I, Tarun Joshi

hereby submit this work as part of the requirements for the degree of:

Doctor of Philosophy

in:

Computer Science and Engineering

It is entitled:

Cross Layer Optimizations for Performance Enhancement of Wireless Networks

This work and its defense approved by:

Chair: Dr. Dharma P. Agrawal

Dr. Kenneth Berman

Dr. Wen-Ben Jone

Dr. James Caffery

Dr. Tim Keener
Cross Layer Optimizations for Performance Enhancement of Wireless Networks

A dissertation submitted to
Division of Research and Advanced Studies
University of Cincinnati

In partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in the Department of
Electrical and Computer Engineering and Computer Science
College of Engineering
2006

by

Tarun Joshi
Bachelor of Engineering (Computer Science and Technology)
Indian Institute of Technology (IIT), Roorkee, India.
November 2002

Dissertation Advisor and Committee Chair: Dr. Dharma P. Agrawal
ABSTRACT

In this dissertation, we focus on designing several cross-layer optimizations to boost the performance of wireless networks. We categorize our efforts into two directions: (a) improvement of spatial reusability and multi-hop performance via Directional Antennas, and (b) analysis and optimization of the IEEE 802.11 multi-rate networks. Since existing protocols are incapable of fully exploiting the benefits of a directional antenna system, we propose a broadcast and routing protocol for such systems. All our proposals assume cross layer interaction between the Network, MAC and PHY layers. Next, we analyze multi-rate networks in the context of IEEE 802.11 DCF networks and propose an analytical model to study the link-delay characteristics of such systems. We then propose an online algorithm Time Fair CSMA (TFCSMA), for guaranteeing air-time fairness and thereby mitigating the recently observed rate anomaly problem of IEEE DCF multi-rate networks. Following the design of TFCSMA, we concentrate on the problem of rate adaptation for time-varying wireless channels. We thoroughly investigate the impact of transmission rate on the performance of a wireless link. We then propose Stochastic Automata Rate Adaptation Algorithm (SARA). SARA is inspired by Stochastic Learning Automata (SLA), a machine learning technique for adaptation in random environments. As opposed to the previous work in this area, SARA is ideally suited for both stationary and non-stationary channel environments and is completely compatible with the existing IEEE 802.11 MAC standard.
To my dear parents,
I express my sincere gratitude towards my advisor and mentor Dr. Dharma Prakash Agrawal for his constant support, guidance, and encouragement during the course of my PhD.

I would also like to thank my committee members Dr. James Caffery, Dr. Wen-Ben Jones, Dr. Ken Berman and Dr. Tim Keener for serving on my dissertation committee and providing me with inputs helpful suggestions. I acknowledge the Ohio Board of Reagents, the NSF, and the Department of Electrical and Computer Engineering and Computer Science (ECECS), that have helped to fund this research.

In the course of my graduate life at Cincinnati, I have had the benefit of interacting with fellow graduate students, and would like to thank them all for their invaluable suggestions. I would particularly like to thank all past and present members of the CDMC lab with whom I have collaborated and often discussed my research.

Lastly, I thank my parents, my sister and my brother-in-law for their constant support, and love during this journey.
Contents

LIST OF FIGURES........................................................................................................................................ VI

LIST OF TABLES........................................................................................................................................ VIII

1. INTRODUCTION........................................................................................................................................ 1
   1.1 ORGANIZATION OF THE DOCUMENT ..................................................................................................... 6

2. IEEE 802.11 STANDARDS.................................................................................................................. 7
   2.1 INTRODUCTION....................................................................................................................................... 7
   2.2 IEEE 802.11 ARCHITECTURE .................................................................................................................. 8
   2.3 IEEE 802.11 MEDIUM ACCESS CONTROL (MAC) SUB-LAYER .............................................................. 8
         2.3.1 IEEE 802.11 Point Coordination Function (PCF) .......................................................................... 9
         2.3.2 IEEE 802.11 Distributed Coordination Function (DCF) ............................................................... 9

3. DIRECTIONAL ANTENNA TECHNOLOGIES.................................................................................. 12
   3.1 INTRODUCTION..................................................................................................................................... 12
   3.2 ANTENNA BASICS ............................................................................................................................... 13
   3.3 THE ANTENNA MODEL......................................................................................................................... 13

4. BROADCASTING OVER DIRECTIONAL ANTENNAS ................................................................. 16
   4.1 INTRODUCTION.................................................................................................................................... 16
5.3 RELATED WORK ...................................................................................................................................... 40
5.4 DYNAMIC SOURCE ROUTING (DSR) PROTOCOL .................................................................................. 41
5.5 PROPOSED PROTOCOL ............................................................................................................................. 42
   5.5.1 Directional Routing Protocol (DRP) ................................................................................................. 42
   5.5.2 DRP Medium Access Control .......................................................................................................... 44
   5.5.3 DRP Route Discovery ......................................................................................................................... 44
   5.5.4 DRP Route Maintenance ..................................................................................................................... 45
5.5.5 Route Recovery Mechanisms ................................................................................................................ 47
   5.5.5.1 Location Tracking Mechanism ...................................................................................................... 47
   5.5.5.2 Local Route Recovery Mechanism .............................................................................................. 50
   5.5.5.3 Zonal Route Recovery Mechanism .............................................................................................. 53
5.6 PERFORMANCE EVALUATION .................................................................................................................. 55
   5.6.1 Simulation Model ............................................................................................................................. 55
   5.6.2 Route Discovery Latency ................................................................................................................... 55
   5.6.3 Routing Control Packet Redundancy .................................................................................................. 56
   5.6.4 Hop Length – Route Optimality ......................................................................................................... 57
   5.6.5 Throughput in Static Scenario .......................................................................................................... 57
   5.6.6 Route Recovery Mechanisms ............................................................................................................. 58
     5.6.6.1 Efficiency of Direction Estimation Algorithm ............................................................................. 59
     5.6.6.2 Location Tracking Mechanism .................................................................................................... 59
     5.6.6.3 Local Route Recovery Mechanism ............................................................................................ 60
     5.6.6.4 Zonal Route Recovery Mechanism ............................................................................................ 62
5.6.7 Mobile Random Scenario ....................................................................................................................... 63
5.7 CONCLUSION ........................................................................................................................................ 64
6. PERFORMANCE EVALUATION OF IEEE MULTI-RATE NETWORKS ..................................................... 66
   6.1 INTRODUCTION .................................................................................................................................. 66
   6.2 LINK DELAY CHARACTERISTIC OF MULTI-RATE NETWORKS ......................................................... 67
     6.2.1 Analytical Model for Link Delay Estimation ...................................................................................... 67
8.2 PRELIMINARIES ........................................................................................................................................... 103
  8.2.1 Effects of Transmission Rate on Performance .............................................................................. 103
  8.2.2 Rate Adaptation Objectives for 802.11 Networks ........................................................................ 105
8.3 RELATED WORK ..................................................................................................................................... 106
8.4 STOCHASTIC LEARNING AUTOMATA ............................................................................................... 109
  8.4.1 Introduction to Stochastic Learning Automata ............................................................................. 109
  8.4.2 General Approach ......................................................................................................................... 110
  8.4.3 Non Stationary Environment and Learning Automata ................................................................. 110
8.5 STOCHASTIC AUTOMATA FOR RATE ADAPTATION (SARA) ............................................................ 112
  8.5.1 Algorithm Outline ......................................................................................................................... 112
  8.5.2 Probabilistic Packet Success Rate Estimator (PPSRE) ................................................................. 116
8.6 PERFORMANCE EVALUATION ............................................................................................................... 119
  8.6.1 Stationary Channel ....................................................................................................................... 120
  8.6.2 Stationary Channel-Random Scenario ......................................................................................... 121
  8.6.3 Non-Stationary Channel Conditions ......................................................................................... 123
8.7 CONCLUSION 125

