A MEDIUM ACCESS CONTROL PROTOCOL WITH RELIABLE MULTICAST SUPPORT FOR WIRELESS NETWORKS

by

Vikram Shankar

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

ARIZONA STATE UNIVERSITY

December 2002

A MEDIUM ACCESS CONTROL PROTOCOL WITH RELIABLE MULTICAST SUPPORT FOR WIRELESS NETWORKS

by

Vikram Shankar

has been approved

December 2002

APPROVED:

____________________, Chair

____

____

Supervisory Committee

ACCEPTED:

____

Department Chair

____

Dean, Graduate College

ABSTRACT

Reliable multicast in wireless applications is gaining importance with the development in technology. Applications like multicast file transfer, distributed computing, chat and whiteboard applications need reliability. However, due to mobility and wireless channel characteristics, obtaining reliability in data transfer is a difficult and challenging task. IEEE 802.11 does not support reliable multicast due to its inability to exchange RTS/CTS and ACKS with multiple recipients. However, several Medium Access Control (MAC) layer protocols have been proposed that provide reliable multicast. For example, the Leader-Based, Probability-Based and Delay-Based Protocols have been proposed. These protocols work around the problem of multiple CTSs/ACKs colliding by providing ways to have only one of the multicast recipient nodes respond with a CTS or an ACK. These protocols perform well in low mobility wireless LANs but the performance degenerates as the mobility of nodes increases. In this thesis, we discuss the inherent drawbacks of these protocols and provide an alternative approach. We present an extension to the IEEE 802.11 MAC layer protocol to provide link level reliability to both unicast as well as multicast data communications. The extension is NAK based and uses tones, instead of conventional packets, to signal a NAK. We also incorporate Dual Tones to prevent an incoming mobile node from interrupting an ongoing transmission. Simulation results suggest that our MAC performs better than Leader Based Protocol in terms of both data throughput as well as reliability.

Dedicated to my dear parents,

Dr. Shiva Shankaran and Bhanumathi Shankar.

You shaped my life.

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.

I would like to place on record my appreciation for my committee members, Dr. Arunabha Sen and Dr. Joseph Hui. Inspite of their busy schedules, they have taken the time to review my thesis and have provided useful feedback.

Suresh Lalwani has been involved in the early development of the Core-based Reliable Multicast Protocol (CRMP). This work is part of the CRMP project. He has contributed much to this project and his drive and hardwork have been a source of inspiration to me.

Aarthi Natarajan has also been working on developing a reliable MAC for mobile ad hoc networks. My discussions with her have always been fruitful and I have learnt a lot from her. The coding for the Leader Based Protocol simulations are entirely hers. I thank Valliappan Annamalai for helping me fix bugs on several occasions.

Last but not the least, I would like to thank my labmates at the Mobile Computing and Wireless Networking Laboratory. Our group meetings have been intellectually stimulating. Our numerous cups of coffee together have also been stimulating.

The work has been partially funded by NSF grants ANI-00196156, ANI-0086020, ANI-0123980 and DGE-9870720.

TABLE OF CONTENTS

Page

LIST OF FIGURES

CHAPTER 1

Introduction

Improvements in technology have made mobile devices smaller and more capable than ever before. The marketplace is flooded with mobile devices. Among them are Personal Digital Assistants, laptops, Global Positioning Systems and cell phones. An interesting trend that has been observed in recent years is that designers of these systems are moving more and more toward connecting these devices to one and another and to the Internet. Technologies have been developed that allow the networking of such devices. For example, the Bluetooth standard [4] was specifically developed to enable communication between devices in a personal area network. On a slightly wider scale, we have the IEEE 802.11 standard that allows devices such as laptops to connect to a local area network. The next level of mobile communication is the cellular system and then finally we have the satellite-based communication that covers most of the world.

Of late, researchers have been showing much interest in mobile computing. This area of research is concerned with how to make devices and services available to a user on the move. There are exciting applications that have been envisaged for mobile computing. A common example is that of conference attendees exchanging information with each other through their laptops. Other examples include location-specific content display on handheld devices, distributed chat via PDAs, and industry-specific hand-held field devices.

Applications in mobile computing have varying communication needs. Some applications such as surfing the net from a laptop require reliable unicast and do not specify any latency requirements. On the other hand, we have mobile multimedia applications that do not require reliability but have stringent latency bounds. Multimedia applications may either be directed to one device or to a group of devices and hence require multicasting. We also have applications such as distributed chat over PDAs that have the dual requirements of both reliability as well as group communication. Our research is targetted at these applications that require both reliability as well as group communication.

Applications that require both reliability as well as group communication may be as trivial as a chat application or may be applied to more critical domains such as assisting in search/rescue. In search/rescue operations, for example, there is typically a group of search units that fan out over a geographical area. These units need to share information on promising search paths and hot trails. Usually one unit is assigned the responsibility of command/control but information also needs to be gathered from and disseminated to the other units in the group. The reliability with which information is transmitted may be critical to the progress of the search operation.

In the next section we give a brief overview of mobile devices, the environments they operate in and their unique characteristics and constraints. In section 2 we ennumerate some of the benefits of addressing reliable multicast issues at the MAC layer. We then give an overview of this thesis, including our results.

1. Mobile Devices and Networks

Mobile devices are usually characterized by their resource constraints. It is essential for mobile devices to be lightweight and hence a lot of resources that are usually expected in desktop computers are usually scaled-down or are even missing. Such devices usually have lower processor speeds and less memory. Also, since these devices have to be carried around, they usually come with their own source of power - usually a battery. There has not been a significant improvement in battery technology over the past three decades [12] and battery lifetime continues to be a serious constraint in mobile computing. Further, in order to provide users with maximum mobility, such devices use wireless communication rather than depend on wired connections. Wireless links are typically less reliable and have very low bandwidth. Further, since most devices use omnidirectional antennas, different communications can interfere with each other. In order to reduce interference between unrelated transmissions, wireless communications require more stringent medium access controls. When we design a communication protocol for mobile devices, we have to take all these factors into account.

Communications for mobile devices may or may not be supported by an infrastructure. By infrastructure, we mean a base station that coordinates medium access among the devices that are withing wireless range of itself. Apart from access coordination, such base stations may also act as gateways for devices in its service area to connect to devices in other service areas or to the internet. For example, the IEEE 802.11 has wireless access points to which a wireless device may connect. The base stations are usually fixed and connected to a wired network. When a mobile node leaves the service area of an access point, it simply accesses the network through another access point. We discuss IEEE 802.11 in greater detail in chapter 3.

Ad hoc networks are not supported by fixed base stations or any kind of infrastructure. The network consists of nodes that cooperate with each other to route messages within the network. Ad hoc networks are typically characterized by their low bandwidth availability, low energy availability, and constrained device resources.

In both the types of networks we have discussed above, the devices may be mobile. In the case of infrastructure based networks, such mobility will cause the set of devices in a service area to change and a device may go out of range of a base station and enter the service area of another base station. Depending on the protocol, the device may or may not need to register with the base new station. In the case of ad hoc networks, such mobility may cause a device's neighbors to change. In both the cases, there is a possibility of a transmitting device to stray into the wireless range of another ongoing transmission and cause a loss of data. The MAC layer that we have developed may be applied to infrastructure-based as well as mobile ad hoc networks (MANETs).

2. Reliable Multicast Communication

There are several ways in which we can send the same message to several recipient nodes. A simple method would be to send unicast packets to each of the nodes in the group. But such a naive method does not take advantage of the fact that several of the recipient nodes may share the same route up to a point. Therefore, we are basically transmitting redundant data along the same links.

Another extreme in communicating common messages would be to flood the network with the messages, or what we call broadcasting. Broadcasting has been shown to be resilient to network faults. However, broadcast messages are sent even to nodes that are not interested in the message and there is a huge and unnecessary consumption of bandwidth.

When data has to be sent to multiple recipients, multicast incurs less network cost when compared to either broadcast or unicast to individual group members. Multicasting limits the transmission of redundant data and saves bandwidth as well as energy, two resources that are scarce in a network of wireless nodes.

Most research on mobile 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 [7] [11] [8] [15] [3] [21]. 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.

