A Mechanism to Improve the Performance of IEEE 802.11 MAC Protocol

Priyadarshini Sabut

(Roll No : 209CS1078)

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Orissa, India

A Mechanism to Improve the Performance of IEEE 802.11 MAC Protocol

Thesis submitted in partial fulfillment of the requirements for the degree of

Master of Technology

in

Computer Science and Engineering

(Specialization: Computer Science)

by

Priyadarshini Sabut

(Roll No : 209CS1078)

Supervisor

Dr. Ashok Kumar Turuk

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Orissa, India

May 2011

Dedicated to my teachers and loved ones.

Department of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769 008, Orissa, India.

Certificate

This is to certify that the work in the thesis entitled ***"A Mechanism to Improve the Performance of IEEE 802.11 MAC Protocol"*** submitted by ***Priyadarshini Sabut*** is a record of an original research work carried out by her under my supervision and guidance in partial fulfillment of the requirements for the award of the degree of Master of Technology in Computer Science & Engineering with specialization in Computer Science from the department of Computer Science and Engineering, National Institute of Technology Rourkela. Neither this thesis nor any part of it has been submitted for any degree or academic award elsewhere.

Dr. Ashok Kumar Turuk
Associate Professor
Department of CSE
NIT Rourkela, Orissa

NIT Rourkela
May, 2011

Acknowledgment

"Imagination is more important than knowledge ..."

Albert Einstein

I owe my deepest gratitude to the ones who have contributed greatly in completion of this thesis.

Foremost, I would like to express my sincere gratitude to my supervisor, Dr. Ashok Kumar Turuk who has provided me with continuous encouragement, guidance and support from initial to final level to carry out this research. His dedication and consistent notation in my writings has motivated me to work for excellence.

I am indebted to all the professors, co-researchers, batch mates and friends at National Institute of Technology Rourkela for their active or hidden cooperation. Their contribution can never be penned with words.

I must acknowledge the academic resources that I have got from NIT Rourkela. I would like to thank administrative and technical staff members of the Department for their cooperation.

Most importantly, none of this would have been possible without the love, support, and patience of my family. I would like to express my heartfelt gratitude to them.

Last but not the least, the one above all of us, the omnipresent God, for answering my prayers for giving me the strength to plod on despite my constitution wanting to give up and throw in the towel, thank you so much Dear Lord. Thank you for showing me the path ...

.

Priyadarshini Sabut
Email : *priya07sabut@gmail.com*

Abstract

Ad hoc networks are gaining popularity due to their less cost and ease deployment. Efficiency of these networks depend on the performance and reliability of the medium access control (MAC) protocol applied in such networks. Since the channel is shared by nodes, an efficient MAC should allow the nodes to access channel without degrading the performance.

The performance of IEEE 802.11 gets degraded due to the presence of hidden and exposed terminal. IEEE 802.11 DCF was designed to overcome these problem using a virtual carrier sensing mechanism. Though IEEE 802.11 DCF is able to overcome the hidden and exposed terminal problem, the throughput and channel utilization is lower due to the inability of the hidden and exposed node to either transmit or receive. In this paper we proposed a mechanism that permits the hidden node to transmit and the exposed node to receive. The proposed mechanism also overcome the RTS-induced and CTS-induced problem. We performed extensive simulation using NS-2 simulator. It is observed that the proposed scheme outperforms 802.11 DCF in-terms of throughput and packet delivery ratio with marginally increased in control overhead.

Contents

List of Figures

List of Tables

List of Acronyms

ACK Acknowledgement

BEB Binary Exponential Backoff

BTMA Busy Tone Multiple Access

CSMA Carrier Sense Multiple Access

CSMA/CA Carrier Sense Multiple Access with Collision Avoidance

CTS Clear-to-Send

DCF Distributed Coordination Function

DIFS Distributed Interframe Space time interval

EIFS Extended Interframe Space time interval

FCTS Forward CTS

IEEE Institute of Electrical and Electronics Engineers

MAC Medium Access Control

MACA Multiple Access with Collision Avoidance

MACA-BI MACA By Invitation

MACAW A Media Access Protocol for Wireless LAN's

MILD Multiplicative Increase and Linear Decrease

NAV Network Allocation Vector

QoS Quality of Service

RINC Receiver Initiated NAV Clearing Method

RTR Ready-to-Receive

RTS Request-to-Send

WLAN Wireless Local Area Network

Chapter 1

Introduction

An ad hoc network is a local network with wireless or temporary plug-in connection, in which mobile or portable devices are part of the network only while they are in close proximity. Ad hoc Network does not necessitates the existence of any fixed infrastructure. They dynamically self-organize into an arbitrary network topology, which is temporary in nature [1]. Each node in the network act as a source as well as a router. Due to their non-reliance on fixed infrastructure,these networks are suitable for emergency/rescue operations like natural disaster, relief effort, and military networks.

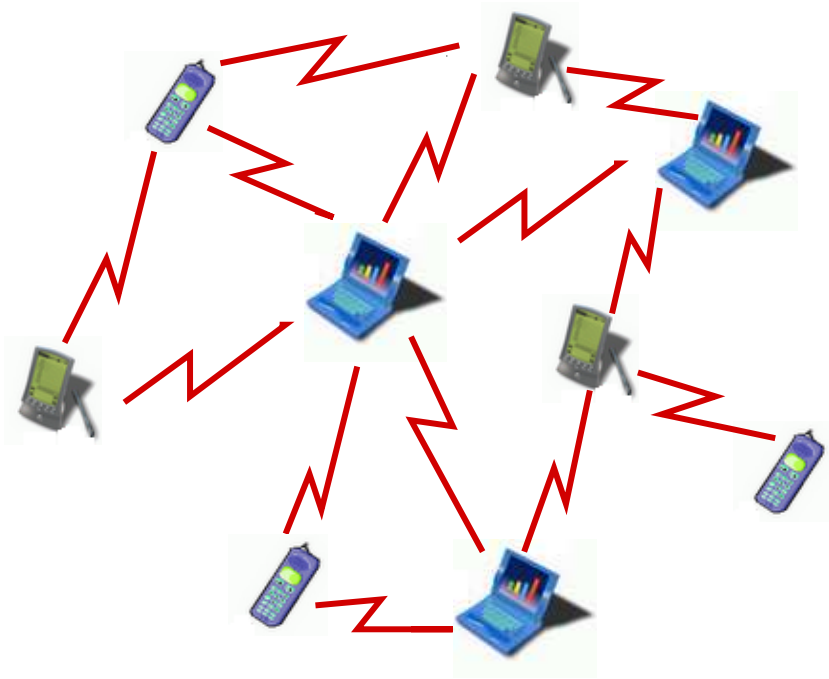

Figure 1.1: An Ad-hoc Wireless Network

The efficiency of ad hoc networks depend on the performance and reliability of Medium Access Control (MAC) protocol applied in such networks [2]. At the end of the 1999 a new high-speed standard for wireless LAN was ratified by the Institute of Electrical and Electronics Engineers (IEEE) 802.11 standards body, the IEEE 802.11b. This standard overtakes the original 1 and 2 Mbs direct sequence physical layer transmission standard to reach the 11 Mbs [3]. Even though the channel bandwidth is significantly increased with the IEEE 802.11b standard, the study of ad hoc network have to still concentrate on the bandwidth consumption. The nodes in an ad hoc wireless network share a common broadcast radio channel. Since the radio spectrum is limited, the wireless bandwidth available for communication is also limited. Keeping a cenralized coordinator in ad hoc network is not possible as the nodes are moving continuously, so the access to this shared medium should be controlled in such a manner that all nodes must receive a fair share of the available bandwidth, so that the available bandwidth must be utilized efficiently [4].

1.1 Issues in Designing A MAC Protocol for Ad hoc Network

The characteristics of the wireless medium are completely different from those wired medium, the protocols of wired medium could not be directly used for wireless networks. The following issues needs to be addressed while designing a MAC protocol for ad hoc wireless network [4].

- ***Bandwidth Efficiency*** **:** In wireless medium the bandwidth available for communication is very limited, so the Medium Access Control (MAC) protocol must utilise the scarce bandwidth in an efficient manner. Bandwidth efficiency can be defined as the ratio of bandwidth used for actual data transmission to the total available bandwidth. The MAC protocol must try to maximize the bandwidth efficiency.

- ***QoS Support*** **:** In ad hoc wireless network, the nodes are mobile most of time. Due to this nature providing Quality of Service (QoS) in such a network is very

difficult. Bandwidth reservation made at one point of time may become invalid once the node moves out of the region where the reservation was made. For time critical traffic sessions such as in military communications, QoS support is essential. Thus, the MAC protocols for ad hoc wireless networks that are to be used in real-time applications must have some kind of resource reservation mechanism.

- ***Synchronization*** **:** The synchronization between nodes in a network is very important for bandwidth reservation by nodes. Exchange of control packets may be required for achieving synchronization, but the control packets should not consume too much of network bandwidth.

- ***Lack of Central Coordination*** **:** Ad hoc wireless networks do not have centralized coordinators, also it is not possible as the nodes keep moving continuously. Therefore, nodes must be scheduled in a distributed manner for gaining access to the channel. This requires exchange of control information. The MAC protocol must make sure that additional overhead, in terms of bandwidth consumption, due to these control information exchange must not very high.

- ***Hidden and Exposed Terminal Problems*** **:** The hidden and exposed terminal problems are unique to wireless networks. The hidden terminal problem refers to the collision of packets at a receiving node due to the simultaneous transmission of those nodes that are not within the direct transmission range of sender, but are within the transmission range of receiver. Collision occurs when both nodes transmit packets at the same time without knowing the transmission of each other. The exposed terminal problem refers to the inability of a node, which is blocked due to transmission by a nearby transmitting node, to transmit to another node. These two problems significantly reduce the throughput of a network when the traffic is high. Thus, the MAC protocol must be free from the hidden and exposed terminal problems.

- ***Mobility of Nodes*** **:** This is the most important factor affecting the

performance of the protocol. As the nodes in ad hoc wireless networks are mobile most of the time, the bandwidth reservations made or the control information exchanged may end up being of no use. The MAC protocol has no role to play in influencing the mobility of nodes, the protocol design must take this mobility factor into consideration so that the performance of the system is not significantly affected by the node mobility.

An efficient MAC protocol through with mobile stations can share the broadcast channel is essential in ad-hoc network as the channel is a scarce resource.

1.2 Summary

In this chapter, a brief introduction of ad hoc network and its applications were discussed. The major issues involved in the designing of a MAC protocol for ad-hoc network were also identified.

1.3 Thesis Organization

In this thesis, a new protocol is proposed that attempts to improve the performance of IEEE 802.11 MAC, simultaneously maintaining the level of fairness between the nodes that IEEE 802.11 has. In chapter 2, some of the related works on MAC protocol for ad hoc network are discussed. Chapter 3 discusses an overview of IEEE 802.11 standard and the working of IEEE 802.11 DCF. Several problems that are common to ad hoc network are also identified. The detailed design of our proposed scheme is discussed and analysed in chapter 4. In Chapter 5, simulations are done to evaluate the performance of the proposed scheme and the results are discussed. Chapter 6 concludes this thesis with a brief summary and discusses possible directions for further development of this work.