9. CONCLUSION AND FUTURE DIRECTIONS ............................................................................................ 127
## List of Figures

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIGURE 1.1</td>
<td>SCHEMATIC OUTLINE OF THE DISSERTATION</td>
<td>5</td>
</tr>
<tr>
<td>FIGURE 2.1</td>
<td>IEEE 802.11 MAC SUB-LAYER</td>
<td>9</td>
</tr>
<tr>
<td>FIGURE 2.2</td>
<td>IEEE 802.11 DCF BASIC ACCESS METHOD</td>
<td>10</td>
</tr>
<tr>
<td>FIGURE 2.3</td>
<td>IEEE 802.11 DCF RTS/CTS METHOD</td>
<td>10</td>
</tr>
<tr>
<td>FIGURE 3.1</td>
<td>SWITCHED SINGLE BEAM ANTENNA</td>
<td>14</td>
</tr>
<tr>
<td>FIGURE 4.1</td>
<td>NEDF</td>
<td>25</td>
</tr>
<tr>
<td>FIGURE 4.2</td>
<td>PRB</td>
<td>26</td>
</tr>
<tr>
<td>FIGURE 4.3</td>
<td>CONNECTIVITY: 6 ANTENNA BEAM</td>
<td>28</td>
</tr>
<tr>
<td>FIGURE 4.4</td>
<td>CONNECTIVITY: 12 ANTENNA BEAM</td>
<td>28</td>
</tr>
<tr>
<td>FIGURE 4.5</td>
<td>CONNECTIVITY: 18 ANTENNA BEAM</td>
<td>28</td>
</tr>
<tr>
<td>FIGURE 4.6</td>
<td>FORWARDING NODES: 6 ANTENNA</td>
<td>28</td>
</tr>
<tr>
<td>FIGURE 4.7</td>
<td>FORWARDING NODES: 12 ANTENNA</td>
<td>29</td>
</tr>
<tr>
<td>FIGURE 4.8</td>
<td>FORWARDING NODES: 18 ANTENNA</td>
<td>29</td>
</tr>
<tr>
<td>FIGURE 4.9</td>
<td>LATENCY: 6 ANTENNA BEAM</td>
<td>29</td>
</tr>
<tr>
<td>FIGURE 4.10</td>
<td>LATENCY: 12 ANTENNA BEAM</td>
<td>29</td>
</tr>
<tr>
<td>FIGURE 4.11</td>
<td>LATENCY: 18 ANTENNA BEAM</td>
<td>30</td>
</tr>
<tr>
<td>FIGURE 4.12</td>
<td>TRANSMITTED PACKETS: 6 ANTENNA BEAM</td>
<td>30</td>
</tr>
<tr>
<td>FIGURE 4.13</td>
<td>TRANSMITTED PACKETS: 12 ANTENNA BEAM</td>
<td>30</td>
</tr>
<tr>
<td>FIGURE 4.14</td>
<td>TRANSMITTED PACKETS: 18 ANTENNA BEAM</td>
<td>30</td>
</tr>
<tr>
<td>FIGURE 5.1</td>
<td>DIRECTIONAL ARP ISSUES</td>
<td>36</td>
</tr>
<tr>
<td>FIGURE 5.2</td>
<td>EFFECT OF NODE MOVEMENT AND MULTI-HOP PACKET FORWARDING</td>
<td>37</td>
</tr>
<tr>
<td>FIGURE 5.3</td>
<td>DRP ILLUSTRATION</td>
<td>43</td>
</tr>
<tr>
<td>FIGURE 5.4</td>
<td>INTERMEDIATE NODE MOVEMENT</td>
<td>49</td>
</tr>
<tr>
<td>FIGURE 5.5</td>
<td>MOVEMENT OF NODE Y FROM BEAM 0 TO 7</td>
<td>50</td>
</tr>
<tr>
<td>FIGURE 5.6</td>
<td>LOCAL ROUTE RECOVERY MECHANISM</td>
<td>52</td>
</tr>
<tr>
<td>FIGURE 5.7</td>
<td>TIMER INTERPLAY AT NODE X</td>
<td>52</td>
</tr>
<tr>
<td>FIGURE 5.8</td>
<td>DRP ALGORITHM</td>
<td>54</td>
</tr>
<tr>
<td>FIGURE 5.9</td>
<td>ROUTE DISCOVERY LATENCY</td>
<td>56</td>
</tr>
<tr>
<td>FIGURE 5.10</td>
<td>REDUNDANCY DUE TO DIRECTIONAL BROADCAST</td>
<td>56</td>
</tr>
<tr>
<td>FIGURE 5.11</td>
<td>HOP LENGTH WITH INTER-NODE SEPARATION</td>
<td>56</td>
</tr>
<tr>
<td>FIGURE 5.12</td>
<td>THROUGHPUT FOR STATIC SCENARIO</td>
<td>58</td>
</tr>
<tr>
<td>FIGURE 5.13</td>
<td>DRP LOCATION TRACKING</td>
<td>60</td>
</tr>
<tr>
<td>FIGURE 5.14</td>
<td>DRP LOCAL ROUTE RECOVERY</td>
<td>61</td>
</tr>
<tr>
<td>FIGURE 5.15</td>
<td>ZONAL ROUTE RECOVERY</td>
<td>63</td>
</tr>
<tr>
<td>FIGURE 5.16</td>
<td>MOBILE RANDOM SCENARIO</td>
<td>65</td>
</tr>
<tr>
<td>FIGURE 6.1</td>
<td>CLASSIFICATION OF NODES ON BASIS OF DATA RATES</td>
<td>72</td>
</tr>
<tr>
<td>FIGURE 6.2</td>
<td>THROUGHPUT DEPENDENCE ON CW_{MIN}</td>
<td>74</td>
</tr>
<tr>
<td>FIGURE 6.3</td>
<td>AGGREGATE THROUGHPUT</td>
<td>74</td>
</tr>
</tbody>
</table>
List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table 5.1</td>
<td>Antenna beams to be scanned for local route recovery</td>
<td>50</td>
</tr>
<tr>
<td>Table 5.2</td>
<td>Efficiency of direction estimation</td>
<td>59</td>
</tr>
<tr>
<td>Table 6.1</td>
<td>IEEE 802.11B parameters</td>
<td>72</td>
</tr>
<tr>
<td>Table 7.1</td>
<td>Aggregate throughput (KBPS)</td>
<td>84</td>
</tr>
<tr>
<td>Table 7.2</td>
<td>Per table</td>
<td>88</td>
</tr>
<tr>
<td>Table 7.3</td>
<td>Snapshot of offline throughput table</td>
<td>90</td>
</tr>
<tr>
<td>Table 7.4</td>
<td>Target throughput table (flows vs number of nodes)</td>
<td>94</td>
</tr>
<tr>
<td>Table 7.5</td>
<td>Aggregate throughput results</td>
<td>95</td>
</tr>
<tr>
<td>Table 7.6</td>
<td>Fairness index</td>
<td>96</td>
</tr>
<tr>
<td>Table 7.7</td>
<td>Simulation setup</td>
<td>96</td>
</tr>
<tr>
<td>Table 7.8</td>
<td>Bias probability</td>
<td>96</td>
</tr>
<tr>
<td>Table 7.9</td>
<td>Offline table: $T_{psr}$</td>
<td>116</td>
</tr>
<tr>
<td>Table 8.1</td>
<td>Snapshot trial table: $T_{trial}$, PSR=0.5</td>
<td>118</td>
</tr>
<tr>
<td>Table 8.2</td>
<td>Sara parameters</td>
<td>119</td>
</tr>
<tr>
<td>Table 8.3</td>
<td>IEEE 802.11G</td>
<td>119</td>
</tr>
<tr>
<td>Table 8.4</td>
<td>Channel success rates for data rates. Rates in MBPS</td>
<td>121</td>
</tr>
<tr>
<td>Table 8.5</td>
<td>Throughput comparison for random channels</td>
<td>122</td>
</tr>
<tr>
<td>Table 8.6</td>
<td>Channel success rates for data rates. Rates in MBPS</td>
<td>123</td>
</tr>
</tbody>
</table>
Chapter 1