In this thesis we are interested in reliable multicast support at the MAC level i.e. reliable transfer of data across single-hop wireless links. Such reliability is desirable because it reduces end-to-end delays by facilitating local error recovery. MAC level reliability does not guarantee end-to-end reliability and we assume that such reliability, if required, is built into a higher layer such as the network or transport layers. In particular, we do not deal with loss of packets that result from a node moving out of the transmission range of the sender.

3. Overview of Thesis

We propose an extension to IEEE 802.11(henceforth called 802.11MX) that supports reliable multicast. Multicast reliability is built into the protocol by the use of NAKs. As mentioned earlier, the major problem with reliable multicast is that several recipient nodes may respond to an RTS or a data packet at the same time. We avoid this problem by using a tone to signal a NAK or NCTS rather than having a NAK/NCTS packet as such. The advantage of this method is that any number of recipient nodes can signal a NAK or a NCTS at the same time and since it is only a tone, collision does not affect it. Towsley et al [20][25] have shown that for multicasting applications, a NAK based reliability scheme fares better than an ACK based scheme both in terms of complexity as well as throughput.

Deng et al [5] have shown that the use of RTS/CTS is not sufficient to avoid packet collisions. Their results show that in highly mobile environments there may be up to 60% of packets loss. They introduced the Dual Busy Tone Multiple Access to reduce packet collisions. By incorporating dual busy tones into our protocol we attempt to reduce the probability of multicast packets being corrupted due to collision and hence we need a lesser number of retransmissions. By reducing the number of retransmissions, we can increase the throughput.

Our MAC protocol is suitable for both infrastructure-based mobile networks as well as ad-hoc mobile networks. In the case of infrastructure-based networks, our system would consist of a base station (that would be the source), and several mobile nodes. In the case of an ad hoc mobile network, our system simply consists of mobile nodes that are within range of each other.

We have analyzed and simulated our protocol and compared its performance against the Leader Based Protocol (LBP) [10]. Simulation results suggest that our protocol performs better than LBP both in terms of data throughput as well as reliability. While only 0.1% of data packets go undelivered in our protocol, LBP losses up to twice as many data packets. Further, unlike in LBP, the throughput of our protocol does not degenerate with increasing node mobility. We have validated our simulation results for throughput by comparing it against a mathematical analysis of the throughput for our MAC.

CHAPTER 2

Related Work

There are two problems that are unique to wireless communication:

1. *Hidden Terminal Problem:* This problem is unique to the wireless medium. With reference to figure 1, when node A wants to send data to node B, it senses the channel around itself before sending the data packet. It does not know whether there is a transmission around the receiving node or not. For example, it would not know if node C is also transmitting. If node C is indeed transmitting, a packet collision will occur at B.

2. *Exposed Terminal Problem:* With reference to figure 2, node B wants to transmit to node A while node C wants to transmit to node D. It is alright for such transmissions to take place because there will be no collisions at the receivers A and D. However, in this case, node B or C (whichever senses the channel later) will find the channel busy and defer its transmission. This problem causes a drop in the bandwidth utilization.

The hidden terminal problem causes packet collisions while the exposed terminal problem causes an inefficient use of bandwidth. In [9], P. Karn proposed the

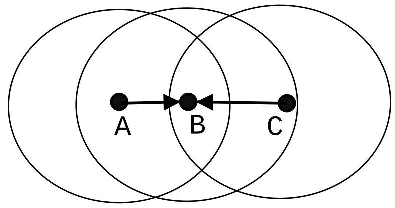

Figure 1. **Hidden terminal problem**

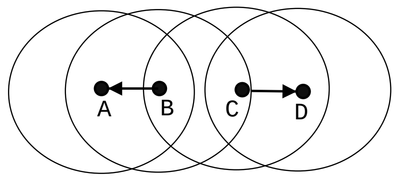

Figure 2. **Exposed terminal problem**

Medium Access with Collision Avoidance (MACA) protocol to limit the effect of these two problems. In MACA, when a node wants to transmit, it transmits a Request-to-Send (RTS) packet. The RTS packet contains information regarding the destination for the yet to be transmitted data packet and the expected duration of the transmission. All the nodes that hear the RTS defer their transmissions if they have any. The node to which the RTS is addressed will send a Clear-to-Send (CTS) packet if it is in a position to receive the data packet. Apart from the node that sent the RTS packet, all the nodes that hear the CTS packet will defer their own transmission until the current transmisison is over. When the sender of the RTS receives the CTS, it sends the data packet. Following this procedure will take care of the hidden terminal problem to an extent.

There is a possibility that more than one node sends an RTS at the same time. This will result in a collision and hence the intended recipients will not respond with the CTSs. When the CTS is not received within a predetermined amount of time, the nodes will back off for a random amount of time and resend the RTS packet. RTS collisions are indeed a waste of channel bandwidth but considering that RTSs are typically much smaller than data packets, such collisions are still better than data packet collisions.

In order to limit the effect of the exposed terminal problem, P. Karn suggested that if a node hears only the RTS but not the CTS, it may assume that the receiving node is out of its range. It can then go ahead and make any transmissions it wants.

The Floor Acquisition Multiple Access (FAMA) proposed in [6] is a class of protocols that attempt to fuse Carrier Sense Multiple Access (CSMA) and MACA. In the original MACA protocol, a node does not sense the channel before transmitting the RTS. Transmissions are deferred only if the RTS or CTS is received correctly and understood. So, if for some reason, a node does not hear either an RTS or a CTS, it may start transmitting even while another data transmission is going on. To limit the effect of this, the authors propose that non-persistent carrier sensing should be done before the RTS is sent. Non-persistent CSMA was chosen because it has been shown that non-persistent CSMA has higher throughput when compared to p-persistent CSMA during high loads and only slightly lesser throughput under low loads. FAMA also allows the transmission of packet bursts rather just one data packet, once the channel has been aquired. In order to avoid channel hogging, the burst size is subject to a maximum limit.

The MACAW [2] protocol is an extension of the MACA protocol. The authors have suggested several improvements including a change in the backoff algorithm and the introduction of new types of control packets. In the original MACA protocol, the contention window is doubled every time a collision occurs and is halved when a transmission is successful. This causes a wide fluctuation in the size of the contention window. To avoid this, the authors suggest that the window size be increased by 1.5 times but reduced only by one slot time. The call this mechanism the Multiplicative Increase Linear Decrease (MILD). Another problem with the original backoff algorithm is its apparent unfairness. All nodes in an area do not have the same view of congestion. There is a difference in the contention windows sizes of the node that won the contention and the nodes that did not. The node that won the contention will have a smaller window than the ones that did not. As a result, the node that won a contention will almost always win the following contentions since it will only backoff for a shorter time. In order to avoid this unfairness, MACAW requires the winning node to inform the other nodes about its window size. This is done by placing the window size value in the packet to be transmitted. This way, all nodes in the neighborhood have the same window size.

In MACA, neighboring nodes that hear the RTS will delay their transmissions irrespective of whether the RTS/CTS dialog was successful or not. In case the dialog was unsuccessful, such delay is an unnecessary waste of bandwidth. MACAW introduces the DS packet to specifically inform the neighbors of the sending node whether the RTS/CTS dialog was successful or not. If a DS packet is not heard by the senders neigbbors, they can go ahead with their transmissions.

When there is heavy traffic in the network and the size of the data packet is much larger than the RTS packet, then the time during which the channel can be sensed to be free is very small. A node that wants to transmit will have to sense the channel during that small period and the chances of that happening are also very small. To help a node find this small period, MACAW introduces the RRTS packet. This packet is sent by the receiver to the sender informing the sender that the channel around the receiver is free and that the sender may transmit an RTS if it so desires.

Deng et al [5] have shown that inspite of using RTS/CTS, the probability of packet collisions in a MACA-based mobile network can be as high as 60%. Collisions are mainly due to nodes 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 nodes of an ongoing transmission.