Chapter 2

Literature Review

Several New MAC protocols are constantly being developed to optimize the performance of the network. Given that the protocols are continually being proposed different metrics are also developed to evaluate the efficiency and robustness of these protocols. One common metrics is channel utilization which is defined as the fraction of time spent for successful transmissions. Another metrics is saturation throughput which represents the maximum achievable throughput under stable conditions. Both the channel utilization and throughput are directly affected by the amount of control overhead for each successful transmission and the efficiency with which collision is resolved. Protocol fairness is also an important issue that allows all the nodes an equal chance of accessing the channel.

2.1 Related Work

This section presents some of the researches that is done in the field of Medium Access Control (MAC) in ad hoc networks.

The earliest MAC protocol proposed for wireless networks is ALOHA, so called random-access mode. In pure ALOHA [5], the nodes transmit whenever they have data to send, without knowing the current state of the medium. A positive Acknowledgement (ACK) frame is used to determine a successful transmission. If no ACK is received by the sender node, a collision is assumed to have occurred and

the node must retransmit after a random delay. Since no carrier sense is one, a data packet is vulnerable to collision, if some other node transmit at that time. A throughput analysis [6] shows that, pure ALOHA utilises 18% of channel bandwidth. The extension to unsynchronised ALOHA is slotted ALOHA, in which the time is divided into number of slots, whose duration is exactly equal to the transmission time of a single packet (assuming constant-length packets). The nodes are allowed to transmit only at the begining of a slot. Here collision can occur only if two or more nodes transmit at the beginning of same slot, they overlap completely rather than partially increasing channel efficiency. By this simple change [5], the maximum throughput in slotted ALOHA increases by a factor of two to 36%.

To further improve the channel utilization, in [5] Kleinrock and Tobagi suggests a third approach for using the channel; namely, the Carrier Sense Multiple Access (CSMA) mode. In this scheme the nodes attempt to avoid collisions by listening to the carrier transmitted by other nodes. Based on the information about the state of the channel, CSMA protocol is divided into non-persistent CSMA and p-persistent CSMA. In *non-persistent CSMA*, the node transmits if the channel is sensed idle. If the channel is sensed busy, the node schedules the retransmission to some later time according to the retransmission delay distribution. In *p-persistent CSMA*, the node sense the channel continuously, until the channel is idle. It then transmits in that given slot with probability p and defers its transmission to next slot with probability $1-p$. If collision occurs, the node wait for a random delay before retransmitting. The special case of p-persistent CSMA is *1-persistent CSMA*, that transmits the packet as soon as it senses a idle channel with probability $p = 1$. This paper also analysed the performance of both ALOHA and CSMA and shows that CSMA protocols outperforms the performance of ALOHA not only in terms of higher throughput, they also performs better under high network loads with a comparable lesser normalised delay. Among other CSMAprotocols *p-persistent CSMA* provides best performance. Considering the benifits of carrier sensing most of the succeeding MAC protocols are based on CSMA mechanism. Throughout this paper they assumed that all are within range and in line-of-sight of each other. Each terminal, however, may not be able to

hear all the other terminal's traffic. This gives rise to "hidden-terminals" problem. The maximum throughput of CSMA with no hidden terminals is 83% approximately.

In [7], Tobagi and Kleinrock shows that the existence of hidden terminals significantly degrades the performance of CSMA. To eliminate problem, they introduce theBusy Tone Multiple Access (BTMA) protocol as an extension of CSMA. This protocol splits the available bandwidth into two separate channels: a busy-tone (control) channel and a message (data) channel. When a node is ready for transmission, it senses the busy tone channel. If the busy-tone is absent, it transmits a busy-tone signal on the busy-tone channel and starts data transmission; otherwise, it reschedules the packet for transmission at some later time. Other nodes sensing carrier on data channel also transmits busy-tone signal on their busy-tone channel. Though this mechanism minimizes the probability of collisions, bandwidth utilization is very poor. The limitations of BTMA are the use a separate channel to convey the state of the data channel.

As CSMA protocol senses the state of the channel only at the transmitter, this protocol does not overcome the hidden terminal problem, when the transmitter and receiver are not in the range of each other. Also the bandwidth utilization is less because of exposed terminal problem.

In [8], Karn proposed a new scheme, Multiple Access with Collision Avoidance (MACA) to overcome the shortcomings of CSMA. The MACA protocol was inspired by the CSMA/CA method [9]. MACA uses two signalling packets: Request-to-Send (RTS) and Clear-to-Send (CTS). Here, the RTS/CTS exchange between source and destination precedes the data transmission. In MACA, any station hearing an RTS defer long enough so that the transmitting station can receive the returning CTS and any station hearing the CTS defer long enough to avoid colliding with the returning data transmission. Thus MACA overcomes the hidden and exposed terminal problem. If a node does not receive a CTS packet in response to RTS send, the node uses the Binary Exponential Backoff (BEB) algorithm to backoff for some time before retrying. According to [10] the Binary Exponential Backoff (BEB) algorithm used in MACA, does not allocate the bandwidth in fair manner, as it adjusts the back-off counter

value very rapidly. Also, MACA does not used any Acknowledgement (ACK) packet for acknowledging the reception of data packet. In this protocol, nodes transmitting a data packet assume the successful reception of that data packet. When data packets suffer a collision, or are corrupted by noise, the error has to be recovered by the transport layer, that increases the end-to-end delay experienced by the application.

Bharghavan et al. [10] proposed A Media Access Protocol for Wireless LAN's (MACAW), as a modification to MACA. To prevent large variations in back-off values, MACAW uses Multiplicative Increase and Linear Decrease (MILD) back-off mechanism. For each collision, the backoff interval is increased by a multiplicative factor (1.5), and for each successful transmission the backoff is decremented by one. The multiplicative increase of backoff value reduces the probability of collision in a highly congested network and the linear decrements reduces the large variations in backoff value as in BEB. To reduce the recovery delay in MACA, this responsibility in MACAW is given to the link layer by using a Acknowledgement (ACK) packet along with RTS-CTS-DATA exchange. Other features, such as coping of backoff value to propagate congestion information, use of data-sending (DS) packet to advertise the use of the shared channel and use of request-for-request-to-send (RRTS) packet to synchronize with the sender of RTS, are incorporated in MACAW further to improve performance.

In [11], Talucci et al. introduced MACA By Invitation (MACA-BI), as a simplified version of Multiple Access with Collision Avoidance (MACA). In MACA-BI, the RTS part of the RTS/CTS handshake is suppressed, leaving only the Ready-to-Receive (RTR) control message which can be viewed as an "invitation" by the receiver to transmit. If the sender node is ready to transmit, it responds by sending a DATA packet. The efficiency of this protocol depends on the ability of the receiver node to predict arrival rate of traffic at sender node accurately. For this prediction, the DATA packets in MACA-BI are modified to carry control information regarding backlogged flows at the transmitter node, number of packets queued and packet lengths.

Hass et al. [12] proposed a new MAC protocol, termed as the dual busy tone multiple access (DBTMA) scheme, as an extension to the BTMA scheme. This

protocol uses two busy-tones on control channel, BT_t and BT_r. This paper shows DBTMA protocol is superior to other schemes that rely on the RTS/CTS dialogue on a single channel or to those that rely on a single busy tone.

The IEEE 802.11 DCF protocol [13] as described in 3.1.1, is similar to MACA, except that it includes link level to enable faster collision detection. This reduces the end-to-end latency experienced by application.

In [14], Shukla et al. describe an enhancements to the IEEE 802.11 DCF MAC Protocol which enable nodes to identify themselves as exposed nodes and to opportunistically schedule concurrent transmissions whenever possible thereby improving utilization and mitigating the exposed node problem.

FAMA [15] combines carrier sensing along with the RTS-CTS control packet exchange. It uses long dominating CTS packets to act as a receive busy tone to prevent any competing transmitters in the receiver's range from transmitting. This requires each node hearing the interference to keep quiet for a period of one maximum data packet to guarantee no collision with the ongoing data transmission. In [16], the simulation results shows FAMA performs better than MACAW in presence of hidden terminal problem. This scheme is not efficient when the RTS/CTS negotiation process fails or the DATA packet is very short.

Ghaboosi et al, in [17] discussed the scenarios where a desired destination is located in the range of other transmitters, resulting destination unreacheable problem which results in throughput and channel utilization degradation. By making slight changes to MAC layer, they proposed a new MAC protocol, to address such problems. This scheme provides better performance than IEEE 802.1 1 and DBTMA, with a little cost of control overhead. The unreachability problem is again addressed in [2] and as a solution to this problem they proposed a novel medium access control (MAC) protocol, called, *eMAC*, where nodes maintain Double Hop Neighborhood (DHN) graphs while exchanging eMAC tables to share their knowledge about their neighbourhood topology and an adaptive table broadcasting technique to facilitate topology information.

In [18], Han et al. proposed an improvement to MACA, named as MACA-RPOLL, that is inspired by point coordination function (PCF). When collisions occur at a node,

without using any backoff mechanism, it polls all of the one-hop neighbor nodes in its polling list to enquire whether it has data to transmit. This scheme ensure no collision at sender, only the polling packet from receiver may collide with other types of packets from some other node.

Du et al. [19], identifies two new problems : RTS/CTS-induced problem that arise due to virtual carrier sense mechanism, via RTS/CTS handshaking used in IEEE 802.11 DCF, to reserve wireless channel. All the neighbor nodes that overhear either the RTS or CTS, update their Network Allocation Vector (NAV) for that duration of time in which the channel will remain busy and these nodes defer their transmission for that duration. The problems arise when RTS or CTS is not successfully received by the addressed node, which causes the channel waste resulted from unnecessary NAV setting. To overcome the RTS-induced problem, the nodes overhearing RTS set their NAV to shortNAV (SIFS time + CTS Time) and sense the channel afterwards for data packet transmission. If any carrier is received within shortNAV period the NAV is set, otherwise the nodes could reattempt channel access. To solve CTS-induced problem, they proposes new scheme Receiver Initiated NAV Clearing Method (RINC), which uses a receiver initiated packet CLR, when the channel at receiver side is idle for a predefined time period, denoted as T_{thr}_CTS, after replying CTS, to clear the NAV set at all the neighboring node. Thus, the channel accessibility is recovered and the resource utilization is expected to be enhanced.

2.2 Summary

This chapter highlights on major MAC protocols that exist for ad hoc wireless networks. In this chapter, the protocols are briefly discussed considering their performance measures and limitations.

Chapter 3

IEEE 802.11

The IEEE 802.11 is standard is a predominant standard for wireless LAN. To date the Institute of Electrical and Electronics Engineers(IEEE) have developed three specifications for wireless LAN Wireless Local Area Network (WLAN) 802.11 family: 802.11-1997 (802.11 legacy), 802.11a, 802.11b, 802.11g, 802.11-2007 and 802.11n. All these specifications use Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA), as path sharing problem. If a source station has data packet to send, it checks the system to see, if the medium is busy. If the medium is not busy, the data packet is sent; if the medium is busy, the station waits until the medium is free.

3.1 AN OVERVIEW OF THE IEEE 802.11 STANDARD

The IEEE 802.11 [13] is the standard for medium access control and physical layers in wireless local area networks (WLANs). The standard currently defines a single MAC which interacts with three PHYs (all of them running at 1 and 2 Mbps) as follows: frequency hopping spread spectrum in the 2.4 GHz band, direct sequence spread spectrum in the 2.4 GHz band, and infra-red.