1. Introduction

Wireless networks provide an attractive alternative to their wired counterparts since they can support the vision of ubiquitous computing. As a result, there has been a proliferation of wireless devices in the market recently. Recent applications of such networks have also gained tremendous popularity in the defense sector as well.

However, there are several fundamental differences between traditional wired LANs and wireless networks. For example, wireless networks support user mobility. Therefore, while an address in a wired environment corresponds to a physical location, the same denotes a message destination in WLANs. Secondly, the Wireless medium is much more susceptible to channel errors. This is due to the time-varying nature of the wireless channel. Thirdly, wireless transmission involves the distribution of energy in the space, which increases the chances of collisions in the network. It also restricts the spatial reuse or the number of simultaneous transmissions in the network. Finally, unlike their wired counterparts, wireless devices cannot listen to the channel to track their own transmissions. Therefore, such networks require proactive protocols to arbitrate the use of the medium.

A Medium Access Control (MAC) moderates the access to a wireless channel. The primary objective of the MAC is to define a set of rules for the channel access. A MAC is required to
provide high throughput while guaranteeing fairness in the network. The Physical Layer (PHY) is responsible for actually delivering the data payload over the channel. Finally, the routing layer is required to discover and maintain routes between the source and destination nodes.

The performance of the routing layer; MAC layer and the physical layer is closely coupled, especially in the case of wireless networks. Physical layer parameters like data rate; transmit power and carrier sensitivity have a huge impact on the performance of the MAC layer. Since the MAC determines the expected link-delay, it also affects the throughput of the routing or network layer. Since issues at one layer affect the performance of other layers, cross-layered designs are encouraged for wireless networks to boost performance. The central idea in these designs is that channel knowledge at one layer can help optimize parameters at another layer. For example, feedback information about a transmission rate at the MAC can help in deciding the optimal modulation at PHY.

Traditionally, wireless networks have been used for cellular telephony. Recently, WLANs [IEEE97] have become popular. They provide single hop connectivity to the internet by eliminating the use of Ethernet or cables. However, the single hop constraint limits their range. Therefore, the concept of mesh-networking has been proposed which allows a multi-hop access to the internet, thereby increasing the area and range of such networks. Other types of wireless networks facilitate peer to peer interaction between the nodes, allowing the setup of an infrastructure-less ad-hoc network. These are popularly known as Mobile Ad-Hoc Networks or MANETs.

This dissertation focuses on optimizing the performance of such networks by exploring cross layer mechanisms which encourage interactions between different protocol stack layers. It
primarily explores the interplay between the MAC and PHY, and the MAC and Routing layers of the wireless protocol stack. Figure 1.1 shows a schematic outline of this dissertation.

