated the tradeoff between multicast throughput and reliability, and proposed MMFP - a flexible protocol that can be configured to operate at any desired point along the tradeoff. However since MMFP is not scalable to large groups, a new protocol must be developed. Future work must include the following:

1. Since several applications may run on any given station, it is still unclear whether allowing application control of the MAC will provide gains that will justify the increased complexity entailed in cross-layer optimization.

2. If gains can be made by cross-layered optimization, we must develop a scalable multicast protocol that will allow applications to choose the desired operating point along the reliability-throughput tradeoff curve.

3. It has been shown that link-level reliability can improved end-to-end reliability and goodput. However, it is unclear whether those advantages exist for multicast traffic. Since improving reliability actually reduces link-level throughput, its effect on end-to-end throughput must be further investigated.

REFERENCES

[1] *Network simulator - ns-2*, 2002, Available via `http://www.isi.edu/nsnam/ns/`; accessed on June 23, 2005.

[2] J.C. Arnbak, *Capacity of slotted ALOHA in Rayleigh fading channels*, IEEE Journal on Selected Areas in Computing **5** (1987), 261–269.

[3] C. Bettstetter, G. Resta, and P. Santi, *The node distribution of the random waypoint mobility model for wireless ad hoc networks*, IEEE Transactions on Mobile Computing **2** (2003), no. 3, 257–269.

[4] G. Bianchi, *Performance analysis of the IEEE 802.11 distributed coordination function*, IEEE Journal on Selected Areas in Communications **18** (2000), no. 3, 535–547.

[5] K. Birman, M. Hayden, O. Ozkasap, Z. Xiao, M. Budiu, and Y. Minsky, *Bimodal multicast*, ACM Transactions on Computer Systems **17** (1999), no. 2, 41–88.

[6] E. Bommaiah, M. Liu, A. McAuley, and R. Talpade, *AMRoute: Ad hoc Multicast Routing protocol*, Internet Draft (work in progress) (1998).

[7] C. Brown, *Nonlinear neural nets smooth WiFi packets*, EETimes (2004), Available via `http://www.eetimes.com/article/showArticle.jhtml?articleId=19501137`; accessed on Nov. 4, 2005.

[8] B. Cain, S. Deering, and et al., *RFC 3376 - Internet Group Management Protocol*, (2002).

[9] M.M. Carvalho and J.J. Garcia-Luna-Aceves, *Delay analysis of IEEE 802.11 in single-hop networks*, IEEE International Conference on Network Protocols (ICNP 03) (Atlanta, Georgia), November 2003.

[10] ———, *A scalable model for channel access protocols in multihop ad hoc networks*, Proceedings of MobiCom'04 (Philadelphia, USA), September 2004.

[11] J. Chang and N.F. Maxemchuk, *Reliable broadcast protocols*, ACM Transactions on Computer Systems **2** (1984), no. 3, 251–273.

[12] P. Chaporkar and S. Sarkar, *On-line optimal wireless multicast*, WiOpt 2004 (Univ. of Cambridge, UK), March 2004.

[13] K. Cheun and S. Kim, *Joint delay-power capture in spread spectrum packet radio networks*, IEEE Transactions on Communications **46** (1998), no. 4, 450–453.

[14] C-Y. Chiu, E.H. Wu, and G-H. Chen, *A reliable and efficient MAC layer broadcast (multicast) protocol for mobile ad hoc networks*, Proceedings of Globecom 2004, 2004, pp. 2802–2807.

[15] I. Chlamtac, A.D. Myers, and V.R. Syrotiuk, *An adaptive Medium Acces Control (MAC) protocol for reliable broadcast in wireless networks*, Proceedings of IEEE International Conference on Communications, June 2000.

[16] IEEE Computer Society LAN MAN Standard Committee, *Wireless LAN medium access control MAC and physical layer specifications, IEEE Std 802.11.*, The Institute of Electrical and Electronics Engineers (1999).

[17] D.H. Davis and S. Gronemeyer, *Performance of slotted ALOHA random access with delay capture and randomized time of arrival*, IEEE Transactions on Communications **28** (1980), 703–710.

[18] J. Deng and Z. Haas, *Dual Busy Tone Multiple Access (DBTMA): A new medium access control for packet radio networks*, Proceedings of IEEE ICUPC (Florence, Italy), vol. 1, October 1998, pp. 973–977.

[19] P. Ding, J. Holliday, and A. Celik, *A leader based priority ring reliable multicast in WLANs*, Communication Systems and Networks (Marbella, Spain), September 2004.

[20] P. Eugster, R. Guerraoui, S. B. Handurukande, and P. Kouznetsov, *Lightweight probabilistic broadcast*, ACM Transactions on Computer Systems **21** (2003), no. 4, 341374.