The standard specifies two medium access control mechanisms, DCF (Distributed Coordination Function), and PCF (Point Coordination Function). This section limiting the discussion only on DCF scheme in detail (since in PCF, mode polling

occurs with a point coordinator determining which station has the right to transmit, so controlled by a central base station and needs a fixed infrastructure and cannot be used in ad hoc networks).

3.1.1 IEEE 802.11 DCF

The Distributed Coordination Function (DCF) is a fundamental mechanism to access the medium in IEEE 802.11 protocol [20]. The Distributed Coordination Function (DCF) in the IEEE 802.11 MAC layer protocol is a random access scheme that primarily employs a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) protocol which works on a "listen before talk" scheme. The standard [13, 20] defines two access methods :

- The two-way handshaking method
- The four-way handshaking method

The ***two-way handshaking method***, basic access method, in IEEE 802.11 MAC protocol is based on a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) MAC protocol that requires every station to perform a Carrier Sensing activity to determine the current state of the channel (idle or busy). If the medium is found to be busy, the station defers its transmission. Whenever the channel becomes idle for at least a Distributed Interframe Space time interval (DIFS), the station (re)starts its Basic Access mechanism (Figure 3.1). This method involves only the exchange of DATA packet and ACK packet. This mechanism [20] is characterized by the immediate transmission of a ACK packet by the destination station, upon successful reception of a packet transmitted by the sender station. Explicit transmission of ACK packet is required because, in wireless medium, a transmitter can not determine a successful reception at destination by listening to its own transmission.

The ***four-way handshaking method***, extends the basic method with the exchange of Request-to-Send (RTS) control packet and Clear-to-Send (CTS) control packet prior to the exchange of DATA and ACK packet. This is also known

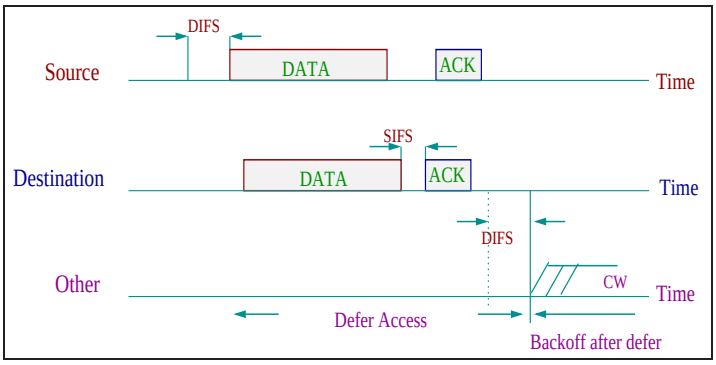

Figure 3.1: Basic Access Mechanism

as request-to-send/clear-to-send (RTS/CTS) access method. Before transmitting a packet, a station operating in RTS/CTS mode "reserves" the channel by sending a special Request-to-Send (RTS) short frame. The destination station acknowledges the receipt of an RTS frame by sending back a Clear-to-Send (CTS) frame, after which normal DATA packet transmission and ACK response occurs. Since collision may occur only on the RTS frame, and it is detected by the lack of CTS response, the RTS/CTS mechanism allows to increase the system performance by reducing the duration of a collision, especially when the DATA packets are large. It also solves the hidden terminal problem experienced in wireless networks. Here we present the main feature of IEEE 802.11 DCF, with respect to RTS/CTS access method. For further details, please refer to [13].

The IEEE 802.11 DCF uses the Binary Exponential Backoff (BEB) algorithm to resolve channel contention. A station with a new packet to transmit must first carrier sense the medium. If the channel is idle for a period of time equal to a distributed interframe space (DIFS), the station transmits. Otherwise, if the channel is sensed busy (either immediately or during the DIFS), the station persists to monitor the channel until it is measured idle for a DIFS. At this point, the station generates a random backoff interval before transmitting (this is the Collision Avoidance feature of the protocol), to minimize the probability of collision with packets being transmitted by other stations. In addition, to avoid channel capture, a station must wait a random backoff time between two consecutive new packet transmissions, even if the medium is sensed idle in the DIFS time. The random backoff value is uniformly chosen from the

interval $(0, CW - 1)$, where CW is the size of node's contention window and depends on the number of transmissions failed for the packet.

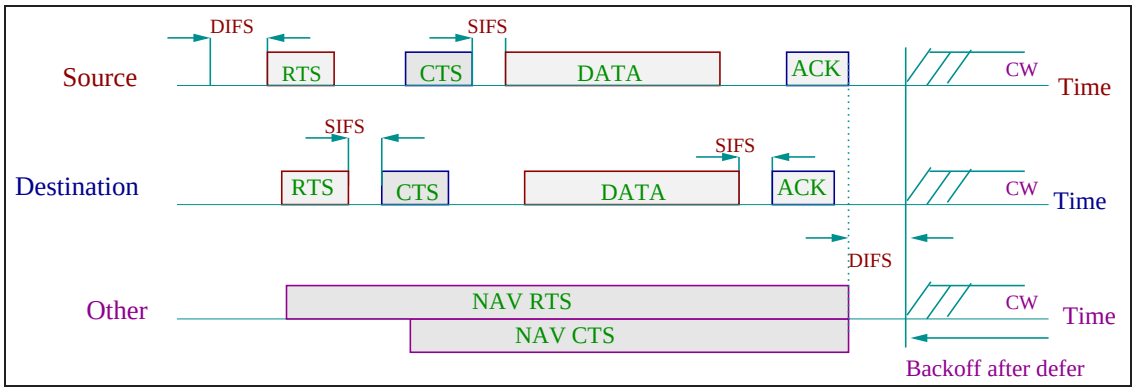

Figure 3.2: RTS/CTS Access Mechanism

The backoff counter value represents the number of idle slots a node needs to waits before it can transmit. The node decrements its backoff counter by one for each idle slot it sense on the channel. The node freezes the backoff counter, when a transmission is detected on the channel and reactivated when the channel is sensed idle again for more than a DIFS.

When the backoff counter reaches zero, the node transmits an Request-to-Send (RTS) packet. If a node receives an RTS packet, it responds, after a SIFS, with a Clear-to-Send (CTS) packet. The transmitting node is allowed to transmit its DATA packet only if the received CTS packet is correct, which is then replied with a ACK packet.

The RTS and CTS packet contains the total duration of packet transmission. All the neighbor nodes that overhear either the RTS or CTS, update their Network Allocation Vector (NAV) for that duration of time in which the channel will remain busy and these nodes defer their transmission for that duration when the NAV is set (Figure 3.2). This allows a collision-free transmission for the DATA and ACK packet. This mechanism of defering transmission based on the NAV is known as *virtual carrier sensing* and it effectively reserves the channel for the current dialog.

The RTS/CTS mechanism is very effective in terms of system performance, especially when large packets are considered [20], as it reduces the length of the control packets involved in the contention process.

The combination of virtual and physical carrier sensing ensures that, collision may occur only when two (or more) nodes transmit within the same slot time. Since a node can not sense the medium while transmitting, so the absence of a reply is taken be an indication that collision has occurred. When this situation occur the node doubles its CW to a maximum value and backoff counter is updated. The nodes that sense this collision update their NAV to Extended Interframe Space time interval (EIFS). As the number of collision increases, the Binary Exponential Backoff (BEB) algorithm at each node increases the backoff exponentially to reduce the probability of further collision. At the first transmission attempt, CW is set equal to a value CW_{min} called minimum contention window. After each unsuccessful transmission, CW is doubled up to a maximum value of CW_{max}. The values of CW_{min} and CW_{max} is dependent on physical layer used. Once the node successfully transmits a DATA packet, it resets the CW value back to minimum value.

According to IEEE 802.11 standard, upon completing a successful transmission, the nodes wait for a random backoff time between two packet transmission. This is to prevent the capturing of a channel by a single node, and to provide a fair share of the channel/available bandwidth to all the nodes in the network.

3.2 Problem Identification

Here we are discussing, the two major issues of ad hoc network : hidden and exposed terminal problem and also RTS/CTS-induced problem identified in [19] that significantly affects the performance of this network.

3.2.1 Hidden Terminal Problem

A hidden node is one which is outside the transmission range of the sender, but within the range of receiver. Due to the limited transmission range of mobile nodes, multiple transmitters within the range of same receiver may not know one another's transmission, in effect hidden from each other. When these transmitters transmit at around the same time, they do not realize that their transmissions collide at receiver.

Assume that there are four nodes A, B, C, and D as shown in Figure 3.3. The dotted circle denote their communication ranges. Let us assume that node A is communicating with node B. Suppose node C wants to transmit to node D. Node C senses the channel as free and start transmitting, resulting collision at B. This problem is called hidden terminal problem, which degrades the throughput significantly.

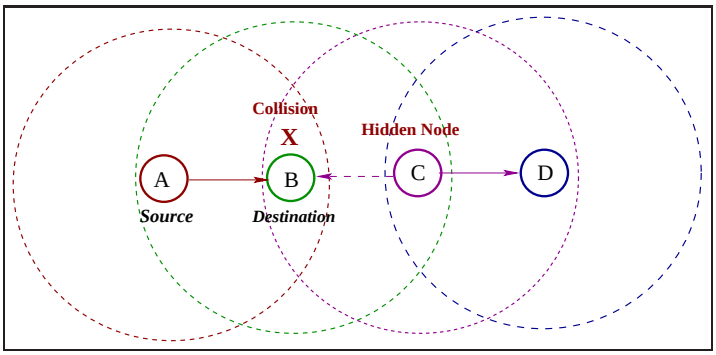

Figure 3.3: Hidden Terminal Problem

3.2.2 Exposed Terminal Problem

An exposed node is one that is within the range of sender but out of the range of receiver. These nodes cause underutilization of bandwidth. Assume that there are four nodes A, B, C, and D as shown in Figure 3.4. The dotted circle denote their communication ranges. Let us assume that node C is communicating to node D. Suppose node B wants to transmit to node A. Node B senses the channel to be busy and could not transmit to A. Although this transmission would not cause a collision at D, but B is prevented from transmitting. The node B is an exposed node. This results an inefficient bandwidth utilization at node B. This problem is called exposed terminal problem.

Hidden and exposed terminal problems can occur frequently in ad hoc network causing a significant degradation in the network throughput. Overcoming the hidden and exposed node problem has become one of the important aspects of MAC protocol design.

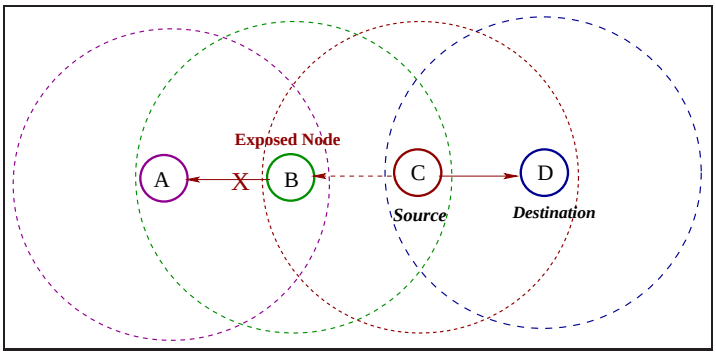

Figure 3.4: Exposed Terminal Problem

3.2.3 RTS -induced and CTS -induced Problem

To overcome the hidden and exposed terminal problem, IEEE 802.11 DCF, uses a mechanism called Network Allocation Vector (NAV) [13, 10, 20]. Nodes overhearing either RTS or CTS set their NAV respectively, and defer their channel access for the expected time to finish the packet transmission. Problems arise when the RTS or CTS packet packet is not correctly received at receiver or sender node respectively, which causes underutilization of channel bandwidth due to NAV setting. These are termed as RTS-induced and CTS-induced problem [19]. We illustrates these two problems in the next paragraph.