We categorize our efforts into two directions: (a) improvement of spatial reusability via Directional Antennas, and (b) analysis and optimization of the IEEE 802.11 multi-rate networks. The first part deals with use of directional antennas to improve the performance of broadcasting [Joshi04] and route recovery [Joshi05] at the routing layer. Traditional wireless networks employ omni-directional antennas, which severely limit the capacity of networks. Directional antennas offer an increased spatial reuse and a larger coverage range as transmitted power is concentrated in a small angular region. However, in a particular switched single beam antenna model (the model used in our research) broadcasting is achieved by sequentially steering the antenna beam across all pre-defined directions, resulting in a sweeping delay. We study suggested techniques to alleviate broadcast storm problems in ad hoc networks, point out their shortcomings, and propose two new schemes, namely, the New Enhanced Directional Flooding (NEDF) and the Probabilistic Relay Broadcasting (PRB). As opposed to omni directional broadcasting, directional schemes are highly effective in sparse networks as they offer great connectivity and low latencies. However, there are greater redundancies too.

We then focus on the issue of routing in mobile ad hoc networks (MANETs) using directional antennas [Gossain06, Joshi05]. Existing directional routing schemes either assume a complete network topology beforehand or simply use omni-directional routing schemes to forward packets in underlying directional environment. We propose a Directional Routing Protocol (DRP) for MANETs. DRP is an on-demand directional routing protocol which assumes a cross layer interaction between routing and MAC layer and is inspired by Dynamic Source Routing (DSR) protocol. The main features of DRP include an efficient route discovery mechanism,
establishment and maintenance of directional routing and directional neighbor tables (DRT and DNT respectively) and novel directional route recovery mechanisms. We have implemented DRP on top of MDA, a MAC protocol for directional antennas and have compared its performance with the DSR protocol over both omni-directional and directional antenna models. Our results show that DRP considerably improves the packet delivery ratio, decreases the end to end packet latency, has lesser routing overhead and is robust to link failures.

Next, we shift our attention towards the cross layer interaction between the MAC and the Physical layer. We investigate the impact on the network performance when different nodes use different data rates. Under a multi rate network scenario, the IEEE 802.11 DCF MAC fails to provide air-time fairness for all competing nodes since the protocol is designed for ensuring throughput fairness. As a consequence, the maximum achievable throughput by any station gets bounded by the slowest transmitting peer. This obviously limits the advantage of employing higher transmission data rates. We present an analytical model to study the delay and throughput characteristics of such networks [Joshi06]. We then propose an online algorithm for guaranteeing air-time fairness and thereby mitigating the rate anomaly problem of IEEE DCF multi-rate networks. We call our proposal Time Fair CSMA (TFCSMA) [Joshi06a]. TFCSMA utilizes an interesting baseline property for estimating a target throughput for each competing station in the network. Each station, then dynamically adjusts its minimum contention window to achieve this target, thereby guaranteeing air time fairness in a distributed manner. As opposed to the previous work in this area, TFCSMA is ideally suited for practical scenarios where nodes frequently adapt their data rates to changing channel conditions. In addition, TFCSMA also accounts for packet errors due to the time varying properties of the wireless channel. We thoroughly compare the performance of our proposed protocol with IEEE 802.11 and other existing protocols, under
different network scenarios and traffic conditions. Our comprehensive simulations validate the efficacy of our method towards providing high throughput and time fair channel allocation.

![Figure 1.1 Schematic Outline of the Dissertation](image)

Finally, we concentrate on the problem of rate adaptation for time-varying wireless channels. Sub-optimal data rates can adversely affect the performance of wireless networks. Therefore, rate adaptation techniques have been proposed to dynamically adjust the modulation to the time-varying wireless channel. Existing rate adaptation algorithms can broadly be classified under two categories: (a) Signal to Noise Ratio measurement based and (b) Statistical Count based. While the former suffers from inaccurate estimations, the latter utilizes pre-defined thresholds for dynamically varying the rate, rendering the adaptation process across varied channel conditions erroneous. In this work [Joshi06b], we first analyze the impact of transmission rate on the performance of a wireless link. Specifically, we emphasize that performance may either improve or deteriorate with increasing transmission rates and hence no specific trends can be generalized for the behavior of performance versus rate. Based on our observations, we then propose...
Stochastic Automata Rate Adaptation Algorithm (SARA). SARA is inspired by Stochastic Learning Automata (SLA), a machine learning technique for adaptation in random environments. SARA assigns a selection probability to each of the transmission rates. It then randomly selects a rate for a transmission attempt and dynamically updates the probabilities based on the obtained feedback from the receiver (ACK/NACK), thereby obviating the need for explicit channel estimation or predefined thresholds. As opposed to the previous work in this area, SARA is ideally suited for both stationary and non-stationary channel environments and is completely compatible with the existing IEEE 802.11 MAC standard. We thoroughly compare the performance of our proposed protocol with Automatic Rate Fallback (ARF) and Adaptive ARF (AARF), under different channel scenarios. It is observed that SARA provides better throughput and faster convergence in both stationary and non-stationary environments.

1.1 Organization of the document

The remainder of this document is organized as follows. Chapter 2 discusses the IEEE 802.11 family of standards. An overview of directional antenna systems is provided in chapter 3. Two new protocols for broadcasting over directional antennas are discussed in chapter 4. Our routing optimizations under a directional medium are investigated in chapter 5. Chapter 6 presents an analytical model to study the performance of IEEE multi-rate networks. We present an air-time fair extension to IEEE 802.11 DCF in chapter 7. Chapter 8 describes the functioning of SARA, a stochastic rate adaptation algorithm. The conclusions and the future course of this dissertation are discussed in chapter 9.
Chapter 2

2. IEEE 802.11 Standards

2.1 Introduction

The IEEE 802.11 is the de-facto standard for the family of Wireless Local Area Networks (WLANs). Vendors follow specifications laid out by the standard body while providing products in the market.

For acceptable performance, wireless networks mandate the need for proper arbitration of the wireless channel. A WLAN Medium Access Control (MAC) layer is responsible for providing this function. A MAC layer should be compatible with different Physical layer specifications. This is necessary to seamlessly benefit from the physical layer advances.

The IEEE formed the working group (WG) 802.11 in the 1990s to provide a standard for wireless communication. The initial draft of the standard for wireless LANs was finalized by 1997. The draft covers the lower two layers of the TCP/IP protocol stack: PHY and MAC sublayer. The growing popularity of the 802.11 based WLANs has led to several WG and study groups within the IEEE 802.11 family to enhance the basic standard to support different services.
2.2 IEEE 802.11 Architecture

The IEEE 802.11 standard specifies two different modes of operation for wireless networks: the infrastructure-based mode and the ad hoc mode. Nodes are generally termed as stations (STA). A Basic Service Set (BSS) comprises of STAs within the contention range of each other. Special STAs known as Access Points (AP) acts as a bridge between the wired and wireless network. This is termed as the infrastructure-mode of operation. A BSS may also function without an AP, if peer to peer interaction among the STAs is allowed. This is known as the Ad Hoc mode of operation. Multiple BSSs can be connected to the backbone internet gateway node through wired links, forming an Extended Service Set (ESS).