The Dual Busy Tone Multiple Access (DBTMA) works as follows. Instead of having one common channel for all transmissions, the authors propose to have separate data and control channels. Control messages such as RTS/CTS are sent on the control channel. There are also two busy tones associated with the control channel. The transmit busy tone BT_t indicates that a node in the vicinity is transmitting in the data channel. The receive busy tone BT_r indicates that a node in the vicinity is receiving data. The busy tones are transmitted for the entire duration of the transmission or reception in the data channel. When a node wants to transmit, it senses for the presence of BT_r. When no tone is detected, the node can go ahead and make a transmission without fear of collision. When the node starts transmitting data, it also starts transmitting the BT_t tone. On the receiving side, the node senses the channel for transmit busy tone. If a tone is sensed, it indicates an ongoing transmission in the data channel that could interfere with its reception. In such a case, it will abstain from sending a CTS and the data transfer is postponed. Since the tones are transmitted for the entire duration of the data transfer, even nodes that stray into the transmission zone and missed the initial RTS/CTS exchange are alerted to the presence of an ongoing transmission. The presence of the receive busy tone eliminates the hidden terminal problem. If a node senses the transmit busy tone but not the receive busy tone, then it can safely transmit data. This takes care of the exposed terminal problem.

Several of the MAC protocols for the wireless medium that have been proposed so far are based on the seminal MACA protocol. The original MACA protocol was designed for reliability in unicast transmissions. When we apply MACA-based protocols to reliable multicast communication, we have to be careful to avoid CTS collisions. CTS collisions will occur when several of the nodes receiving the multicast data send a CTS at the same time in response to an RTS sent by the source.

Examples of MACA-based protocols that use dual busy tones include [19] and [22]. PAMAS [19] is a medium access protocol that was developed with the objective of conserving energy in an ad hoc network. Energy is conserved by powering off a node when it is not actively transmitting or receiving a packet. PAMAS is based on the MACA protocol. However, unlike in MACA, control packets are sent over a separate signalling channel rather than the channel on which the data is sent. By an exchange of control packets and BusyTones, a node decides if it should power off. Simulation results provided by the authors indicate that there are power savings of between 10% and 60%. PAMAS was designed for unicast communication and does not support reliable multicasting. Further, the BusyTone that is used in PAMAS is simply a control packet and is prone to collisions. In our MAC protocol, on the other hand, the tones that we use for NAKing are sine wave signals like those proposed by [5]. The advantage of using sine wave tones rather than a control packet tones is that tones from two or more nodes can be transmitted at the same time without corrupting each other.

In [22], the authors propose a MAC protocol that uses both power control as well as the concept of dual tones to increase channel utilization. The idea behind the protocol is that if a node uses just enough transmit power to reach the receiving node, then the range of its transmission interference will be lesser than what it would be if the node always transmitted at the maximum transmit power. Like in [5], the authors use Dual Busy Tones to prevent a node from transmitting in the vicinity of an ongoing transmission. Analysis and simulation results presented by the authors suggest that their protocol does indeed improve channel utilization. Apart from increased channel utilization, power control also has the additional advantages of battery energy saving and reduction in co-channel interference. This protocol is also based on the MACA protocol. However, the control packets are transmitted on a separate control channel rather than on the common data channel. This MAC protocol does not support reliable multicast.

In [23], the authors extend their Dynamic Channel Allocation (DCA) protocol proposed in [24]. By applying the concept of power control to DCA, the authors have shown that interference is reduced and hence there is better channel reuse and the radio spectrum is used more efficiently. Here again, this protocol is designed for unicast communication and does not support reliable multicasting.

IEEE 802.11 does not support reliable multicast[1]. However there are several extensions to IEEE 802.11 that have been proposed for multicasting in wireless networks. Most of them use a combination of ACKs and NAKs to attain a degree of reliability. The problem with the use of ACKs during multicasting is that only one recipient should send an ACK to avoid a collision. Three methods have been proposed in [10] to handle this problem.

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

2. **Probability-Based Protocol (PBP):** When a data packet is received, recipient nodes sent an ACK packet with some probability p. Here again it is possible that two nodes 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 nodes. Only the leader is allowed to send an ACK. The other nodes are silent if they received the data packet correctly or else they send a NAK.

Analytical results in [10] suggest that LBP performs better than both DBP and PBP. However, LBP itself has a major drawback when applied to mobile nodes. A new leader will have to be selected each time the current one leaves the cell. As the mobility of nodes increases, this problem becomes prominent.

Towsley et al [20][25] have shown that for multicasting applications, a NAK based reliability scheme fares better than an ACK based scheme both in terms of complexity as well as throughput.

CHAPTER 3

Overview of IEEE 802.11 MAC Layer

The IEEE 802.11 MAC and PHY specification [1] is an industry standard that is being widely used for wireless LANs. It belongs to a family of specifications that include the Ethernet (802.3) and token ring (802.5). These specifications are defined for different network topologies and for different communication media but they all provide a common interface to the layers above the MAC.

The MAC layer of the IEEE 802.11 standard supports reliable unicast communication. Though it does support multicast, it makes no attempt at reliability. The 802.11 MAC is usually used for infrastructure based networks but it is also capable of handling ad hoc networks.

Since the MAC protocol we propose is an extension of the IEEE 802.11, it is imperative that we understand the working of 802.11. In this chapter we give a brief overview of the protocol. We also ennumerate some of the problems with the protocol and why it cannot be used directly for multicast in mobile ad hoc networks.

1. System Architecture

The IEEE 802.11 is designed for both the infrastructure-based as well as the ad hoc networks. In the infrastructure-based network, the mobile nodes are connected to a set of access points. An access point together with the nodes connnected to it is called the Basic Service Set (BSS). Communication within the BSS is through the wireless medium. The access points can also be connected with one another via a distributed system to form one network. This system with multiple BSSs is then called an Extended Service Set (ESS). The ESS may itself have a portal which will connect it to other networks.

There is no access point support in the ad hoc network. Instead, the mobile devices within wireless range of each other form a Basic Service Set.

2. Protocol Architecture

The standard specifies the physical layer and the MAC sublayer of the data link layer. The logical link control (LLC) sublayer of the data link layer provides a common interface between the 802.x family of protocols with the layers above them.

The physical layer is subdivided into the physical layer convergence protocol (PLCP) and the physical medium dependent (PMD) sublayer. The PLCP provides a carrier sense signal in the form of clear channel assessment (CCA). It also provides a common service access point (SAP) that is independent of the transmission technology used. Transmission technologies may include Code Division Multiple Access, Frequency-Hop Spread Spectrum and Infra-red. The PMD is dependent on the actual transmission technology used and is concerned with the modulation and encoding/decoding of signals. The physical layer also has a management module called the PHY management. The PHY management does channel tuning and also maintains the PHY management information base (MIB).

Functionalities of the MAC layer include medium access, fragmentation of user data and encryption. At the MAC level, there is a MAC management module that supports the roaming between access points by allowing association and reassociation of a device to access points. This module also handles authentication mechanisms, encryption, synchronization and power management. Finally, it also maintains the MAC management information base.

3. Medium Access Control

The MAC layer offers different types of access mechanisms. The mandatory basic access mechanism is based on CSMA/CA. This mechanism is offered in both the infrastructure-based network mode as well as the ad hoc network mode. Then there is an extension of the basic mechanism that attempts to address the hidden terminal problem. The third is a contention-free polling mechanism for time-bound service. The first two mechanisms are referred to as the **distributed coordination function (DCF)** and the third mechanism is referred to as the **point coordination function (PCF)**. PCF is not offered in the ad hoc network mode. These mechanisms are also called the *distributed foundation wireless medium access control* (DFWMAC).

The access mechanism is controlled by three parameters:

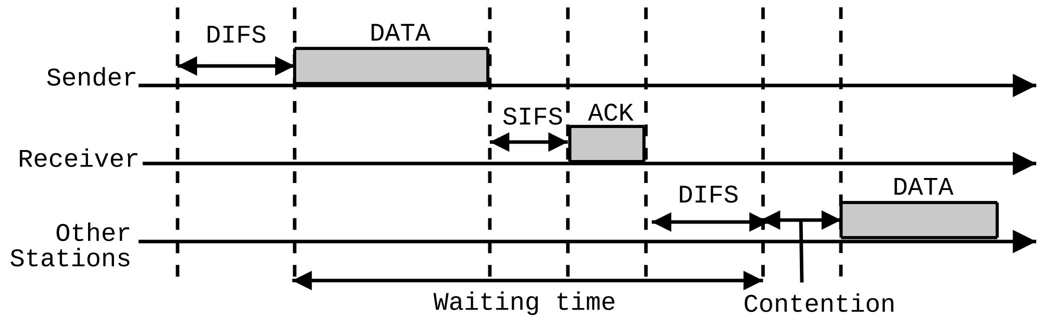

Figure 3. **Basic reliable unicast in IEEE 802.11**

DCF Inter-frame Spacing (DIFS): This is the longest wait period and corresponds to the lowest priority for medium access.