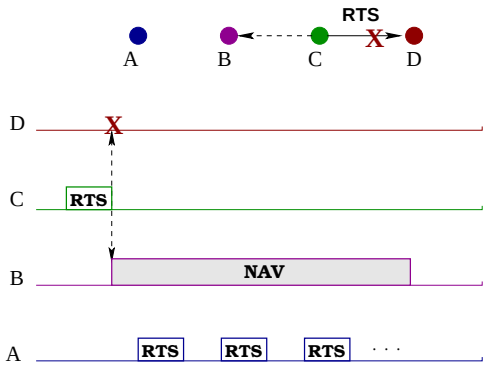

Figure 3.5: RTS-induced Problem

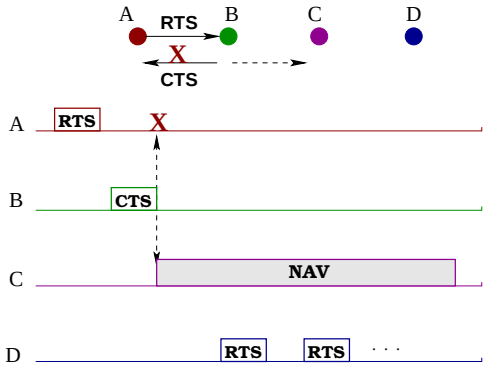

Figure 3.6: CTS-induced Problem

The RTS-induced problem occurs when the RTS packet is not correctly received at the receiver node. Assume that there are four nodes A, B, C, and D as shown in Figure 3.5. Node C initiates its transmission by sending an RTS packet to node D. Upon hearing RTS from node C, node B sets its NAV to the expected time required to finish the transmission. If the reception of RTS fails at D, the transmission from

node B is unnecessarily deferred for a period as set in its NAV. The RTS-induced problem is depicted in Figure 3.5.

Similarly, CTS-induced problem occurs when the CTS packet is not correctly received at the sender node. Assume that there are four nodes A, B, C, and D as shown in Figure 3.6. Node A initiates its transmission by sending an RTS packet to node B. The node B sends an CTS to node A, as a response to the RTS packet. Upon hearing the CTS packet from node B, node C sets its NAV to the expected time required to finish the transmission. If the reception of CTS fails at node A, transmission from node C is unnecessarily deferred for a period equal to the setting in NAV.

3.3 Motivation

Several solutions were proposed for hidden and exposed terminal problem, as discussed in section 2.1, to improve the performance and fairness of 802.11 MAC. By default 802.11 MAC uses CSMA/CA, which prevents the possibility of parallel communication by two neighboring nodes that are either at sender or at receiver. Because of the use of 4-way handshaking, the neighboring node that receives a RTS or CTS packet set their NAV, and are forced to defer their transmission for the whole duration of ongoing transmission. The hidden nodes as discussed in section 3.2.1 could have received data safely, from the nodes those are placed at a distance of two-hop from the receiver node, but at a distance one-hop from the hidden node, without causing any collision at the receiver. The exposed nodes as discussed in section 3.2.2 could have transmitted the data safely because it would not have collided with ongoing transmission, as the collision takes place at the receiver not at the sender. In IEEE 802.11 DCF scheme, all the nodes overhearing either RTS or CTS set their Network Allocation Vector (NAV) respectively, and defer the channel access for the expected time to finish the packet transmission. The problems arise when RTS or CTS is not successfully received by the addressed node, which causes the channel waste resulted from unnecessary NAV setting. This is identified as RTS-induced and CTS-induced problem [19].

An efficient MAC protocol is essential that would provide the level of fairness

by overcoming all these problems without compromising the network performance. With this motivation for developing a new protocol, the major goals of this thesis is identified.

3.4 Proposed Solution

Based on the motivation outlined in section 3.3, the proposed scheme addresses the problem of hidden and exposed terminals and also RTS-induced and CTS-induced problem. This work allows concurrent transmissions by utilizing the information heard from the neighboring nodes during the exchange of control packets in the presence of hidden and exposed terminals. Nodes in the proposed scheme maintain the status of transmitter and receiver of itself and of its neighboring nodes. In the proposed scheme, a hidden node can receive and an exposed node can transmit without causing collision with the ongoing transmission. It achieves successful overlapping transmissions by using an extra packet. The scheme also overcomes the RTS-induced and CTS-induced problems discussed in chapter 4.

3.5 Summary

In this chapter, the IEEE 802.11 MAC is discussed. The access methods defined by IEEE 802.11 DCF were discussed briefly. This chapter addresses the problem of hidden and exposed terminals and the RTS-induced and CTS-induced problems as discussed in literature section 2.1. This chapter also discusses the motivation of this thesis work and gives a brief idea of solution for overcoming these problems.

Chapter 4

Proposed Solution Design

4.1 Assumption

At any point of time, the status of the transmitter and receiver of a node can be in one of the following state:

- *Free* :- Transmitter set to *Free* indicates, the node is not transmitting. Receiver set to free indicates, the node is not receiving.
- *Busy* :- Transmitter set to *Busy* indicates, the node is transmitting. Receiver set to busy indicates, the node is receiving.
- *Unknown* :- Indicates that status of the transmitter and receiver of a node is not known.

Initially, each node set the status of its own transmitter and receiver as *Free* and neighbor nodes transmitter and receiver as *Unknown*. A node will set the status of its own transmitter and receiver as either *Free* or *Busy* and will never be set to *Unknown*. Each node in the network maintains the status of its own and neighboring nodes transmitter and receiver.

4.2 The Proposed Mechanism

In this subsection, we describe the working of the proposed mechanism. Let a source S wants to transmit data to destination D. Since S and D are in neighborhood of each other, they maintain the status of each others transmitter and receiver. The source first checks the status of its own transmitter and destination node's receiver. If both are *Free* for the expected duration of transmission, then it directly transmits data to its destination. If destination node's receiver is busy, it runs an exponential algorithm. For a free transmitter and unknown receiver the following action is performed:

Source Transmitter Status	**Destination Receiver status**	**Action Performed at Source**
Busy	Busy	Defer Transmission for a Duration = Busy Period
Busy	Free	Defer Transmission for a Duration = Busy Period
Free	Busy	Defer Transmission for a Duration = Busy Period
Free	Free	Transmit Data Directly
Free	Unknown	Send RTS

Table 4.1: Action Performed At The Source

Table -4.2, shows the action performed by the source node depending on the status of its transmitter and destination node's receiver as maintained by it.

Initially, a node will set the status of its neighbor transmitter and receiver to *Unknown*. When a node wants to transmit and the receiver status is *Unknown*, it performs the following steps. Let S be the source and D be the destination.

1. The source S sends a RTS packet to node D. The RTS packet contains the duration of data transmission. The neighbor node of S that overhears the RTS packet, starts a timer and wait to overhear data from node S.

2. The destination node D on receiving the RTS packet, performs the following actions:

   (a) Checks the status of its own receiver and transmitter. If either or both are

Busy, then node D does nothing.

(b) If both the receiver and transmitter of destination node is *Free*, then does the following actions:

i. Set the status of source node's transmitter and receiver to *Busy* for duration of data transmission,

ii. Set its own transmitter and receiver to *Busy* for duration of data transmission, and

iii. Transmits a CTS packet in response to RTS packet.

3. Following changes are made by the nodes other than the source, on receiving CTS.

(a) Set the status of their own transmitter to busy for the duration of data transmission,

(b) Set the status of destination node's transmitter and receiver to *Busy* for the duration of data transmission, and

(c) Transmit a Forward CTS (FCTS) packet.

4. Nodes including the destination make the following changes on receiving the FCTS packet.

(a) Set the status of the transmitter of the source of the FCTS packet to *Busy*, and

(b) Set the status of the receiver of source of the FCTS packet to *Free* for duration of data transmission.

5. The source S on receiving the CTS packet, schedules the data transmission.

6. The neighboring nodes of source S, on receiving the DATA packet(before the timer expires), does following changes:

(a) Set the status of their own receiver to *Busy* for duration of data transmission and also

(b) Set the status of the transmitter and receiver of source node S to *Busy* for duration of data transmission.

7. The node D, after receiving the data packet and sends an ACK packet to acknowledge the reception of the DATA packet.

4.3 Notations Used

T^N : denotes the status of transmitter of node N.

R^N : denotes the status of receiver of node N.

$T^{N,M}$: denotes the status of transmitter of node N as seen by node M.

$R^{N,M}$: denotes the status of receiver of the node N as seen by node M.

4.4 Proposed Algorithm

Node S is the source and node D is the destination.

N denotes any node in the network.

if Node S has an DATA packet to send **then**
  The Node checks T^S and $R^{D,S}$ status from the Status Table.
  if Both are free **then**
    the Node will the calculate $Free\_Duration = Remaining\_Busy\_Time - Current\_Time$. If the duration of data transmission is less than or equal to Free_Duration, then source node S sends data directly.
  else if Both or either is busy **then**
    The node will defer for busy period.
  else if T^S status is free and $R^{D,S}$ status is unknown **then**
    Sends a RTS packet destined to node D.
  end if
end if
if Node has received an RTS packet **then**
  Check whether the node is the destination node or not.
  if destination **then**

14: Check whether T^D and R^D are free or not.
15: **if** Both are free **then**
16: Set T^D, R^D, $T^{S,D}$, and $R^{S,D}$ to busy for the duration of data transmission.
17: Send a CTS packet in response to RTS packet destined to node *S*.
18: **else if** either or both are busy **then**
19: do nothing.
20: **end if**
21: **else if** not destination (**Exposed Node N**) **then**
22: Set $T^{S,N}$, $R^{S,N}$ and its own R^N to busy for the duration of data transmission.
23: Set a timer $current\_time + \Delta$. goto step-40 {To check whether the RTS-CTS exchange was successful}
24: **end if**
25: **end if**
26: **if** Node has received a CTS packet **then**
27: Check whether the node is the destination for CTS packet or not.
28: **if** destination *i.e the source node* **then**
29: Set T^S, R^S, $T^{D,S}$, $R^{D,S}$ to busy for duration of data transmission.
30: Wait FCTS packet transmission time. goto step-40.
31: **else if** not the destination (**Hidden Node N**) **then**
32: Set the $T^{D,N}$, $R^{D,N}$ and its own T^N to busy for duration of data transmission.
33: Broadcasts a FCTS packet indicating that the *Receiver* is free.
34: **end if**
35: **end if**
36: **if** Node N has received a FCTS packet from node M **then**
37: Set $T^{M,N}$ to busy for duration of data transmission.
38: Set $R^{M,N}$ to free for duration of data transmission.
39: **end if**
40: **if** Timer expired at node N **then**
41: Check for previously received packet.
42: **if** Previously received packet is RTS **then**
43: Set $T^{S,N}$, $R^{S,N}$ and own T^N to be free {Assuming RTS-CTS exchange was unsuccessful}

44: **end if**
45: **if** Previously received packet is CTS **then**
46: Transmit DATA packet.
47: **end if**
48: **end if**
49: **if** Node has received DATA packet **then**
50: Check if the node is destination or not.
51: **if** destination **then**
52: Send an ACK packet for acknowledging the reception of DATA packet.
53: **else if** not destination **then**
54: Check for timer.
55: **if** Timer is busy **then**
56: stop the timer.
57: **end if**
58: **end if**
59: **end if**

4.5 Example

We illustrate the working of our proposed scheme by means of an example. Consider five nodes *A*, *B*, *C*, *D*, and *E*. Initially, the status of neighboring node's transmitter and receiver are set to *Unknown* and its own node transmitter and receiver are *Free* by a node. The Figure 4.1 shows a node's own transmitter and receiver status as well as the neighboring node's transmitter and receiver status as seen by it.