This dissertation focuses heavily on the performance and analysis of the IEEE 802.11 MAC layer. In the following subsections we briefly overview the mechanism of the IEEE 802.11 MAC layer. This will serve as foundation to understand the optimizations suggested later in this dissertation. In the remainder of this document, we refer to stations as nodes, unless mentioned otherwise.

2.3 IEEE 802.11 Medium Access Control (MAC) Sub-layer

A MAC protocol arbitrates the use of a shared channel amongst the nodes in a wireless network. In IEEE 802.11, MAC protocol follows the popular Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA) procedure. Nodes sense the channel (or the carrier) prior to transmission and proactively try to avoid collisions. This is a departure from the conventional wired networks, which follow the CSMA/CD or Carrier Sense Multiple Access/Collision Detection mechanism. Wired networks allow nodes to sense their own transmissions to detect the possibility of collisions. This however is not possible under a wireless environment.
The channel can alternate between a contention period (CP), and a contention free period (CFP). During a CP, all nodes can contend to access the channel, whereas, a CFP uses a centralized controller to arbitrate the channel access. IEEE 802.11 specifies two medium access control protocols, Point Coordination Function (PCF) and Distributed Coordination Function (DCF). PCF uses polling and is a centralized access mechanism. DCF on the other hand, facilitates channel access in a totally distributed asynchronous manner.

2.3.1 IEEE 802.11 Point Coordination Function (PCF)

PCF is an optional capability, which provides a contention free, connection-oriented frame transfer. PCF uses a Point Coordinator (or PC) to poll nodes during a CFP. Typically, the functions of PC are performed by the AP in a BSS. PCF can co-exist with the DCF and logically sits on top of DCF. Figure 2.1 provides a schematic for the IEEE MAC sub-layer.

2.3.2 IEEE 802.11 Distributed Coordination Function (DCF)

The IEEE 802.11b standard provides for both a distributed access mechanism and a contention free arbitration access method for media access control. While the DCF uses CSMA/CA for channel access, the PCF provides for contention free access via an arbitrator that resides in access points. The standard allows both the methods to co-exist, a contention period followed by
a contention free period. The DCF access mechanism follows the CSMA/CA principle: a node wishing to transmit senses the channel prior to transmitting. If the channel is monitored idle for a period of Distributed Inter-Frame Space (DIFS), the node transmits. Otherwise, it continues to monitor the channel until it is sensed idle for a period of DIFS. It then enters the collision avoidance phase, where it generates a random back-off interval (exponential back-off scheme) to minimize the probability of collision with other contending nodes. For every packet transmission, the back-off period is uniformly chosen in the range \((0, w-1)\). The value of \(w\) is called the contention window and its value is contingent on the number of failed transmission attempts for the packet. For the first packet transmission, \(w\) is set equal to a value of \(CW_{min}\) called the minimum contention window. With every unsuccessful transmission, \(w\) is doubled till a maximum value of \(CW_{max} = 2^m CW_{min}\). The value of \(m\) is a tunable parameter. During the collision avoidance phase, the back-off counter is decremented as long as the channel is sensed idle. If a transmission is detected on the channel, the counter is "frozen" and reactivated when the channel is sensed idle again. Once the counter reaches 0, the node transmits.

A successful packet reception is marked by the transmission of a positive acknowledgement (ACK) by the destination node after another fixed period of time (SIFS- Short InterFrame Space). The failure of reception of an ACK frame is assumed as a collision. Figure 2.2 illustrates the contention process for three nodes attempting to transmit using the DCF procedure. Node A
selects a back-off window of 8 slots. However it has to freeze at slots 6 and 2, since at these instants node B and node C complete their respective back-offs and hence transmit. The station then retries transmitting the packet by doubling its contention window and following the above procedure again.

A common problem with the basic access method is that of a hidden terminal. This occurs when two nodes transmitting to the same destination are not within hearing range of each other. To overcome this, DCF defines an additional four way handshaking procedure known as the RTS/CTS mechanism. This access mechanism strives to reserve the channel for the entire packet transmission duration and hence minimizes the impact of collisions. The additional cost is a slightly increased transmission overhead (i.e. RTS/CTS frame exchange). Figure 2.3 depicts the RTS/CTS mechanism. The sender first transmits a request to send (RTS) instead of the data. The destination sends a clear-to-send (CTS) after receiving the RTS and waiting for SIFS. When the sender receives the CTS, it waits for SIFS again and transmits the data. The rest of the protocol is identical to the basic access method. The RTS-CTS method reduces the overhead incurred due to collisions as the collision time is now limited to RTS collisions in most cases.

With this overview of the IEEE 802.11, we study directional antenna systems in the next chapter. We specifically concentrate on a switched single beam antenna, the model used in this work. Chapter 3 also discusses some of the major issues that arise in directional communication and helps to understand the trade-offs involved in the use of such systems.
Chapter 3

3. Directional Antenna Technologies

3.1 Introduction

 Omni-Directional Antennas behave as an isotropic source and distribute energy in all directions other than just the intended direction. This has two consequences: (i) unnecessary interference is generated towards other nodes, and (ii) the transmissions range gets restricted. This can collectively reduce the capacity of wireless networks.

 Directional communication can improve the capacity of the system. Directional antennas concentrate energy towards the intended receivers and thereby reduce interference; offer a longer transmission range and also promote spatial reuse in the network.

 However, existing MAC and routing layer protocols for wireless networks have been designed assuming an omni-directional antenna at the physical layer. Therefore, careful modifications to existing protocols need to be done if the benefits of directional communication have to be exploited for wireless networks.
3.2 Antenna Basics

Radio antennas couple electromagnetic energy from one medium to another. Specifically, they transfer energy from the space to the receiver circuit via a wire or a coaxial cable. While omni-directional antennas behave as isotropic sources and distribute energy evenly in all directions; directional antennas concentrate energy in a fixed region in a space. The directionality of an antenna is qualified by a dimensionless quantity termed as the “antenna gain”. Given a direction $\vec{d}=(\theta,\phi)$, the gain of a directional antenna is specified by [Ramanathan01]:

$$G(\vec{d})=\eta \frac{U(\vec{d})}{U_{avg}}$$

where $U(\vec{d})$ is the power density in the direction $\vec{d}$ and $U_{avg}$ is the power density in all the directions. The efficiency of an antenna is defined by $\eta$. The gain indicates the relative power in a particular direction as compared to an omni-directional antenna. Higher directionality corresponds to a higher gain. The units of gain are decibels (dbi).