PCF Inter-frame Spacing (PIFS): The PIFS wait period is between that of the DIFS and SIFS. It is used for time-bounded service.

Short Inter-frame Spacing (SIFS): This is the shortest wait period and corresponds to the highest priority for medium access. It is usually used for sending control messages such ACKs.

Of the three access mechanisms mentioned above, only the first two are relevant to this thesis and we will describe only those two.

3.1. Basic DFWMAC-DCF using CSMA-CA. The basic medium access mechanism is as shown in figure 3. When a node wants to transmit, it senses the channel. If the channel is free for a DIFS duration of time, the node can start transmission of the data packet. If the data packet is received without error, the destination node waits for SIFS duration before sending an ACK packet to the sender. No packet is sent if the data was received in error. All nodes will have to go through the channel access mechanism irrespective of whether the packet to be transferred is fresh data or is just a retransmission of an older packet.

When a node wants to transmit a packet but finds the medium busy, it will have to go through a contention phase. Each node will pick a random duration within a contention window. The random duration is in multiples of a slot time where the slot time is derived from the medium propagation delay, transmitter delay and other physical layer dependent paramenters. Each node will then start a backoff timer for the chosen random duration. During this time, if the channel becomes busy, it implies that some other node has won this round of contention. This node will then stop its backoff timer and wait for the channel to become free again for a DIFS duration of time. When the channel does become free for DIFS duration, the node starts its backoff timer again. If the channel is still free when the timer expires, it implies that the node has won this round of contention and can start its transmission immediately.

It is clear to see that in this mechanism, it is possible for two or more nodes to sense a free channel at the same time. If this happens, the nodes will start transmitting their data packets at about the same time and a collision results. If a collision does occur, the nodes will have to backoff a random amount of time and contend for the channel once again. If a collision occurs once again, the nodes double the size of the contention window and choose a random time from this larger window. This doubling of the window each time a collision occurs during access is called exponential backoff. Exponential backoff allows shorter wait times during low channel load and avoids frequent collisions during high traffic. The contention window starts with a size of 7 slots and can grow to a maximum size of 255 slots.

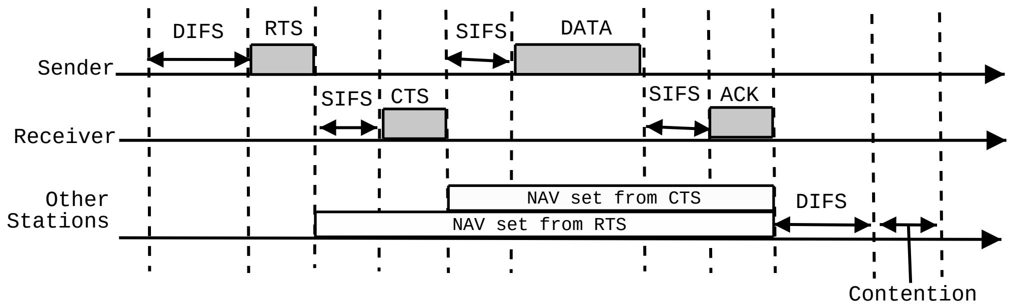

Figure 4. **Reliable unicast in IEEE 802.11 with RTS/CTS exchange**

The problem with this basic access mechanism is that it is highly prone to the hidden and exposed terminal problems.

3.2. DFWMAC-DCF with RTS/CTS Extension. In order to reduce the effect of the hidden and exposed terminal problems, a MACA-type extension was added to the IEEE 802.11 specifications. Two new control packets have been added: Request-To-Send (RTS) and Clear-To-Send (CTS). A dialog is initiated with an RTS/CTS exchange. The use of this extension is optional but every node has to have the functionality to react appropriately when an RTS or CTS is received.

This extended medium access mechanism is shown in figure 4. A node wanting to transmit a data packet senses the channel. If the channel is free for DIFS duration of time, the node sends an RTS packet. The RTS packet contains information on the intended destination of the data packet and the expected duration of the data transmission that will follow. When the destination node receives the RTS packet, it waits for SIFS duration and responds with a CTS packet if it is ready to receive the data packet. The CTS packet contains the source address, destination address and the duration of the data transmission that is to follow. On receiving the CTS packet from the destination node, the source node waits for SIFS duration before it starts transmitting the data packet. If the data packet is received without error, the destination node sends an ACK packet after a SIFS duration.

Each node maintains what is called a Network Allocation Vector (NAV). When a node that is not the destination of the current transmission hears an RTS or CTS, it sets its NAV in accordance with the duration specified in the RTS/CTS packets. The NAV indicates the earliest time at which the network becomes free. This mechanism reserves the medium exclusively for one sender and is therefore also referred to as virtual reservation scheme. It is to be noted that the set of nodes that hear the RTS need not be the same set of nodes that hear the CTS.

Here again, like in the basic access mechanism, there is a possibility that two or more nodes find the channel free at the same time and start transmission at about the same time. This causes a collision. Contention is handled here in exactly the same way as was described for the basic access scheme.

The RTS/CTS dialog is overheard by neighboring nodes and informs them of the impending data transmission. Nodes that can disrupt a transmission should be in range of the receiver. They are alerted by the CTS packet and hence the hidden terminal problem is taken care of on the receiver side. We have to take care of the hidden terminal problem on the sender side as well because the sender will expect to hear an ACK packet from the receiver. The problem on this side is handled by the RTS packet.

The RTS/CTS dialog is effective only if the neighboring nodes hear them. In a static wireless network, this is usually not a problem. But in a mobile network, there is a very good chance that a node may stray into the range of an ongoing transmission after the RTS/CTS exchange.

3.3. Multicast in IEEE 802.11. This standard supports only unreliable multicast. Channel access is done without the RTS/CTS dialog and receivers send neither an ACK nor a NAK on reception of the data packet. At the MAC level, multicast is equivalent to a broadcast.

CHAPTER 4

Protocol Description

In this chapter we give a detailed description of our 802.11MX protocol. First we make a few definitions that will help in describing the protocol. For clarity, we also provide state diagrams and an algorithmic description of the protocol.

1. Definitions

1. **Inter-Frame Spacings:** The Short Inter-frame Spacing (SIFS) and the Distributed Inter-Frame Spacing (DIFS) are the same as defined in the IEEE 802.11 specifications.

2. **Not-ready Response Time (NRT):** NRT is the time within which a node can send a tone to indicate that it is not yet ready to receive data or that the last packet it received was in error.

3. **Tone:** A tone is a narrow-bandwidth sinusoidal wave signal. A tone is transmitted on a frequency that is outside of the channel that is used for packet transmission.

4. **Busy Tone:** A busy tone is transmitted by nodes to indicate that they are currently busy with some communication. A node wanting to transmit data will do so only if it does not sense the presence of a busy tone.

5. **NCTS Tone:** This tone is transmitted in response to an RTS and indicates that the node is not yet ready to receive data or that the RTS packet was received in error.

6. **NAK Tone:** This tone is transmitted by nodes that received the data packet in error.

2. System Model

Our system consists of a *cluster* of nodes that are within communications range of a *cluster-leader*. In an infrastructure-based network, a *cluster-leader* would correspond to a base station or an access point. In an ad hoc network, the *cluster-leader* is any node that is the *source* of multicast information or is forwarding multicast information. In the context of ad hoc networks, a *cluster* is the set of all nodes that are within transmission range of this node. All nodes in the *cluster* will receive multicast data at the same time.

We assume that an antenna is capable of picking up both data bits as well as busy tones. Each node will have two antennas. One antenna is required for transmitting and receiving the busy tones while the other antenna is for transmitting and receiving data packets and NAK tones. The antennas are not capable of sensing the channel and transmitting at the same time.

We assume that multicast session information is maintained at a higher level in the OSI protocol stack. Multicast addresses are resolved at a higher level as well. At the MAC level, multicast and broadcast data are treated similarly and one-hop reliability is provided to both.