[21] H. Fisher, *Multicast issues for collaborative virtual environments*, IEEE Computer Graphics and Applications (2002), 68–75.

[22] C.L. Fullmer and J.J. Garcia-Luna-Aceves, *Solutions to hidden terminal problems in wireless networks*, Proceedings of the ACM SIGCOMM '97 conference on Applications, technologies, architectures, and protocols for computer communication, ACM Press, 1997, pp. 39–49.

[23] A.J. Ganesh, A.-M. Kermarrec, and L. Massoulie, *Reliable probabilistic communication in large-scale information dissemination systems*, Tech. Report MSR-TR-2000-105, Microsoft Research, October 2000.

[24] J.J. Garcia-Luna-Aceves and E.L. Madruga, *The Core Assisted Mesh Protocol*, IEEE Journal on Selected Areas in Computing, Special Issue on Ad-Hoc Networks **17** (1999), no. 8, 1380–1394.

[25] D. Goldsman and G. Tokol, *Output analysis procedures for computer simulations*, Proceedings of the 32nd Winter Simulation Conference (Orlando, Florida), December 2000, pp. 39–45.

[26] H. Gossain, N. Nandiraju, K. Anand, and D. P. Agrawal, *Supporting MAC layer multicast in IEEE 802.11 based MANETs: Issues and solutions*, Proceedings of the 29th Annual IEEE International Conference on Local Computer Networks, 2004.

[27] S.K.S. Gupta, V. Shankar, and S. Lalwani, *Reliable multicast MAC protocol for wireless LANs*, IEEE International Conference on Communications (Anchorage, Alaska), May 2003.

[28] S.K.S. Gupta and P.K. Srimani, *Cored-Based Tree with Forwarding Regions(CBT-FR): A protocol for reliable multicasting in mobile ad hoc networks*, Journal of Parallel and Distributed Computing, Special Issue on Routing **61** (2000), no. 9, 1249–1277.

[29] C. Huitema, *The case for packet level FEC*, Proceedings of the 5th Workshop on Protocols for High Speed Networks (Sophia Antipolis, France), October 1996, pp. 109–120.

[30] D.B. Johnson and D.A. Maltz, *Dynamic source routing in ad hoc wireless networks*, Mobile Computing (1996), 153–181.

[31] P. Karn, *MACA - a new channel access method for packet radio*, Proceedings of the ARRL/CRRL Amatuer Radio Computer Networking Conference, 1990, pp. 134–140.

[32] S. Khurana, A. Kahol, S.K.S. Gupta, and P.K. Srimani, *Performance evaluation of distributed co-ordination function for IEEE 802.11 wireless LAN protocol in presence of mobile and hidden terminals*, Modeling, Analysis and Simulation of Computer and Telecommunication Systems, 1999. Proceedings. 7th International Symposium on, 1999, pp. 40–47.

[33] A. Kochut, A. Vasan, U. Shankar, and A. Agrawala, *Sniffing out the correct physical layer capture model in 802.11b*, Proceedings of the 12th IEEE International Conference on Network Protocols (ICNP'04) (Berlin, Germany), October 2004.

[34] K.Tang and M. Gerla, *MAC reliable broadcast in ad hoc networks*, MILCOM 2001 - IEEE Military Communications Conference, vol. 2, October 2001, pp. 1009–1014.

[35] J. Kuri and S.K. Kasera, *Reliable multicast in multi-access wireless LANs*, Wireless Networks **7** (2001), no. 4, 359–369.

[36] S. Kurkowski, T. Camp, and M.Colagrosso, *MANET simulation studies: The Icredibles*, Mobile Computing and Communications Review **9** (2005), no. 4, 50–61.

[37] S.-J. Lee, M. Gerla, and C.C. Chiang, *On-demand multicast routing protocol*, Proceedings of IEEE WCNC'99 (New Orleans, LA), September 1999, pp. 1298–1304.

[38] J.C. Lin and S. Paul, *RMTP: A reliable multicast transport protocol*, INFOCOM (San Francisco, CA), March 1996, pp. 1414–1424.

[39] A. Arora M. Sun, L. Huang and T. Lai, *Reliable MAC layer multicast in IEEE 802.11 wireless networks*, International Conference on Parallel Processing (ICPP'02) (Vancouver, B.C., Canada), August 2002.

[40] M.K. Marina, G.D. Kondylis, and U.C. Kozat, *RBRP: A robust broadcast reservation protocol for mobile ad hoc networks*, Proceedings of IEEE International Conference on Communications, vol. 3, June 2001, pp. 878–885.