An Antenna Pattern is a theoretical concept to characterize the gain in each direction in space. Typically, an antenna pattern comprises of a main lobe (of peak gain) and side lobes (smaller gain). While the majority of the transmitted energy is concentrated around the main lobe, the side lobes can account for unwanted interference.

Another important concept with antenna systems is beam width. Beam width refers to the angle subtended by the two directions on either side of the direction of the peak gain that are 3 dB down in gain. For directional antennas, lower beam width implies a greater gain.

3.3 The Antenna Model

We have implemented a complete and flexible directional antenna module in Network Simulator (NS – version 2.26) [ns226]. Our module implements a comprehensive switched beam directional antenna system which is cheaper in cost compared to both adaptive array and
Multiple Input Multiple Output (MIMO) systems. This model can operate in two separate modes: Omni and Directional. This may be seen as two separate antennas: an omni-directional and a single switched beam antenna which can point towards specified directions. The Omni mode is used for reception only, while the Directional mode is used for both transmission as well as reception.

In Omni mode, a node is capable of receiving signals from all the directions with a gain of $G^O$. While idle (i.e., neither transmitting nor receiving), a node stays in the Omni mode. As soon as a signal is sensed, a node can detect the direction (beam) through which the signal is strongest and goes into the Directional mode in that particular direction.

![Switched Single Beam Antenna](image)

Figure 3.1 Switched Single Beam Antenna

In the Directional mode, a node can point its beam towards a specified direction with gain $G^d$ (with $G^d$ typically greater than $G^O$). In addition, the gain is proportional to number of antenna beams (i.e., inversely proportional to the beam-width) given that more energy can be focused towards a particular direction, thus resulting in increased coverage range. A node provides coverage around it by a total of $M$ non-overlapping beams. The beams are numbered from 1 through $M$, starting at the three o’clock position and running counter clockwise. At a given time,
a node can transmit or receive in only one of these antenna beams. In order to perform a broadcast, a transmitter may need to carry out as many directional transmissions as there are antenna beams so as to cover the whole region around it. This is called *sweeping*. In the sweeping process, we assume there is negligible delay in beam-forming for various directions. This model has been widely studied in the literature [Roy02a, Ko00, Agrawal04, Roy02].

To model antenna side lobes, we assume that the energy contributed to the side lobes is uniformly distributed in a circular area. [Roy02]. Although energy contributed to the side lobes depends on the actual radiation pattern, which is governed by the configuration and weighting of elements in the antenna array, for the purpose of our simulation we assume that the side lobe gain is fixed and is set to a very small value. Finally, we assume that all the nodes use the same directional antenna patterns and can maintain the orientation of their beams at all times [Hu03].

In the next chapter we investigate the issues in directional broadcasting pertaining to delay and redundancy. We then propose two novel directional broadcast protocols which aim to minimize the broadcast delay and also serve to mitigate redundant broadcast packets in the network.
Chapter 4

4. Broadcasting over Directional Antennas

4.1 Introduction

Network-wide broadcasting is a widely employed mechanism in mobile ad hoc networks (MANETs) wherein a node tries to communicate to all other nodes in a network. On demand routing protocols such as Dynamic Source Routing (DSR) [Johnson96] and Ad Hoc on Demand Distance Vector (AODV) [Perkins99], broadcast route request packets to establish routes to destinations. Others such as Zone Routing Protocol (ZRP) [Haas98] and Location Aided Routing (LAR) [Ko99] use variants of a network-wide broadcasting (i.e., flooding) to maintain and establish the routes. The dynamic nature of MANETs makes it all the more necessary for nodes to frequently communicate amongst themselves in order to maintain fresh routes. Clearly, an inefficient broadcast mechanism can cause a lot of redundancy and consequently consume a lot of scarce bandwidth.

In a MANET, the wireless channel is shared with all neighboring nodes. Omni directional antennas distribute energy in all directions other than just the intended direction. This leads to increased contention and generates unnecessary interference to other nodes and considerably reduces network capacity. On the other hand, with directional transmission both transmission
range and spatial reuse can be substantially enhanced by having nodes concentrate transmitted energy only towards their destination’s direction, thereby achieving higher signal to noise ratio.

In particular, the implementation of directional communication with switched single beam antenna is receiving large attention [Ramanathan01, Roy02, Ko00, and Nasipuri00]. With this antenna model, broadcasting is achieved by sequentially sweeping across all the pre-defined beams of the antenna system. Due to the increased range of a directional beam, a greater number of nodes are covered in a single broadcast sweep as compared to an omni directional broadcast. However, as pointed out in [Roy02], a broadcast by sweeping incurs a delay. Moreover, sweeping can lead to a greater number of redundant packets in the network as each forwarding node will transmit, in effect, M (the number of antenna beams) packets into the network.

Almost all routing protocols use flooding for broadcasting. Here, each node forwards individual broadcast packet it receives exactly once. Obviously, for N node network, there would be N-1 forwarding nodes and consequently N packet transmissions in all (assuming all nodes receive the broadcast packet). However many such rebroadcasts are not required and they collectively lead to the broadcast storm problem [Ni99]. Redundant rebroadcasts exaggerate contention and collisions in the shared medium. Consequently, several nodes may not receive the broadcast packet at all, leading to poor connectivity in the network.

Interestingly, with a switched beam antenna with M pre-defined beams, the number of broadcast packet transmission shoots up to M*N transmissions. For 100 node network each equipped with a 6 beam antenna, this translates into 600 broadcasts in the network. In contrast, with omni-directional antennas we would have had just 100 packet retransmissions. For any practical purpose, such a high number is clearly undesirable.
In this chapter, we focus on reducing redundancy in broadcasting in MANETs with directional antennas. However, at the same time we do not compromise on latency or connectivity. Our goal is to devise a broadcast protocol which can provide very high connectivity even in sparse node distributions, with reasonable latency and reduced redundancy. These new challenges lead us to propose methods that would reduce redundancy and minimize the sweeping delay in broadcasting over directional antennas. We compare our protocol against simple directional flooding, omni directional flooding and a fixed probabilistic approach.