Packet transmissions are prone to collisions with other packet transmissions. If the packets are of comparable power, then both the packets are dropped. However, if the power of one packet is much lower than the other one, "capture" effect takes place wherein the packet of the higher power is received with or without error. Apart from packet collisions, we also assume a channel bit error rate of 10^{-5}. This error may be introduced for one or more of several reasons including multipath propogation, interference from adjacent channels or from distant transmissions. In our study of reliability, we do not account for packet losses due to a node moving out of range of the sender. Recovery from such packet losses should be done at the transport layer and is not a MAC layer function.

3. Description of Algorithm

Each node has access to four channels - data channel, NAK channel and the two busy tone channels. The data channel and the NAK channel share a common transceiver while the busy tone channels share a transceiver. All packets including control packets are transmitted in the data channel. The NAK channel is a narrow-band channel used solely for the purpose of transmitting the NAK tone. Of the two busy tone channels, one is for transmitting the Transmit Busy Tone and the other is

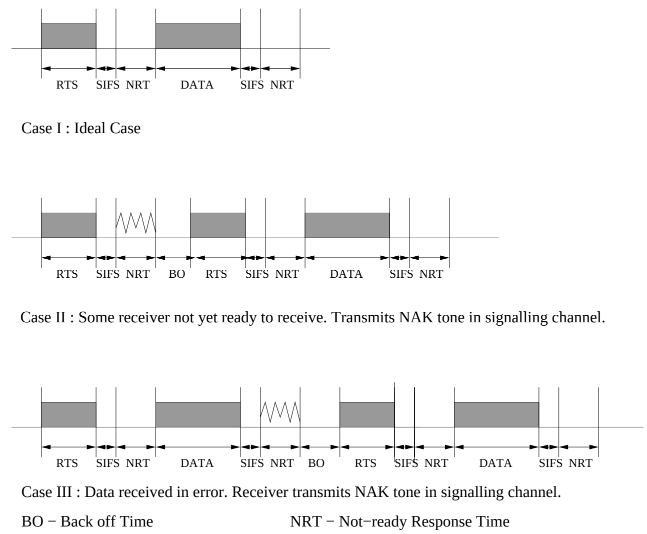

Figure 5. **Slot diagrams of the protocol**

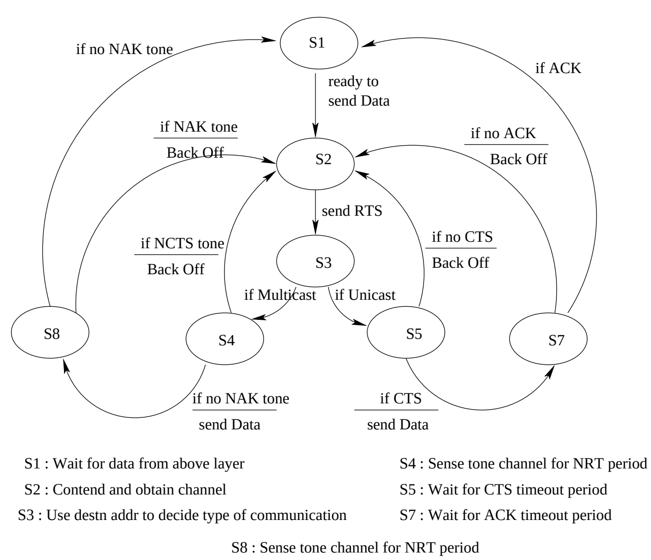

Figure 6. **State diagram of the sender (base station).**

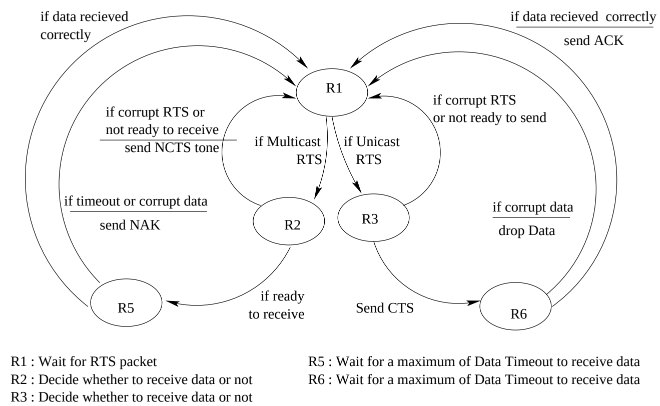

Figure 7. **State diagram of the multicast receiver.**

for the Receive Busy Tone. The busy tone channels are also narrowband channels.

We have designed our protocol around the IEEE 802.11 MAC. When a node wants to transmit any data (state S1 of fig. 6), it senses the channel. If the channel is free for DIFS period, the node resumes its backoff timer. If the channel is still free when the backoff timer expires, the node can start transmission. In unicast transmission, the node sends an RTS and expects to receive a CTS (states $S2 \rightarrow S3 \rightarrow S5$). If a CTS is received, the node transmits the data packet and waits for an ACK (states $S5 \rightarrow S7$). If the ACK is received, the data transmission was successful (return to state $S1$). Otherwise, the node goes through the channel access process all over again. For unicast data, we simply use the IEEE 802.11 specification.

When multicast data is to be sent, channel access is done just like in unicast. But when the RTS is sent (states $S2 \rightarrow S3 \rightarrow S4$), the sender node does not expect to receive a CTS packet. Instead, it listens on the signalling channel to see if any node is transmitting a NCTS tone. If no such tone is sensed, the sender begins transmission of the multicast data packet (states $S4 \rightarrow S8$). At the end of the packet transmission, the node senses the signalling channel again to see if any node is transmitting a NAK tone. If there is no NAK tone, the sender assumes that the data transmission was successful (states $S8 \rightarrow S1$).

On the receiver side, when a node receives an RTS it checks to see if the RTS is addressed to it or if the address is that of a multicast group to which it belongs. If the RTS is specifically addressed to it, then this is a unicast transmission and the receiver node is expected to send a CTS if it is ready to receive (states $R1 \rightarrow R3 \rightarrow R4$ of fig. 7). If the RTS is addressed to a multicast group to which this node belongs, then this node will have to prepare to receive a multicast data packet (state $R1 \rightarrow R2 \rightarrow R5$). If it is not ready to receive a packet, it sends a NCTS tone in the signalling channel and terminates the packet exchange (state $R1 \rightarrow R2 \rightarrow R1$). If the node is ready to receive, it starts a timer. If the data packet is not received before the timer expires, it probably means that some other recepient node has sent a NCTS and terminated the packet exchange. The node will send a NAK and go back to its idle state. If a data packet is received before the timeout, the recepient nodes check to see if the packet was received without error. If the packet was received without error or with Forward Error Correction (FEC) correctable errors, the packet is accepted and sent to the higher layers ($R5 \rightarrow R1$). If the packet is received with uncorrectable errors, the packet is dropped and a NAK tone is transmitted in the signalling channel (states $R5 \rightarrow R1$). As the response is only a tone, it does not matter if the NAKs from multiple recepients collide. In case of a NAK, the *source* will have to re-aquire the channel again by sending an RTS. Fig 5 shows the working of the protocol.

Nodes that receive the RTS packet but are not interested in the transmission will simply update their Network Allocation Vectors for virtual carrier sensing. The slot diagram for the protocol is shown in fig. 5.

Following is an algorithm for the working of the protocol:

[1] **Source:** Acquire Channel

[2] **Source:** Send RTS.

[3] **Recipients:**

 if interested in data

  if unicast transmission **and** ready to receive

    Send CTS.

  else

   if ready to receive

    Prepare to receive data

    Start Timer

    Start Busy Tone

   else

    Send NCTS tone.

 Update Network Allocation Vector.

[4] **Source:**

 If unicast transmission

  if received CTS

Send data.

else

Goto step [1].

else

Check for NCTS tone.

if no NCTS tone

Start Busy tone.

Send Data.

else

Goto step [1].

[5] **Recipients:**

if received data

if No error in data

Accept data.

if unicast reception

Send ACK. Goto step [7].

else

Drop data.

if multicast reception

Send NAK tone.

else

Goto step [7].

Terminate busy tone.

[6] **Source:**

if unicast transmission

if Did not receive ACK

Goto step [1].

else

Goto step [7].

else

if NAK tone present

Goto step [1].

else

Goto step [7].

Disable busy tone.

[7] **Done**.