[41] L.E. Miller and B.J. Kwak, *Cumulative acknowledgement multicast repetition policy for wireless LANs or ad hoc network clusters*, International Conference on Communications (ICC 2002), April 2002.

[42] R. Mud, J. Boer, A. Kamerman, H. Van Driest, W. Diepenstraten, R. Kopmeiners, and H. Von Bokhorst, *Wireless LAN with enhanced capture provision*, 1999, US patent no. US5987033.

[43] T. Nilsson, G. Wikstrand, and J. Eriksson, *Early collision detection in CSMA/CA networks*, Proceedings of the Fourth IEEE Conference on Mobile and Wireless Communications Networks (MWCN 2002), 2002.

[44] J. Nonnenmacher, E.W. Biersack, and D. Towsley, *Parity based loss recovery for reliable multicast transmission*, Proceedings of SIGCOMM'97 (Cannes, France), September 1997, pp. 289–300.

[45] T. Ozaki, J.B. Kim, and T. Suda, *Bandwidth-efficient multicast routing protocol for ad-hoc networks*, Proceedings of IEEE ICCCN'99, 1999, pp. 10–17.

[46] E. Pagani and G.P. Rossi, *Reliable broadcast in mobile multihop packet networks*, Proceedings of 3rd ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM'97) (Budapest, Hungary), September 1997, pp. 34–42.

[47] J. Pereira, L. Rodrigues, and R. Oliveira, *Semantically reliable multicast: Definition, implementation, and performance evaluation*, IEEE Transactions of Computers **52** (2003), no. 2, 150–165.

[48] David C. Plummer, *RFC 826 - Ethernet Address Resolution Protocol: Or converting network protocol addresses to 48.bit ethernet address for transmission on ethernet hardware*, November 1982, Available via `http://www.faqs.org/rfcs/rfc826.html`; accessed August 2, 2005.

[49] J. Polastre, J. Hill, and D. Culler, *Versatile low power media access for wireless sensor networks*, SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems (New York, NY, USA), ACM Press, 2004, pp. 95–107.

[50] R. Ramaswami and K. Parhi, *Distributed scheduling of broadcasts in a radio network*, INFOCOM '89. Proceedings of the Eighth Annual Joint Conference of the IEEE Computer and Communications Societies (Ottawa, Ont., Canada), vol. 2, April 1989, pp. 497–504.

[51] T. S. Rappaport, *Wireless communications: Principles and practices*, Prentice Hall, 1996.

[52] L. Rizzo and L. Vicisano, *A reliable multicast data distribution protocol based on software fec techniques (rmdp)*, Proceedings of Fourth IEEE Workshop on the Architecture and Implementation of High Performance Communication Systems (HPCS'97) (Chalkidiki, Greece), June 1997.

[53] V. Shankar, *A medium access control protocol with reliable multicast support for wireless networks*, Master's thesis, Dept. of Comp. Sci. and Engg., Arizona State Univ., December 2002.

[54] S-T. Sheu, Y. Tsai, and J. Chen, *A highly reliable broadcast scheme for IEEE 802.11 multi-hop ad hoc networks*, In Proceedings of IEEE International Conference on Communications (ICC 2002), April 2002, pp. 610–615.

[55] W. Si and C. Li, *RMAC: A reliable multicast MAC protocol for wireless ad hoc networks*, Proceedings of the 2004 International Conference on Parallel Processing (ICPP 2004), August 2004.

[56] S. Singh and C. S. Raghavendra, *PAMAS: Power aware multi-access protocol with signalling for ad hoc networks*, ACM Computer Communication Review **28** (1998), no. 3, 5–26.

[57] K. Tang and M. Gerla, *MAC layer broadcast support in 802.11 wireless networks*, 21st Century Military Communications Conference Proceedings, MILCOM 2000 (McLean, VA), October 2000, pp. 544–548.

[58] ———, *Random access MAC for efficient broadcast support in ad hoc networks*, IEEE Wireless Communications and Networking Conference (Chicago, IL), September 2000, pp. 454–459.

[59] F.A. Tobagi and L. Kleinrock, *Packet switching in radio channels: Part II - the hidden terminal problem in carrier sense multiple-access modes and the busy-tone solution*, IEEE Transactions on Communications **23** (1975), no. 12, 1400–1416.

[60] J. Tourrilhes, *Robust broadcast: Improving the reliability of broadcast transmissions on CSMA/CA*, Proceedings of the Ninth International Symposium on Personal, Indoor and Mobile Radio Communications (New York), vol. 3, 1998, pp. 1111–5.

[61] D. Towsley, J. Kurose, and S. Pingali, *A comparison of sender-initiated and receiver-initiated reliable multicast protocols*, IEEE Journal on Selected Areas in Communications **15** (1997), no. 3, 398–406.