The rest of the chapter is organized as follows. Section 4.2 discusses the IEEE 802.11 broadcast procedure. Simple flooding over omni-directional and directional antennas is discussed in section 4.3. Section 4.4 deals with related work of broadcasting in MANETs. In section 4.5 we point out the deficiencies in these schemes and propose two new schemes. We thoroughly compare the performance of the existing and new schemes over six, twelve and eighteen antenna beam models in section 4.6. In Section 4.7 we discuss the suitability of our protocol. Finally, we conclude and discuss future work in section 4.8.

4.2 Basic IEEE 802.11 Broadcast Protocol

IEEE 802.11 states that whenever a broadcast packet needs to be transmitted by a node, only the basic access procedure is supposed to be used. A broadcast is unreliable and hence to that effect, there is no exchange of control packets (RTS/CTS) or any receipt of an acknowledgment (ACK).

Under the DCF access method, a node may transmit if it detects the medium to be idle for greater than or equal to a DIFS period or an EIFS period if the immediately preceding medium-busy event was due to an incorrectly received packet. However, if under these conditions the
medium is sensed to be busy by the carrier sense mechanism, then the random back-off
algorithm is initiated. For complete details, the reader can refer to [IEEE97].

4.3 Simple Flooding

In this subsection we overview the mechanism of simple flooding in the context of omni and
directional antennas. This will serve as foundation to compare the effects of broadcasting over
these two antenna models.

4.3.1 Omni Directional Flooding (ODF)

The vast majority of ad hoc routing protocols use simple flooding for route discovery. Here,
each node forwards a broadcast packet exactly once. To facilitate this, the routing layer
maintains a table which caches the sequence number of the last broadcast packet received from a
source. Duplicate broadcast packets are discarded while fresh packets are handed down to the
MAC layer for retransmission. In order to avoid any global synchronization, every node waits for
a random delay before forwarding the packet. Transmission rules are governed by basic IEEE
802.11.

4.3.2 Simple Directional Flooding (SDF)

Similar to ODF, each node forwards a broadcast packet exactly once. However, in a switched
single beam model, a node transmits in only one direction at an instance. As we discussed
before, sweeping is needed across all antenna beams in order to cover all of a node’s one hop
neighbors. This sequential transmission in each sector incurs a sweeping delay along with a
greater redundancy. On the other hand, directional transmission increases the coverage area
which tantamount to more nodes getting covered in one complete sweep. Also, directional
transmissions improve spatial reuse by allowing multiple simultaneous transmissions in different
directions in a network. These counter-effecting characteristics of directional broadcasting pose new interesting challenges.

4.4 Related Work

Broadcasting in MANETs has been an extensively researched topic. The ultimate goal of all broadcast approaches is to reduce latency and redundancy while still maintaining a high connectivity. Intuitively, transmitting a large number of packets should guarantee a high connectivity. However, this approach wastes the precious network bandwidth and may even result in higher latencies. On the contrary, fewer rebroadcasts will lead to lower bandwidth wastage and maybe lesser latency too. However, it can result in poor connectivity of the network too since broadcasts are unreliable.

If the network topology is known apriori, then broadcasting can be reduced to a dominant set problem which is a classical NP-complete problem. However, assuming knowledge of the entire topology is an unrealistic assumption, particularly in case of MANETs where topology is changing dynamically.

A variety of literature is available on broadcasting in ad hoc networks. The broadcast storm problem in MANETs is discussed in [Ni99]. Williams et al. [Williams02] provide an excellent taxonomy and comparison of existing broadcast schemes for omni directional antennas. They classify broadcasting techniques into five groups: simple flooding, probabilistic, counter based, area based, and neighbor-knowledge based. As discussed above, simple flooding has a lot of redundant rebroadcasts. We have probabilistic and deterministic methods to reduce redundancy. However, the work in [Hu03] exploiting directional antennas for broadcasting is of particular interest here. We discuss two schemes proposed in [Hu03] for mitigating the broadcast storm
problem with directional antennas. We refer to them as Simple Enhanced Directional Flooding (SEDF) and Single Relay Broadcast (SRB).

4.4.1 Simple Enhanced Directional Flooding (SEDF)

In SEDF, whenever a node receives a packet requiring to be forwarded it starts a delay timer. If the same packet is received again before the expiration of this timer, the node makes a note of all the beams where that packet arrived at, and sets them to passive mode. Upon expiration of the delay timer, the node will forward the packet in only those beams/directions other than those in which the packet arrived (i.e., which have been marked as passive).

4.4.2 Single Relay Broadcast (SRB)

In any relay based scheme, whenever a node receives a broadcast packet it chooses a subset of its neighbors to forward the packet. Only members of this subset are allowed to forward the packet. It is the responsibility of the broadcasting node to explicitly designate the broadcast relay nodes within a broadcast packet header.

In the particular case of SRB, each node designates one and only one relay node in each direction on the basis of the received signal strength of hello packets in this particular direction. For the purpose of maintaining one-hop neighbor information, every node periodically transmits hello packets. The node whose hello packet is received with the weakest signal (likely to be the farthest) is selected to be the relay in that direction. Before forwarding, a node waits for a random delay and does not designate any relay node in directions where the packet arrives. It should be noted that a node may also discard a packet even if it has been designated as a relay, which happens in case it has already seen the packet before. Finally, we note that one-hop neighbor information is required in this scheme.
4.5 Proposed Schemes

The SEDF and SRB schemes have been evaluated in [Hu03] over a multiple beam antenna model which can simultaneously transmit packets in different beams. Over such models, these protocols do not suffer from a sweeping delay. However the implementation costs and power consumption of these complex multiple beam antenna systems are significantly higher than single beam antenna systems, rendering them as cost inefficient solutions for current ad hoc networks. Thus, we now propose two novel schemes for switched single beam antenna systems which attempt to reduce the sweeping delay and make use of the larger coverage range offered by our model. Our proposed enhancements require transmitting in fewer necessary beams instead of all the beams.

4.5.1 Protocol Design Considerations

All the schemes we introduce here make use of some common design considerations. In this section we discuss these mechanisms which enhance the proposed protocols performance while overcoming the limitations found in existing schemes.

4.5.1.1 One Hop Neighbor Awareness

We assume a periodic exchange of hello packets amongst the nodes. A node at any time is aware on which antenna beam its one hop neighbor lies. This exchange comes at no additional cost, since most directional MAC/Routing protocols need one-hop neighbor awareness to operate [Agrawal04].