4. Incorporation of Dual Busy Tones

In a network of mobile nodes, an RTS-CTS exchange does not guarantee that the channel is available solely to one transmission at a time. For instance, a node that was originally out of communications range of a pair of nodes may not have heard the RTS-CTS exchange. Such a node may stray into the range of the communicating nodes and may sense the channel during the long silence between an RTS and data packet transmission (for multicast data). This silence could be mistaken for the channel being free and the straying node may start transmitting, thereby possibly causing a collision. The receiving nodes will detect such packet corruption and send a NAK causing a retransmission of the data. Hence the algorithm described so far is resilient enough to handle such straying nodes. However, we note that retransmission has an adverse effect on channel throughput. We can prevent such collisions by using the Dual Busy Tones, a mechanism proposed by [5].

After sending the RTS, if the *source* does not detect a tone in the signalling channels during the NRT period, it will assume that the channel is available. After the NRT period, both the *source* and the receivers will transmit a busy tone in the corresponding busy-tone channel to indicate that a data transmission is in progress in data channel. Any node that wants to transmit will sense this tone to find that the channel is busy. The busy tones are disabled at the end of transmission. In case the transmission is cut short due to an NCTS, the transmitting node will disable its tone on sensing the NCTS tone. The other nodes will disable their tones when their timers expire without any data being received.

CHAPTER 5

Analysis and Simulation

1. Mathematical Analysis

In our analysis, we study the performance of our protocol in terms of Mean Channel Holding Time. For a given data packet size, the higher the mean channel holding time, the lower the throughput. We have analysed our protocol for both noiseless 1.1.1 as well as noisy 1.1.2 channels. Though [10] has analysed the Leader-Based protocol with respect to mean channel holding time they have not accounted for delays that are caused due to the leader leaving the group. In subsection 1.2, we take into account leader mobility while analyzing the mean channel holding time of the Leader-Based Protocol. In order to simplify our analysis, we assume that there is only a single *source* and there is no other communications apart from the multicast communication. That is to say, we assume that there is no channel contention.

1.1. Analysis of our Protocol.

1.1.1. *Performance in a Noiseless Channel.* If we assume an error-free channel, data will have to be transmitted only once. Then the time required to transmit a

Table 1. **Definition of symbols**

T_{bo}	Back off time in *sec.*
T_{rts}	RTS packet transmission delay in *sec.*
T_{ifs}	Inter frame spacing between two transmissions in *sec.*
T_{data}	Data packet transmission delay (including propagation delay) in *sec.*
R	Transmit radius of the leader in *meters*.
N_l	Number of nodes leaving the cluster during one data pkt transmission.
N_e	Number of nodes entering the cluster during one data pkt transmission.
d	Nodal density in $nodes/m^2$.
V	Velocity of mobile nodes in m/s
X	Average number of data packet retransmissions each node entering a cluster requests.
T	Time to transmit one data packet after accounting for retransmissions and mobility, in *sec.*

packet is given by:

$$T_t = T_{rts} + T_{ifs} + T_{nrt} + T_{data} + T_{ifs} + T_{nrt}. \tag{5.1}$$

The useful data transmitted is T_{data} and the overhead is $T_t - T_{data}$, where T_t is the channel hold time for transmission of each packet in an error-free channel. Therefore the utility of the system, without error, is:

$$U = \frac{T_{data}}{T_t}. \tag{5.2}$$

1.1.2. *Performance in an Error-Prone Channel.* Let us assume that the probability of error in a RTS packet is P_{erts} and the probability of error in a data packet is P_{edata}. We have:

$$T_1 = T_{bo} + T_{rts} + T_{ifs} + T_{nrt}, \tag{5.3}$$

$$T_2 = T_{data} + T_{ifs} + T_{nrt}. \tag{5.4}$$

Then the total time to transmit RTS is given by:

$$\begin{aligned}
T_{erts} &= (1-P_{erts})T_1 + P_{erts}[T_1 + (1-P_{erts})T_1 \\
&\quad +P_{erts}[T_1 + (1-P_{erts})T_1 + P_{erts}[... \\
&= (1-P_{erts})T_1[\frac{1}{1-P_{erts}}] + P_{erts}T_1[\frac{1}{1-P_{erts}}] \\
T_{erts} &= \frac{T_1}{1-P_{erts}}
\end{aligned} \tag{5.5}$$

and the total time (T_{te}) required to transmit a packet of data is given by:

$$\begin{aligned}
T_{te} &= T_{erts} + (1-P_{edata})T_2 + P_{edata}[T_{erts} \\
&\quad +T_2 + (1-P_{edata})T_2 + P_{edata}[...
\end{aligned} \tag{5.6}$$

The above equation is recursive because there is possibility of error even in the retransmissions. Expanding the above equation we get,

$$\begin{aligned}
T_{te} &= T_{erts} + (1-P_{edata})T_2 + P_{edata}(T_{erts}+T_2) \\
&\quad +P_{edata}(1-P_{edata})T_2 + P^2_{edata}(T_{erts}+T_2) + ..., \\
&= T_{erts} + (1-P_{edata})T_2[\frac{1}{1-P_{edata}}] + P_{edata}(T_{erts}+T_2)[\frac{1}{1-P_{edata}}], \\
&= \frac{T_{erts}+T_2}{1-P_{edata}} \\
T_{te} &= \frac{\left(\frac{T_1}{1-P_{erts}}\right)+T_2}{1-P_{edata}}.
\end{aligned} \tag{5.7}$$

The total time taken is T_{te} and time taken for actual data transmission is T_{data}.

therefore, the utilitization of the noisy channel will given by:

$$U_e = \frac{T_{data}}{\frac{\left(\frac{T_1}{1-P_{erts}}\right)+T_2}{1-P_{edata}}}. \tag{5.8}$$

1.2. Analysis of Leader-Based Protocol.

1.2.1. *Performance in a Noiseless Channel.* In the absence of mobility, the channel holding time is:

$$T_{tl} = T_{rts} + T_{cts} + T_{data} + T_{ack} \tag{5.9}$$

Let the probabilty that a node will move out of the cluster be P_{out} and let there be N nodes in the cluster. Therefore, the probability that a leader will move out of the cluster during a transmission is:

$$P_l = \frac{P_{out}}{N} \tag{5.10}$$

When the leader moves out of the cluster (or group), no one will send a CTS to any RTS that the source sends. When successive RTSs go unanswered, the source dissolves the group and the nodes interested in multicast data will have to resubscribe. The total time required to realize that the leader has left, dissolve the group and re-establish the communications is given by:

$$T_{ro} = n(T_{rts} + T_{cts}) + T_{gto} + T_{sub} \tag{5.11}$$

where T_{ro} is the overhead time associated with reorganizing the cluster and selecting a new leader and T_{gto} is the time it takes to realize that the leader has left and dissolve the group. Therefore, in presence of mobility, the mean channel holding time is given by:

$$T_{tlm} = (1 - P_l)T_{tl} + P_l\,(T_{ro} + T_{tlm}) \tag{5.12}$$

Solving the above equation, we will get:

$$T_{tlm} = T_{tl} + \left(\frac{P_l}{1 - P_l})\right) T_{ro} \tag{5.13}$$

1.2.2. *Performance in an Error-Prone Channel.* First, we will consider the case when there is no mobility. Let the probability of receiving a CTS in error be P_{ects} and let the probability of receiving an ACK in error be P_{eack}. The time required to send rts will be:

$$T_{erts} = (1 - P_{erts})T_{rts} + P_{erts}(T_{rts} + T_{erts}) \tag{5.14}$$

Solving the above equation, we obtain:

$$T_{erts} = \frac{T_{rts}}{1 - P_{erts}} \tag{5.15}$$

The time required till the completion of CTS transmission will be

$$T_{ects} = (1 - P_{ects})(T_{erts} + T_{cts}) + P_{ects}(T_{erts} + T_{cts} + T_{ects}) \tag{5.16}$$

Solving the above equation, we obtain:

$$\begin{aligned} T_{ects} &= \frac{T_{erts} + T_{cts}}{1 - P_{ects}} \\ &= \frac{\left(\frac{T_{rts}}{1-P_{erts}}\right) + T_{cts}}{1 - P_{ects}} \end{aligned} \tag{5.17}$$

If we proceed in the same manner, we can find out the time required till sending of ack, which is nothing but the mean channel holding time. the final eqaution for mean channel holding time will be:

$$T_{tle} = \frac{\left(\frac{\left(\frac{\left(\frac{T_{rts}}{1-P_{erts}}\right)+T_{cts}}{1P_{ects}}\right)+T_{data}}{1-P_{edata}}\right)+T_{ack}}{1-P_{eack}} \tag{5.18}$$

If we introduce the probabilty for leader node moving out of the cluster then mean channel holding time will be:

$$\begin{aligned} T_{tlme} &= (1-P_l)T_{tle} + P_l(T_{ro} + T_{tle}) \\ &= T_{tle} + \left(\frac{P_l}{1-P_l}\right)T_{ro} \end{aligned} \tag{5.19}$$

2. Simulation

We used network simulator *ns-2* [13] to simulate both our protocol as well as the Leader-Based Protocol. In order to study the throughput as well as the reliability of the protocols, we have run the simulations for two kinds of scenarios. For both the scenarios, we ran simulations for mobility ranging from 0 to 35 m/s in steps of 5 m/s. We used the Random Waypoint Model for node mobility. For each speed, we took the average of five simulation runs. For throughput study, each run was 500 seconds and for the reliability study, each run was the time taken for the multicast source to send 50,000 data packets.