[62] C. Ware, J. Chicharo, and T. Wysocki, *Modeling capture behaviour in IEEE 802.11 radio modems*, IEEE International Conference on Telecommunications, 2001.

[63] C. Wu and V. Li, *Receiver-initiated busy-tone multiple access in packet radio networks*, Proceedings of the ACM workshop on Frontiers in computer communications technology, ACM Press, 1988, pp. 336–342.

[64] C.W. Wu, Y.C. Tay, and C.-K. Toh, *Ad hoc Multicast Routing protocol utilizing Increasing id-numberS (AMRIS) functional specification*, Internet-Draft, draft-ietf-manet-amris-spec-00.txt (Work in progress) (1998).

[65] Shih-Lin Wu, Yu-Chee Tseng, and Jang-Ping Sheu, *Intelligent medium access for mobile ad hoc networks with busy tones and power control*, IEEE Journal on Selected Areas on Communication **18** (2000), no. 9, 1647–1657.

[66] J. Xie, A. Das, S. Nandi, and A. K. Gupta, *Improving the reliability of IEEE 802.11 broadcast scheme for multicasting in mobile ad hoc networks*, Proceedings of IEEE Wireless Communications and Networking Conference (WCNC), 2005, pp. 126–131.

[67] K. Xu, M. Gerla, and S. Bae, *How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks?*, Globecom, 2002.

[68] M. Yamamoto, J. Kurose, D. Towsley, and H. Ikeda, *A delay analysis of sender-initiated and receiver-initiated reliable multicast protocols*, Proceedings of the IEEE Infocom 97 (Japan), 1997.

[69] Roy D. Yates and David J. Goodman, *Probability and stochastic processes: A friendly introduction for Electrical and Computer Engineers*, John Wiley and Sons, 1998.

[70] R. Yavatkar, J. Griffoen, and M. Sudan, *A reliable dissemination protocol for interactive collaborative applications*, ACM Multimedia, 1995, pp. 333–344.

[71] Seong-Won. Yuk and Dong-Ho. Cho, *Parity-based reliable multicast method for wireless LAN environments*, Proceedings of IEEE VTC'99, October 1999, pp. 1217–1221.

[72] C. Zhu and M.S. Corson, *A Five-Phase Reservation Protocol (FPRP) for mobile ad hoc networks*, Wireless Networks **7** (2001), no. 4, 371–384.

APPENDIX A

PARAMETER VALUES USED IN SIMULATIONS

Table 6. Channel Holding Times

Parameter	**Description**	**Value**
T_{SLOT}	Slot time	20 μs
T_{SIFS}	Short Interframe Spacing	10 μs
T_{DIFS}	DCF Inter-frame Spacing	50 μs
T_{FT}	Feedback Time	Protocol Dependent

Table 7. Protocol Parameters

Parameter	**Description**	**Value**
L_{DATA}	Data Packet Payload Length	512 bytes
L_{RTS}	RTS Packet Length	Protocol Dependent
L_{CTS}	CTS Packet Length	31 bytes
L_{NAK}	NAK Packet Length	30 bytes
L_{ACK}	ACK Packet Length	30 bytes
Queue Length		25
CWMin_	Minimum Contention Window Size	31
CWMax_	Maximum Contention Window Size	1023
PreambleLength_		144 bit
PLCPHeaderLength_		48 bits
PLCPDataRate_		1 Mbps
RTSThreshold_		150 bytes
ShortRetryLimit_	Retransmission Limit	7
LongRetryLimit_	Retransmission Limit	4
FEC Strength	Error Correction Capability	1 bit correction

Simulation code may be obtained either through our website (http://impact.asu.edu) or by writing to the author at vikram.shankar@ieee.org.

Table 8. Antenna and Channel Parameters

Parameter	**Description**	**Value**
C	Total Channel Capacity	2Mbps
P_T	Transmission Power	0.2818 W
CSThresh_	Carrier Sense Threshold	3.652e-10 W
RXThresh_	Receive Threshold	3.650e-10 W
CPThresh_	Capture Threshold	10
L_	Antenna Loss	1
G_t	Antenna Transmitter Gain	1
G_r	Antenna Receiver Gain	1
$freq$_	Antenna Operating Frequency	2.4GHz

Table 9. Simulation Environment

Parameter	**Value**
Simulator	ns-2 version 2.28
Platform	Cygwin (Setup version 2.457.2.2)
Tcl	version 8.4.5
Tk	version 8.4.5
OTcl	version 1.9
Tclcl	version 1-1.16
PRNG seed	variable (1-10)
Operating System	Windows XP Professional