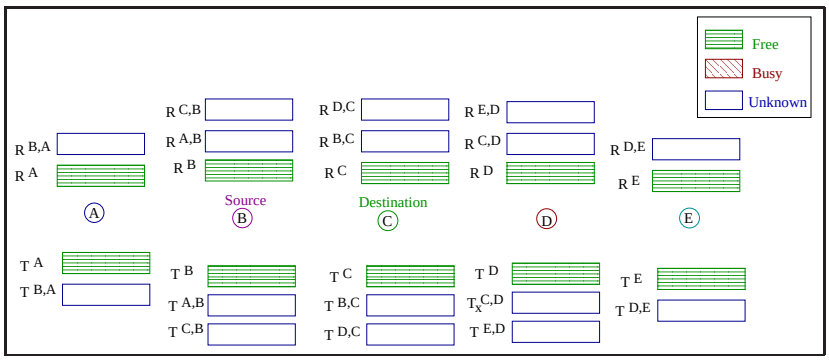

Figure 4.1: Initial setting of transmitter and receiver status by a node

Suppose node B is the source and node C is the destination as shown in Figure 4.1. Node B checks the status of its transmitter which is set to *Free* and node C's receiver as maintained by it, which is set to *Unknown*. Node B transmits an RTS packet to node C. This is depicted in Figure 4.2.

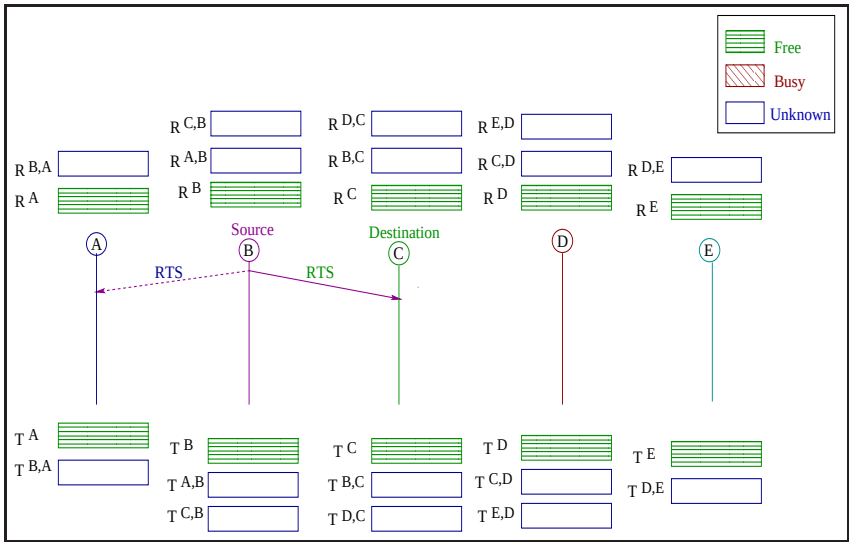

Figure 4.2: RTS from node B to node C.

Neighbors of source node B, in our example node A(not the destination), on receiving the RTS packet, set the transmitter and receiver of node B to *Busy*. Also set its own receiver to *Busy* for duration of data transmission and run a timer for the duration of Δt and starts listening for the data packet from node B During the duration of Δt, the source is expected to receive CTS packet from the destination C and start transmitting data packet. This is depicted in Figure 4.3. If node A does not receive data from node B before the timer expires, then it set the transmitter and receiver of node B as *Unknown*.

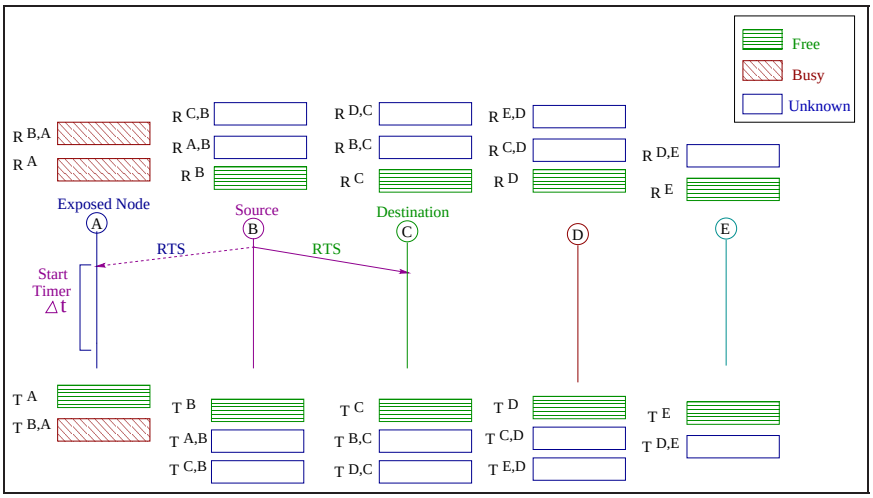

Figure 4.3: Node A start a timer for the duration of Δt.

Destination Node C, on receiving the RTS packet checks the status of its own receiver and transmitter. If either or any one of the transmitter and receiver is *Busy*, then it discards the RTS packet. If both the receiver and transmitter are *Free*, then it sets the transmitter and receiver of node B and its own transmitter and receiver to *Busy* for the duration of data transmission. Reply a CTS packet in response to RTS packet from source node B. This is depicted in Figure 4.4.

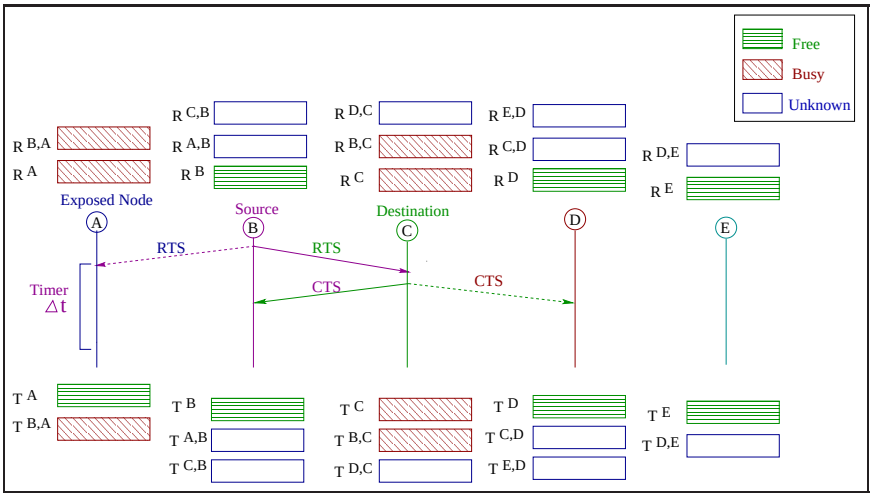

Figure 4.4: CTS packet from node C in response to RTS packet from node B.

Nodes other than the source node, on receiving CTS packet, in present scenario node D, set the transmitter and receiver of node C to *Busy* and also set its own transmitter to *Busy* for duration of data transmission. Node D, then broadcasts a

FCTS packet to its neighbor indicating that its transmitter is *Busy*, but receiver is *Free* for the duration of data transmission. This is depicted in Figure 4.5.

A node can send data directly without the exchange of RTS-CTS packets. The source must acknowledge the receipt of such data.

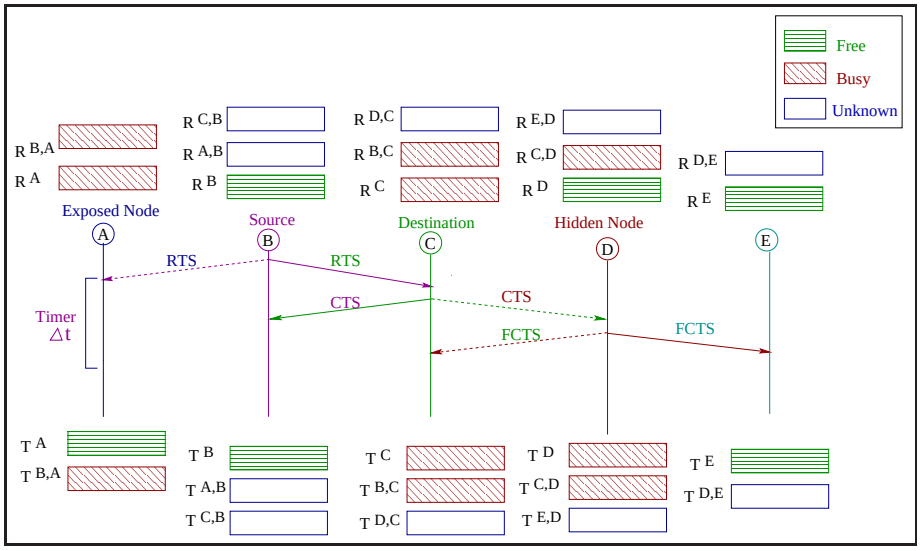

Figure 4.5: FCTS packet from node D on receiving CTS from node C.

On receiving the FCTS packet the nodes, in present scenario the node E, set the transmitter of node D to busy and the receiver of node D to free for duration of data transmission. This is shown in Figure 4.6.

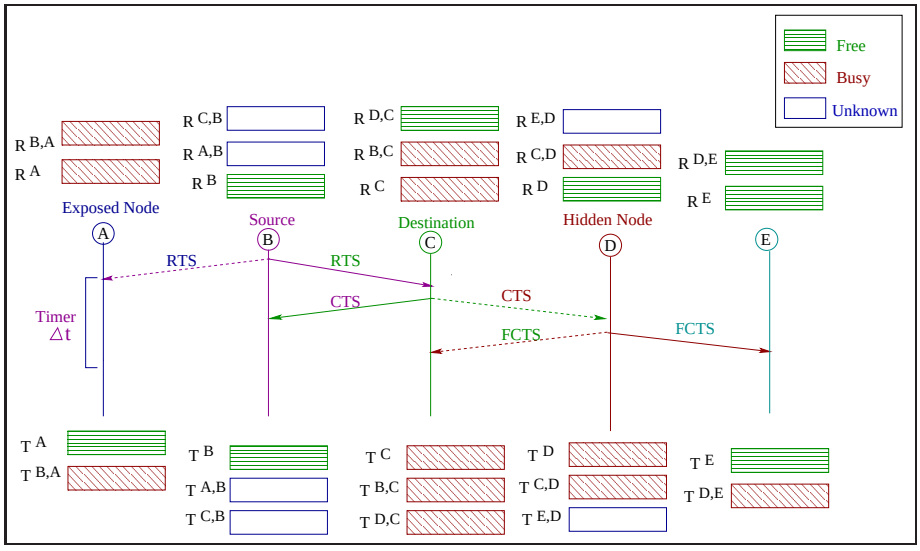

Figure 4.6: Processing of FCTS packet at node C and E

The source node, on receiving the CTS packet, sets the transmitter and receiver of node C and its own transmitter and receiver to busy for the duration of data transmission and schedules data transmission, as shown in Figure 4.7.