2.1. Data Throughput. Fig. 8(a) shows the scenario we used for testing the data throughput of the protocols as a function of node mobility. Though we used a 2Mbps channel, the actual throughput is much lower for both the protocols because of overhead in terms of control packets, interframe spacings, random backoffs and packet retransmissions. In this scenario, we have only one transmiting node. There is no contention for the channel. Packet errors can only occur due to channel error. We have assumed a constant Bit Error Rate of 10^{-5}. Care was taken to ensure that there was atleast one node in the base station's transmission region. This was done so that the Leader-Based Protocol's throughput is not adversely affected due to the lack of a leader node.

In order to validate our simulations, we did a mathematical analysis of the throughput of 802.11MX. Here we provide only the final equations of the analysis. Equation 5.20 gives the channel holding time of 802.11MX.

$$\begin{aligned}
T_1 &= T_{DIFS} + T_{bo} + T_{RTS} + T_{SIFS} + T_{NCTS} \\
T_2 &= T_{SIFS} + T_{DATA} + T_{SIFS} + T_{NAK} \\
T_{CH-MX} &= \frac{T_1 + T_2(1 - L_{RTS}P_{err})}{(1 - L_{RTS}P_{err})(1 - L_{DATA}P_{err})}
\end{aligned} \tag{5.20}$$

$$\begin{aligned}
T_3 &= T_{DIFS} + T_{bo} + T_{RTS} + T_{SIFS} + T_{CTS} + T_{SIFS} \\
T_{tle} &= \frac{\left(\frac{\left(\frac{\left(\frac{T_3}{1-L_{RTS}P_{err}} \right) + T_{CTS} + T_{SIFS}}{1-L_{CTS}P_{err}} \right) + T_{DATA} + T_{SIFS}}{1-L_{DATA}P_{err}} \right) + T_{ACK} + T_{SIFS}}{1 - L_{ACK}P_{err}} \\
T_{CH-LBP} &= T_{tle} + \frac{nT_3P_L}{1 - P_{RF}}
\end{aligned} \tag{5.21}$$

where L_x is the length of packet x, T_{bo} is the backoff time, and P_{err} is the probability that a bit is in error, the inverse of bit error rate.

Fig. 9 shows the relative throughputs of 802.11MX and LBP. In order to give the reader a reference point to compare with, we have also plotted the throughput of IEEE 802.11 unicast. It is seen that the throughput for 802.11MX obtained through simulation agrees very well with the throughput obtained through mathematical analysis. It is also seen that LBP's throughput gradually decreases with increasing node mobility. This can be attributed to the fact that with increasing mobility, the probability of a leader moving out of the base station's transmission region increases. This increases the frequency of selection of a new leader. The time taken for the base station to detect that the leader is missing added with the time it takes to select a new leader has an adverse effect on the channel throughput. From the figure, it also appears that our multicast protocol has better throughput performance than IEEE 802.11 unicast. This is partly attributed to the fact that the NAK tone is transmitted out-of-band of the data channel and hence the data channel becomes free at the end of the data packet transmission. On the other hand, in IEEE 802.11 and LBP the data channel becomes free only at the end of the ACK packet transmission. Further the NCTS/NAK tones do not carry any information and their transmission duration only needs to be as long as it takes for the receiver to sense the presence or absence of a tone. In IEEE 802.11 and LBP, on the other hand, control packets with much information are transmitted for the same purpose and are several 10's of bytes long. Finally, control packets such as ACK are themselves prone to channel error and may result in unneccassary retransmissions. The cumulative effect of these factors contribute to a significant difference in the data throughputs of IEEE 802.11 and LBP when compared with our protocol.

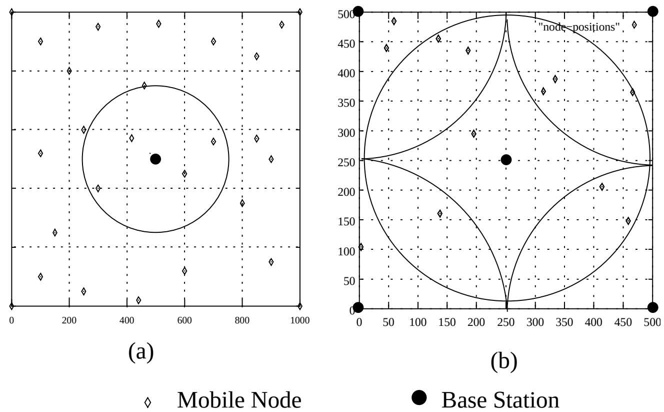

Figure 8. **Scenario for studying (a) throughput (b) reliability of the MAC protocols.**

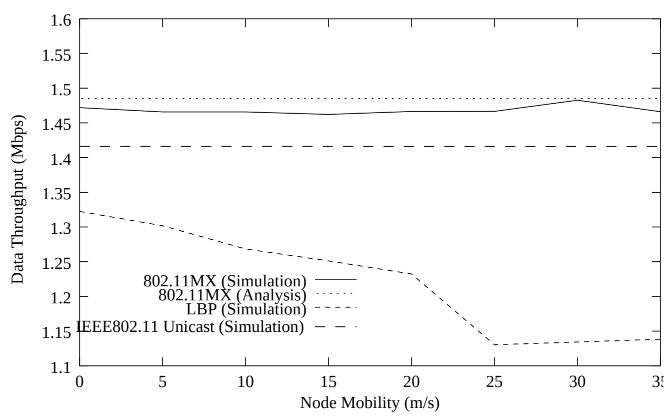

Figure 9. **The throughput of LBP drops with increasing node mobility due to the frequent reselection of leader node.**

2.2. Reliability. Fig. 8(b) shows the system configuration used for testing the reliability of the protocols as a function of node mobilitiy. There are five base-stations with overlapping transmission regions. The base stations themselves cannot communicate directly but it is possible that a node can receive from multiple base stations at the same time. Further, we also introduce random broadcast and unicast communications between some mobile nodes. We study a set of 10 mobile nodes that are always within the range of the central base station. The packets recieved by these nodes are prone to collision with packets from other base stations and with packets from the unicast and broadcast transmissions. Further, packets can also be corrupted due to channel error. Here we have taken Bit Error Rates of 10^{-5} and 10^{-6}.

To guage the relative reliability of the protocols, we had the central base station of fig 8(b) transmit 50,000 multicast data packets. We then measured the average number of data packets that each of the 10 designated nodes should have received but did not. It is possible that for some packets, some nodes received them while others did not. This metric gives us an idea of the efficacy of the local error-recovery capability of the various protocols. Packets that are missed (dropped) by this local error-recovery mechanism will have to be recovered by the higher layers in the protocol stack of the individual nodes.