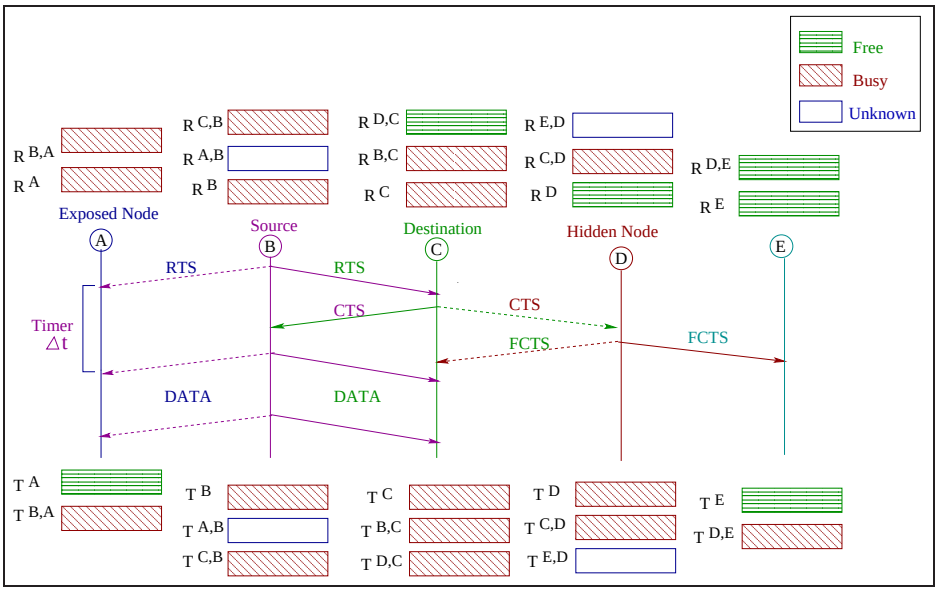

Figure 4.7: Processing of CTS packet at source node B

The neighbor of source node, on receiving DATA packet, in present scenario node A, checks the timer (Δt). If timer does not expire, stop the timer. Otherwise sets the transmitter and receiver of node B to *Free* and also set its own receiver to *Free*. This is illustrated in Figure 4.8.

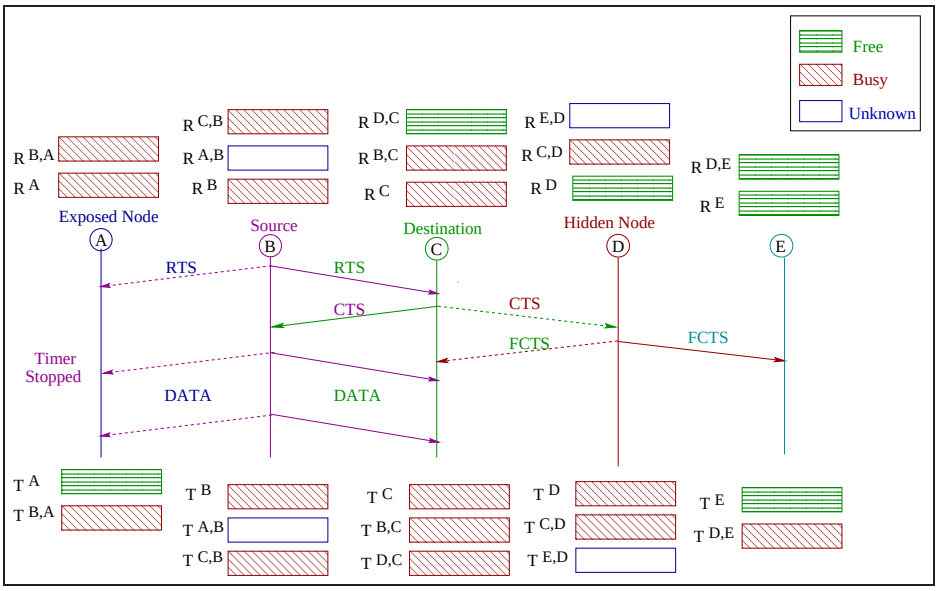

Figure 4.8: Processing of DATA packet at node A

The destination node C, on receiving the data packet send an ACK packet to the source B. This is shown in Figure-4.9.

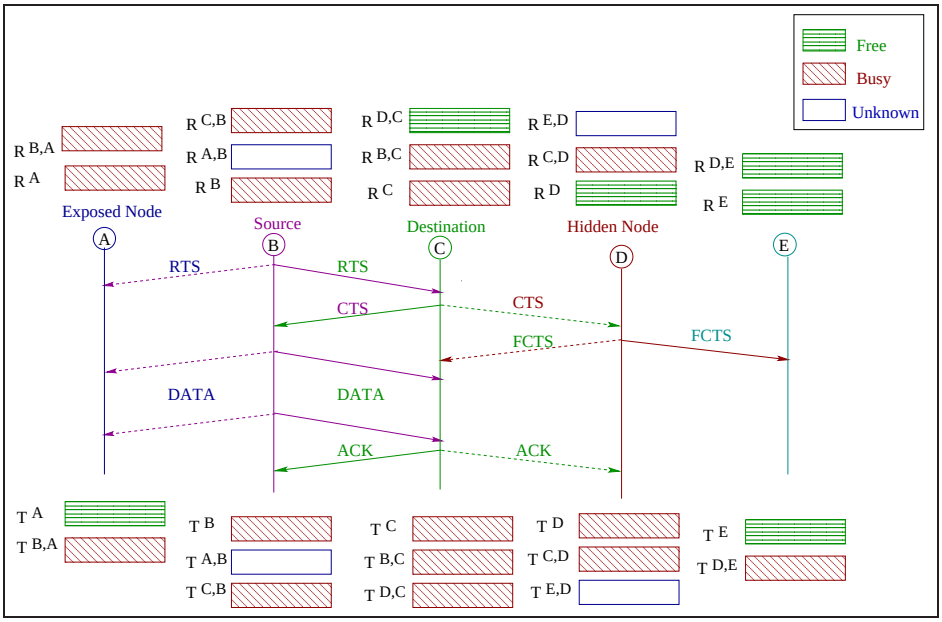

Figure 4.9: ACK packet from node C on receiving DATA packet

After the duration of transmission is over, each node set the status of other nodes transmitter and receiver to unknown, this is shown in Figure-4.10.

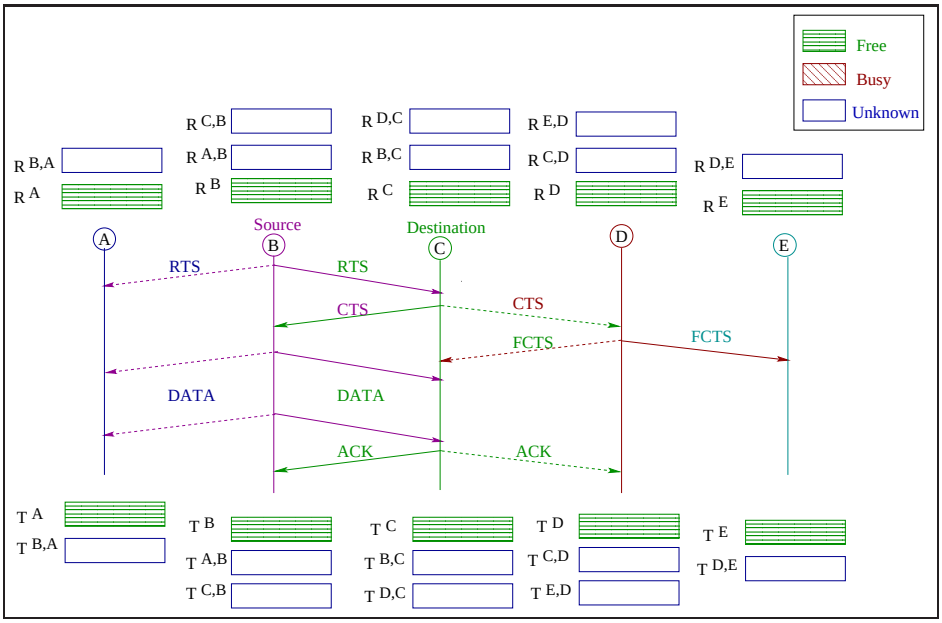

Figure 4.10: Status of all the nodes after Successful DATA transmission.

4.5.1 Analysis

This section analyzes the performance of the proposed scheme, to overcome the problems discussed in section 3.2, theoretically.

- ***Solution to Hidden Terminal Problem :*** As shown in Figure 4.5 upon receiving CTS packet from node C, node D (the hidden terminal of node B) sets its transmitter to *Busy* for the expected duration of data transmission. Thus node D do not transmit anything during the period, when B is sending DATA packet to node C. This prevents the hidden terminals of the intended sender to interfere with the reception at the receiver node. Moreover the proposed mechanism ensures no DATA packet are dropped due to hidden terminal problem.

- ***Solution to Exposed Terminal Problem :*** In Figure 4.3, node A is the exposed terminal of node B, when node B is the sender and node D is receiver. The node A, on overhearing the RTS packet from node B, sets its receiver to *Busy* and its transmitter is *Free* for the duration of data transmission. This mechanism allows node A to transmit its DATA packet directly when it gets a free receiver in its neighborhood. Thus, in the proposed scheme the exposed terminals could transmit DATA packets in parallel with the transmission in its 2-hop neighboring nodes, which enhance the spatial reuse of the channel.

- ***Solution to RTS-induced Problem :*** In the proposed scheme, upon reception of the RTS packet, non-destination nodes run a timer for duration Δt as depicted in Figure 4.3, during this period the source node B is expected to receive the CTS packet from node C and starts transmitting data. If the source node B does not transmit any DATA packet during that duration, due to unsuccessful CTS reception, then the non-destination node A resets its receiver to be *Free* and reattempts the channel access without deferring for the whole duration of data transmission. This guarantees the absence of RTS-induced problem.

- ***Solution to CTS-induced Problem :*** In Figure 4.5 the node D on receiving

the CTS packet, set its transmitter to *Busy* and its receiver to *Free*. If the CTS reception at source node B is unsuccessful, the node D will only defer from transmission during that period of time and is allowed for reception from its neighboring nodes which are present outside the range of receiver. This partially removes the CTS-induced problem.

In this work, without using the NAV setting method, we are allowing the exposed nodes to transmit during data transmission at its neighbor node. Upon RTS reception at non-destination nodes, without updating the Status Table, we are using a timer to check the successfulness of RTS/CTS exchange. By this, these non-addressed nodes are inhibited from reception only in successful RTS/CTS exchange and can transmit throughout that period, but when the RTS/CTS exchange is not successful, the nodes are allowed for transmission and reception without deferring. This solves the RTS-induced problem. Where as in traditional IEEE 802.11 using NAV setting method, the non-destination nodes those are exposed to the data transmission are deferred from transmission as well as reception during the whole period of time. Upon CTS reception at non-source nodes, the nodes are only deferred from transmission during the data transmission at their neighbor node and are allowed to receive the data during that period, without causing any collision at the receiver node. Here, we are sharing this information of the hidden nodes with their neighbors by broadcasting an FCTS packet.

Thus, in our proposed scheme we are avoiding the collision due to hidden terminals as well as utilizing the wasted bandwidth at these exposed and hidden nodes by allowing the exposed node to transmit and the hidden nodes to receive during data transmission at its neighbor node and also minimizing the possibility of RTS-induced and CTS-induced problems.

4.6 Summary

This chapter discusses the complete design of the proposed solution for the problems discussed in section 3.2. The solution is illustrated with the help of an example

and is analysed for solving the discussed problems while maintaining the level of fairness between the nodes that IEEE 802.11 has. The next chapter emphasizes on the performance evaluation of the proposed protocol and analyzes the simulation results.

Chapter 5

Performance Evaluation

In order to evaluate the performance of our proposed protocol, we provide extensive simulations for the system throughput, packet delivery ratio, and control packet overhead and compare the achieved results with IEEE 802.11 MAC.

5.1 Simulation Model

The simulations were performed using the Network Simulator (Version 2) [21, 22, 23], widely known as NS2, which is one of the well-known simulation tools. NS-2 is a discrete event-driven simulator, in which each event occurs at an instant in time, that has proved useful in studying the dynamic nature of communication networks. NS is an object oriented simulator, written in C++, with an OTcl interpreter as a front end. It supports large number of network protocols for simulation and provides results for wired, wireless and wired-cum-wireless scenarios, targeting at simulation research. To evaluate the performance of the proposed protocol, several simulations are performed. Table - 5.1 shows the simulation parameters.

5.2 Performance Metrics

Based on the simulation environment as depicted above, we obtained the performance comparison on terms of throughput, packet delivery ratio, and control overhead.

Simulation Time	500sec
Simulation Area	1000 × 1000
Number Of Nodes	5 - 20
Speed of Nodes	10 m/sec
Node Mobility	Random
Data Packet Size	Varying from 100 - 500 bytes
Radio Transmission Range	250m
Traffic Generated	CBR Traffic
Routing Protocol	DumbAgent
Radio Propagation Mode	Two-Ray Ground Propagation
Data Rate	1Mbps

Table 5.1: Simulation Parameter

- ***Average System Throughput :*** It is defined as the amount of MAC layer Service Data Unit(MSDU) transmitted per unit of time i.e. the amount of bits that can be transmitted in unit second. Throughput is calculated in kilo bits per second(kbps).

- ***Packet Delivery Ratio :*** The ratio between the number of received data packets at the intended destination node and the number of transmitted data packets at the source node. It specifies the packet loss rate, which limits the maximum throughput of the network. The better the delivery ratio, the more complete and correct is the MAC protocol.

- ***Average Control Overhead :*** The ratio between the total number of control packets to the data packets. The number of control packets means the number of transmitted RTS, CTS, FCTS, and ACK packets, for successfully transmitting a DATA packet from a source node to destination node.

5.3 Result Analysis

We plot the throughput vs pause time in Figure 5.1 to Figure 5.4 varying the number of nodes. It is observed from the figure that the proposed scheme has higher throughput.

This is due to the fact that in the proposed scheme, exposed nodes are allowed to transmit and hidden nodes are allowed to receive during the ongoing transmission.

It is seen from Figure 5.5 that the throughput increases with the increase in number of nodes in the proposed scheme.

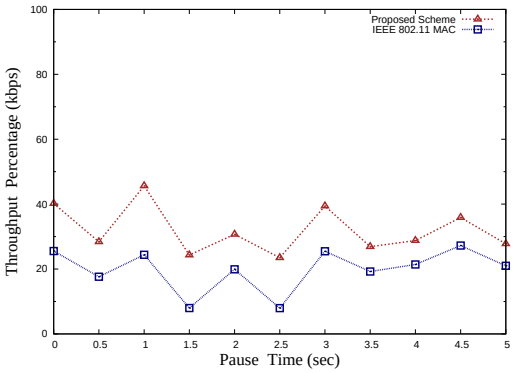

Figure 5.1: Throughput(kbps) Vs Pause Time(sec) for 5 Number of Nodes

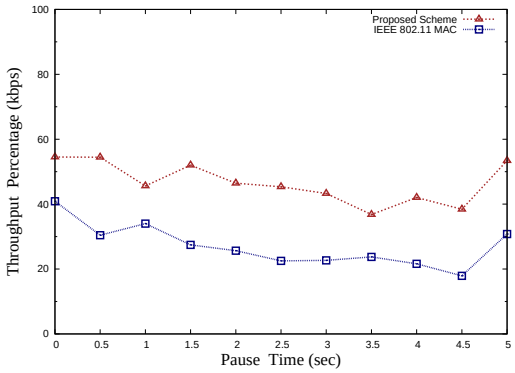

Figure 5.2: Throughput(kbps) Vs Pause Time(sec) for 10 Number of Nodes

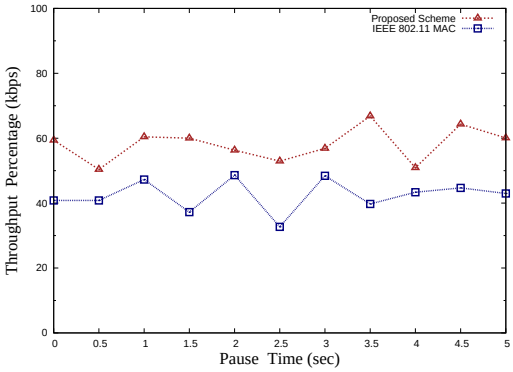

Figure 5.3: Throughput(kbps) Vs Pause Time(sec) for 15 Number of Nodes

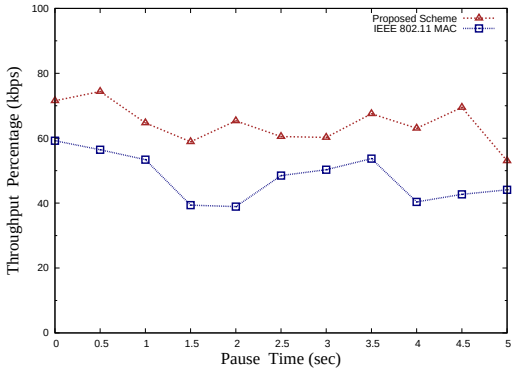

Figure 5.4: Throughput(kbps) Vs Pause Time(sec) for 20 Number of Nodes

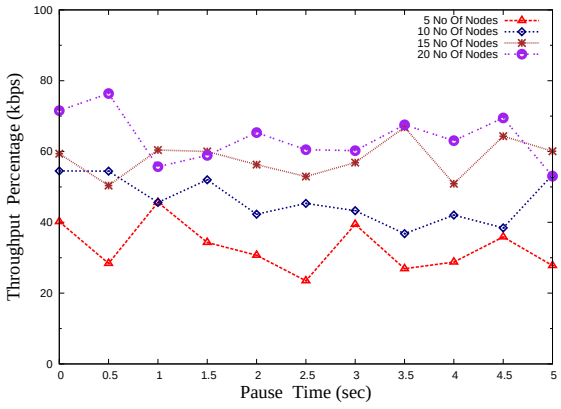

Figure 5.5: Throughput(kbps) Vs Pause Time(sec) for 5-, 10-, 15-, and 20- Number of Nodes

Figure 5.6 to figure 5.9 shows the plot for packet delivery ratio and pause time varying the number of nodes. It is observed from the figure that the proposed scheme also has higher packet delivery ratio as compared to the traditional IEEE 802.11 MAC.

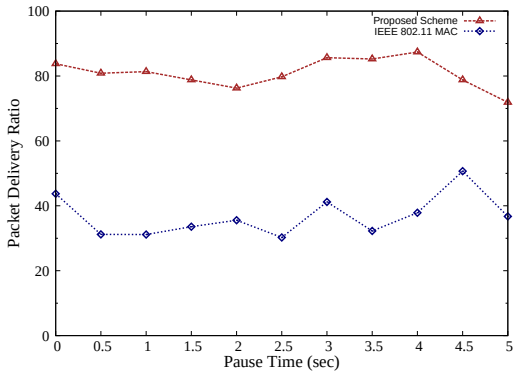

Figure 5.6: Packet Delivery Ratio Vs Pause Time for 5 Number of Nodes

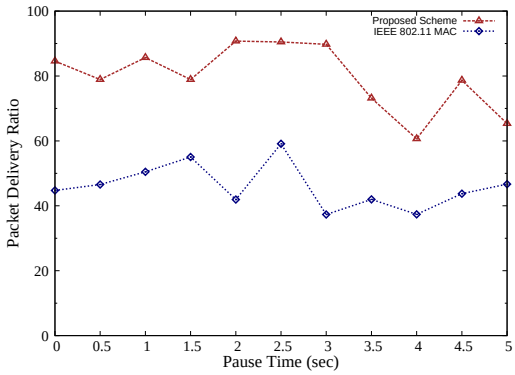

Figure 5.7: Packet Delivery Ratio Vs Pause Time for 10 Number of Nodes

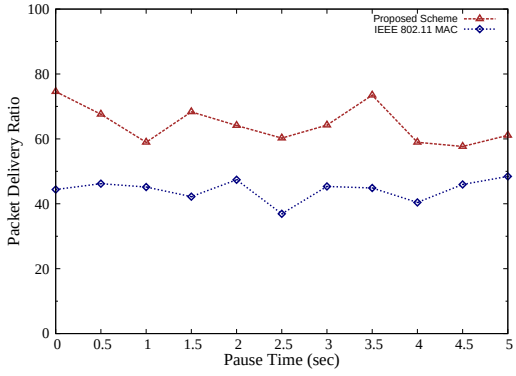

Figure 5.8: Packet Delivery Ratio Vs Pause Time for 15 Number of Nodes

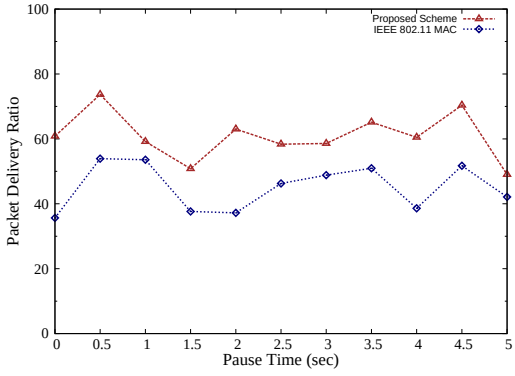

Figure 5.9: Packet Delivery Ratio Vs Pause Time for 20 Number of Nodes

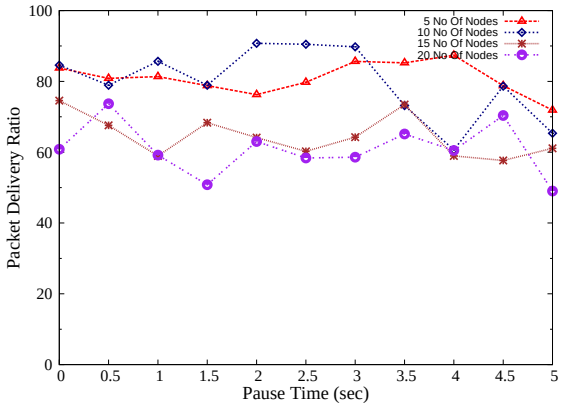

Figure 5.10: Packet Delivery Ratio Vs Pause Time for 5, 10, 15, and 20 Number of Nodes

Figure 5.11 illustrates the packet delivery ratio vs number of nodes for a pause time of 1.5sec. From the figure it can be seen that the packet delivery ratio is more when number of nodes are less and decreases gradually with increase in number of nodes and then saturates. This decrease is attributed to the resulting collision in RTS packet as the number of nodes increases.