Fig. 10 shows the percentage of packets dropped by each of the protocols as a function of node mobility. None of the protocols studied provide 100% reliability. However, it is seen from the graph that on an average our protocol drops only half as many packets as LBP. Our protocol drops 0.1% of the packets while LBP drops

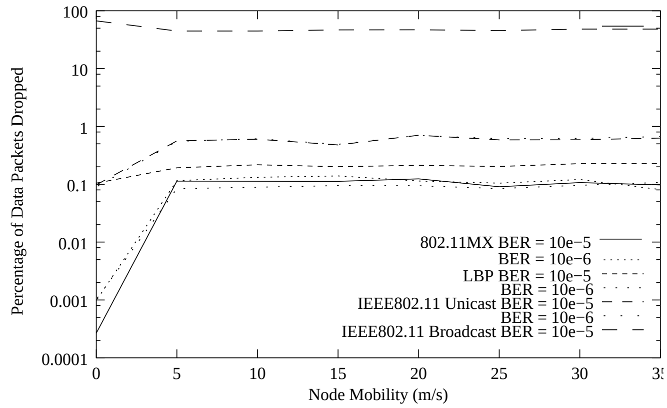

Figure 10. **The reliability of both LBP as well as 802.11MX are not significantly affected by mobility.**

0.2%. The IEEE 802.11 broadcast drops as much as 40% of the data packets under similar traffic conditions. Interestingly, the graph shows that both our protocol as well as LBP multicast provide higher reliability than the IEEE 802.11 unicast. We attribute this to the fact that in a multicast environment there are more number of nodes and there is a greater diversity of channel conditions and node locations, and hence there is a higher probability of atleast one of these several nodes detecting that something is wrong with a transmission and requesting a retransmission. In unicast, if the lone receiver node does not detect a transmission problem for some reason, no local error-recovery is done.

3. Additional Hardware Requirement

The higher data throughput and reliability of 802.11MX comes at the cost of an additional transceiver. One tranceiver is required to handle the dual busy tones while another is required for the data/NCTS/NAK. Further, the tranceiver that handles the data/NAK/NCTS should be capable of handling both modulated data signals as well as narrowband sinusoidal wave signals. Though compared to the data channel the tone channels have negligible bandwidths, they are still an overhead. *It should be noted that 802.11MX can be adapted to work without tones in the absence of addtional tranceiver, however with a degraded performance.*

CHAPTER 6

Conclusions and Future Work

1. Conclusions

We identified the unique characteristics of ad hoc networks and saw that mobility makes providing reliable communication even more difficult. We surveyed a few MAC protocols that have been proposed so far to provide reliability and identified some problems with them. We also saw that the IEEE 802.11 standard does not support reliable multicast.

In this document, we proposed an extension to the IEEE 802.11 MAC to improve link-level reliability for multicast data. A novel feature of this extension is that it uses tones rather than packets to signal a NAK and hence is not prone to NAK packet collisions. Further, dual busy tones have been incorporated to reduce packet collisions due to node mobility. The reduction of these two types of collisions results in a significant increase in data throughput and reliability.

We analyzed our algorithm and compared it against the Leader Based Protocol. We also simulated the two protocols. Our results suggest that our protocol extension performs better than LBP in terms of both actual data throughput as well as reliability. Our protocol drops only half as many data packets as comparted to the Leader Based Protocol. The throughput of neither of the protocols is affected by mobility.

2. Future Work

We intend to extend our basic MAC protocol to work with multiple data channels. In light of the fact that bandwidth is limited for wireless communications, we want to find out what the optimal number of channels would be. We also intend to incorporate power control and study the trade-off between algorithm complexity and energy-efficiency.

We are also currently working on a routing protocol for reliable multicast in mobile ad hoc networks. We intend this to be a lightweight protocol that dynamically adapts to continuously varying network topology.

REFERENCES

[1] ANSI/IEEE Standard 802.11-wireless LAN medium control (MAC) and physical layer (PHY) specifications, In 1999 Edition.

[2] V. Bhargavan, A. Demers, S. Shenker, L. Zhang, "MACAW: A Media Access Protocol for Wireless LAN's", In *Proceedings of the ACM SIGCOMM 1994*, Page(s) 212-225, Aug 1994.

[3] E. Bommaiah, M. Liu, A. McAuley, R. Talpade, "AMRoute:Ad hoc Multicast Routing Protocol", *Internet Draft*, draft-talpade-manet-amroute-00.txt, Aug 1998, Work in progress.

[4] Bluetooth, Available via http://www.bluetooth.com, *[Accessed on August 2002].*

[5] J. Deng, Z. J. Haas, "Dual Busy Tone Multiple Access (DBTMA): A New Medium Access Control for Packet Radio Networks", In IEEE ICUPC'98, Florence, Italy, 1998.

[6] C. L. Fullmer, J. J. Garcia-Luna-Aceves. "Floor Acquisition Multiple Access

(FAMA) for Packet Radio Networks", In *Proceedings of the ACM SIGCOMM 1995*, Page(s) 262-273, Aug 1995.

[7] J.J. Garcia-Luna-Aceves, E.L. Madruga, "The Core Assisted Mesh Protocol", In *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, Aug 1999.

[8] S.K.S. Gupta, P.K. Srimani, "Cored-Based Tree with Forwarding Regions(CBT-FR): A Protocol for Reliable Multicasting in Mobile Ad Hoc Networks", In *Journal of Parallel and Distributed Computing, Special Issue on Routing*, Volume 61, Issue 9, Page(s) 1249-1277, 2001.

[9] P. Karn, "MACA - A New Channel Access Method for Packet Radio", In *Proceedings of the ARRL/CRRL Amatuer Radio Computer Networking Conference*, 1990, page(s) 134-140.

[10] J. Kuri, S. Kasera, "Reliable Multicast in Multi-access Wireless LANs", In *Proceedings of INFOCOM '99*, New York, March 1999.

[11] S.-J. Lee, M. Gerla, and C.C. Chiang, "On-Demand Multicast Routing Protocol", In *Proceedings of IEEE WCNC'99*, New Orleans, Sep. 1999, pp. 1298-1304.

[12] M. Srivastava, "Designing Energy-Efficient Mobile Systems", *MOBICOM'2000 Tutorial*, August 2000.

[13] Network Simulator - ns-2, Available via http://www.isi.edu/nsnam/ns/, *[Accessed on August 2002].*

[14] K. Obraczka, G. Tsudik, "Multicast Routing Issues in Ad Hoc Networks", In *IEEE ICUPC*, October 1998.

[15] T. Ozaki, J.B. Kim, T. Suda, "Bandwidth-Efficient Multicast Routing Protocol for Ad-Hoc Networks", In *Proceedings of IEEE ICCCN'99.*

[16] E.M. Royer, C.E. Perkins, "Multicast Operation of the Ad hoc On-Demand Distance Vector Routing Protocol", In *Proceedings of Mobicom'99*, Seattle, Aug 1999, pp 207-218.

[17] J. Schiller, Mobile Communications, In Addison Wesley (2000).

[18] S. Singh, M. Woo, C. S. Raghavendra, "Power-Aware Routing in Mobile Ad Hoc Networks", In *Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking*, 1998.

[19] S. Singh, C. S. Raghavendra, "PAMAS: Power Aware Multi-Access Protocol with Signalling for Ad Hoc Networks", In *ACM Computer Communication Review*, Vol. 28(3), Page(s) 5-26, 1998.

[20] D. Towsley, J. Kurose, S. Pingali, "A Comparison of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols", In *IEEE Journal on Selected Areas in Communications*, April 1997.

[21] C.W. Wu, Y.C. Tay, C.-K. Toh, "Ad Hoc Multicast Routing Protocol utilizing Increasing id-numberS (AMRIS) Functional Specification", In *Internet-Draft, draft-ietf-manet-amris-spec-00.txt*, Nov. 1998.

[22] S. Wu, Y. Tseng, J. Sheu, "Intelligent medium access for mobile ad hoc networks with busy tones and power control", In *IEEE Journal on Selected Areas in Communications*, Vol. 18(9), Page(s): 1647-1657, September 2000.

[23] S. Wu, Y. Tseng, C. Lin, J. Sheu, "A Multi-channel MAC Protocol with Power Control for Multi-hop Mobile Ad Hoc Networks", In *Computer Journal*, Volume 45, Issue 1, 2002, page(s): 101-110.

[24] S. Wu, Y. Tseng, J. Sheu. "A New Multi-channel MAC Protocol with On-demand Channel Assignment for Mobile Ad Hoc Networks", In *International Symposium on Parallel Architectures, Algorithms and Networks*, December 2000.

[25] M. Yamamoto, J. Kurose, D. Towsley, H. Ikeda, "A Delay Analysis of Sender-Initiated and Receiver-Initiated Reliable Multicast Protocols", In *Proceedings of the IEEE Infocom 97*, Japan, 1997.