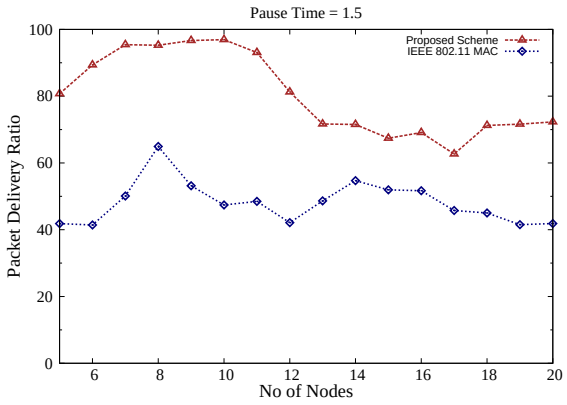

Figure 5.11: Packet Delivery Ratio Vs Number of Nodes

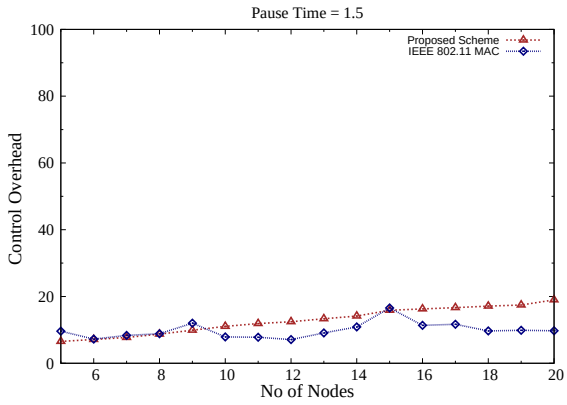

Figure 5.12: Control Overhead Vs Number of Nodes

In Figure 5.12 we plot the graph for control overhead vs number of nodes. The control overhead in the proposed scheme is marginally higher than the traditional IEEE 802.11 MAC. This is the expected result, as the proposed scheme uses an additional control packet.

5.4 Summary

This chapter evaluates the performance of the proposed scheme through simulation and analyzes the simulation results with the help of graphs. The simulation result shows that the proposed scheme gains performance improvement over the traditional scheme with a little increase in control overhead. The next chapter concludes the thesis and discusses further scopes of improvement.

Chapter 6

Conclusion and Future Work

This thesis discusses the mechanism of 802.11 DCF and identifies the limitations of this mechanism, prohibiting the exposed nodes to transmit and hidden nodes to receive, as discussed in literarture. This work is focussing on the utilization of bandwidth at the exposed and hidden terminals, which is wasted in the traditional 802.11 MAC. This is due to the fact that, 802.11 MAC does not keep any information about the ongoing transmissions in its neighborhood. The lack of overlapping transmission and reception at exposed and hidden nodes leds to degradation in throughput due to inefficient channel utilization. This work is also concentrating on two other problems, named as RTS/CTS-induced problem, caused due to unneccessary NAV settings at neighboring nodes, even if the RTS/CTS packets are not correctly exchanged.

6.1 Contribution

The default behaviour of 802.11 MAC is modified to solve these issues. In this work, we proposed a mechanism to improve the performance of 802.11 MAC protocol.

The proposed mechanism allows the nodes to share the information about the ongoing transmission by using one more control packet. Each node in the network are keeping the information about their neighbor nodes during the period of data transmission. The nodes then utilizing this information to schedule their own

transmission.

The proposed mechanism is implemented and simulated using Network Simulatior (Version-2). To measure the performance of the proposed mechanism, various network parameters such as throughput, packet delivery ratio, and control overhead is considered. In our proposed mechanism a hidden node can receive and an exposed node can transmit simultaneously with the ongoing transmission. With marginal increased in the control overhead, it outperforms the 802.11 DCF in terms of throughput and packet delivery ratio.

6.2 Future Work

There are significant scope for further improvements. The proposed scheme suffers from implementation issue. As the neighbor changes dynamically, it may be costly to keep the status of the neighboring nodes transmitter and receiver. The control overhead of the proposed scheme can be minimized further by restricting the trasmission of FCTS packets. The CTS-induced problem can be resolved fully by the use of some mechanism at the receiver node. These works are open for further developments of this research work.

Bibliography

[1] Imrich Chlamtac, Marco Conti, and Jennifer J.-N. Liu. Mobile ad hoc networking: imperatives and challenges. *Ad Hoc Networks, Elsevier*, 1(1):13–64, Jan-March 2003.

[2] Kaveh Ghaboosi, Matti Latva-aho1, and Yang Xiao. Receiver blocking problem in mobile ad hoc networks: Challenges & solutions. In *Second International Conference on Future Generation Communication and Networking*, pages 279–282. IEEE, MAY 2008.

[3] Rsffaele Bruno, Marcn Conti, and Enncn Gregnri. Ieee 802.11 optimal performances: Rts/cts mechanism vs. basic access. *The 13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications - PIMRC 2002*, 4(1):1747–1751, 15-18 September 2002.

[4] C. Siva Ram Murthy and B. S. Manoj. *Ad Hoc Wireless Networks: Architectures and Protocols*. Prentice Hall, 2004.

[5] Leonard Kleinrock and Fouad A. Tobagi. Packet switching in radio channels: Part i-carrier sense multiple-access modes and their throughput-delay characteristics. *IEEE Transactions on Communications*, 23(12):1400–1416, December 1975.

[6] Norman Abramson. The aloha system—another alternative for computer communications. In *Fall Joint Computer Conference*, volume 37, pages 281–285, 1970.

[7] Fouad A. Tobagi and Leonard Kleinrock. Packet switching in radio channels: Part ii–the hidden terminal problem in carrier sense multiple-access and the busy-tone solution. *IEEE Transactions on Communications*, 23(12):1417–1433, December 1975.

[8] Phil Karn. Maca - a new channel access method for packet radio. In *Proccedings os ARRL/CRRL Amateur Radio Computer Networking Conference*, pages 134–140, September 1990.

[9] Chris Romans and Jean Tounilhes. A medium access protocol for wireless lans which supports isochronous and asynchronous traffic. In *The Ninth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications, 1998.*, volume 1, pages 147–152, September 1998.

[10] Vaduvur Bharghavan, Alan Demers, Scott Shenker, and Lixia Zhang. Macaw: A media access protocol for wireless lan's. In *SIGCOMM '94 Proceedings of the conference on Communications architectures, protocols and applications*, volume 24, pages 212–225, October 1994.

[11] Fabrizio Talucci, Mario Gerla, and Luigi Fratta. Maca-bi (maca by invitation)-a receiver oriented access protocol for wireless multihop networks. In *Personal, Indoor and Mobile Radio Communications, 1997. 'Waves of the Year 2000'. PIMRC '97., The 8th IEEE International Symposium on*, volume 2, pages 435–439, September 1997.

[12] Zygmunt J. Haas and Jing Deng. Dual busy tone multiple access (dbtma) - a multiple access control scheme for ad hoc networks. In *IEEE Transactions on Communications*, volume 6, pages 975–985, June 2002.

[13] LAN MAN Standards Committee of the IEEE Computer Society. *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.* ANSI/IEEE Std. 802.11, 1999 Eddition.

[14] Deepanshu Shukla, Leena Chandran-Wadia, and Sridhar Iyer. Mitigating the exposed node problem in ieee 802.11 ad hoc networks. In *Computer Communications and Networks, 2003. ICCCN 2003. Proceedings. The 12th International Conference on*, pages 157–162, October 2003.

[15] Chane L. Fullmer and J.J. Garcia-Luna-Aceves. Floor acquisition multiple access (fama) for packet-radio networks. In *SIGCOMM '95 Proceedings of the conference on Applications, technologies, architectures, and protocols for computer communication*, volume 25, pages 262–273, October 1995.

[16] Chane L. Fullmer and J. J. Garcia-Luna-Aceves. Solutions to hidden terminal problems in wireless networks. In *In Proceedings ACM SIGCOMM'97*, volume 27, pages 39–49, October 1997.

[17] Kaveh Ghaboosi and Babak Hossein Khalaj. Amaca - a new multiple access collision avoidance scheme for wireless lans. In *Wireless Ad-Hoc Networks, 2004 International Workshop on*, pages 238–242, May-June 2004.

[18] Tao Han and Weijie Lu. An improvement of maca in alleviating hidden terminal problem in adhoc networks. In *Wireless Communications, Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on*, pages 1–4. IEEE, September 2009.

[19] Lei Du and Lan Chen. Receiver initiated network allocation vector clearing method in wlans. In *Asia-Pacific Conference on Communications*, pages 616–619, October 2005.

[20] Giuseppe Bianchi. Performance analysis of the ieee 802.11 distributed coordination function. *IEEE Journal on Selected Areas in Communications*, 18(3):535–547, MARCH 2000.

[21] http://www.isi.edu/nsnam/ns/.

[22] Kevin Fall and Kannan Varadhan, editors. *The ns Manual(formerly ns Notes and Documentation).* LBNL's Network Simulator, December 2007.

[23] Teerawat Issariyakul and Ekram Hossain. *Introduction to Network Simulator NS2.* Spinger, 2009.

[24] Lei Du and Lan Chen. Receiver initiated network allocation vector clearing method in wlans. In *Asia-Pacific Conference on Communications*, pages 615–619, Perth, Western Australia, October 2005. IEEE.

[25] Apichet Chayabejara, Salahuddin Muhammad Salim Zabir, and Norio Shiratori. An enhancement of the ieee 802.11 mac for multihop ad hoc networks. In *Vehicular Technology Conference*, pages 3020–3024. IEEE 58th, May 2003.

[26] Shugong Xu and Tarek Saadawi. Does the ieee 802.11 mac protocol work well in multihop wireless ad hoc networks? *Communication Magazine, IEEE*, 39(6):130–137, June 2001.

[27] Mohammad Sayad Haghighi, Maryam Najimi, Kamal Mohamedpour, and Yousef Darmani. An adaptive network allocation vector for ieee 802.11-based multi-hop networks. In *Second International Conference on Future Generation Communication and Networking*, pages 279–282. IEEE, MAY 2008.

[28] Deepanshu Shukla, Leena Chandran-Wadia, and Sridhar Iyer. A markovian framework for performance evaluation of ieee 802.11. *IEEE Transactions on Wireless Communications*, 6(4):1276–1285, April 2007.

[29] Yunli Chen, Qing-An Zeng, and Dharma P. Agrawal. Performance analysis and enhancement for ieee 802.11 mac protocol. In *Communication Magazine, IEEE*, pages 860–867. IEEE, jhyg 2003.

[30] Ikramullah Khosa, Usman Haider, and Haris Mosood. Evaluating the performance of ieee 802.11 mac protocol using opnet modeler. In *International Conference on Electronics and Information Engineering (ICEIE 2010)*, pages 91–95. IEEE, 2010.

[31] Kaixin Xu a, Mario Gerla, and Sang Bae. Effectiveness of rts/cts handshake in ieee 802.11 based ad hoc networks. *Ad Hoc Networks, Elsevier*, 1(1):107–123, July 2003.