To accomplish this, a node, say $S$, has to resort to a circular directional transmission of the broadcast packet through all its antenna sectors. It should be noted that while $S$ is engaged in this circular sweep, it remains deaf to any incoming packet. Hence, it has been showed in [Agrawal04] that a sender node $S$ needs to inform its neighbors the additional time they should
wait before they initiate a transmission towards it. To this end, the sender node $S$ includes in the broadcast packet the value $(K - c - 1)$ where $c$ is an integer (initially equal to zero) that keeps track of how many sectors the hello packet has been already sent. We define $K$ as the number of idle antenna beams at $S$. If $T$ is the time the receiver takes to completely receive the broadcast packet, the receiver, say node $R$, then waits for an additional time equals to $(K - c - 1)*T_{be}$ before initiating any transmission in the direction from which it received the packet from node $S$.

4.5.1.2 Novel Optimized Deferring while Sweeping

In both schemes, the IEEE 802.11 basic Carrier Sense Multiple Access (CSMA) is followed before transmitting in the first beam of a particular sweep. For subsequent beams of the same sweep, we simply carrier sense and transmit. However if a beam has been marked as busy (i.e., the Directional NAV [Takai02] is set in this direction), that beam is ignored and the next free beam is chosen. It should be noted that we do not wait for the beam to become free. Deferring in every beam would lead to an extremely high sweeping delay. We hope to considerably reduce sweeping delays with this optimization.

4.5.1.3 Random Delay Timer

Before forwarding a packet, a node starts a random delay timer (RDT) in order to avoid any global synchronization.

4.5.2 New Enhanced Directional Flooding (NEDF)

In NEDF, whenever the MAC layer receives a broadcast packet to be forwarded, it marks the beams where it will not retransmit as passive. First of all, similar to SEDF it will not rebroadcast in the antenna beams where it received the broadcast packet. Further, using the neighbor table,
NEDF will mark as passive those beams where there are no neighbors. Obviously, it will also not rebroadcast in beams which have been marked as busy (i.e., Directional NAV set).

Next, amongst the resulting selected beams the goal should be to transmit first in regions with maximum uncovered nodes and hence reduce the overall latency. Also, in regions where there is a high chance that nodes have already received the broadcast packet, re-broadcasting should be delayed. Keeping these objectives in mind, we define an order in which the transmission will be carried out. Initially, we select the beams which are vertically opposite to the beams where the node received the broadcast packet. Next, the beams which are adjacent to these vertically opposite beams are chosen. This will continue till all the selected beams are covered.

NEDF uses the received power of an incoming packet to decide the order of transmission in the vertically opposite beam. The beam vertically opposite to the one where the lowest power packet was received is selected first. Clearly, the idea here is that this packet came from the farthest node. Therefore, transmitting first in this beam guarantees maximum additional coverage, and if a uniform distribution of nodes is assumed, this translates into covering the maximum number of uncovered nodes. For example, in Figure 4.1 node (a) receives broadcast packets from nodes (b) and (c) through beams 5 and 4, respectively, before its RDT expires. It shall hence rebroadcast only in beams 1, 0, 2, 7, and 3 in that order. Here we note that beam 6 is ignored as it has no one hop neighbors in that direction. Finally, beam 1 is chosen over beam 0 as node (b) is farther from node (a) than node (c).

**4.5.3 Probabilistic Delay Broadcasting (PRB)**

An inherent flaw in the SRB scheme (discussed earlier) is that it uses a single relay node in each direction. This can lead to a partition in the network. For example, in Figure 4.1 node (h) is the obvious choice to serve as a relay for node (a) through its beam 7 (as it is the farthest).
Hence, node (i) shall not forward the broadcast it hears from node (a). As a consequence, if nodes (j), (k) and (l) are outside the radio range of node (h), they will never receive the broadcast packet thereby resulting in a partition. Further, it may also happen that the relay is engaged in a conversation with another node and hence be deaf to the broadcast packet. This too can lead to a network partition. Therefore, although a single relay node will translate into very few packet transmissions it may also result in higher latencies and poor connectivity.

Figure 4.1 NEDF

The PRB protocol aims at overcoming the drawbacks found in SRB. In PRB, each node is required to record the received power of the hello packet from the farthest node (weakest signal) in each beam. Let us denote this power as $P_f$. Upon receiving a broadcast packet and after the expiration of RDT, the node forwards the packet on all the beams except the ones on which it received the packet. For each beam, it includes $P_f$ of the corresponding beam in the packet header. Whenever a node receives this packet, it retrieves its received power, say $P_r$, and calculates the ratio of $P_f/P_r$. In PRB, this is the probability with which it will re-broadcast. In addition, the order of re-broadcasting will be similar to the one proposed in NEDF – vertically opposite beams followed by their adjacent beams. Similarly, neighbor-less and busy sectors will be ignored.
Therefore, in PRB nodes which are very close to the broadcast originator have very little probability to rebroadcast. In PRB there is still the option of eliminating the idea of very close nodes forwarding at all. With this option, in each sector only nodes which receive the packet at a power less than or equal to \(2\times P_f\) will retransmit with probability \(P_f/P_r\). Note that the farthest node in each sector has probability 1 to rebroadcast. Figure 4.2 illustrates this idea, where nodes (b) and (c) do not forward at all while nodes (d) and (e) forward with probability \(P_f/P_r\). Node (f) will definitely forward as its farthest node in that direction.

![Figure 4.2 PRB](image)

### 4.6 Performance Evaluation

We have implemented a directional antenna module in NS-2 (version 2.26) [ns226]. This module models most of the aspects of a directional antenna system including variable number of antenna beams, different gains for different number of antenna beams among others. As for the protocol support, we have implemented Directional Medium Access Control (DMAC) [Roy02a] protocol at the MAC layer. We compare the performance of ODF, SDF, SEDF, NEDF, SRB and PRB.

**4.6.1 Simulation Environment**

We assume an ad hoc network with 100 randomly distributed nodes. We vary the node density by varying the area from 4000m*4000m to 1000m*1000m. The range of an omni directional, 6, 12 and 18 beam antenna is 250, 450, 710 and 890 meters respectively [Agrawal04, Roy02a]. All
results are averaged over twenty different random topologies. During each run, one node is randomly picked as the broadcast initiator. In addition, for all the four different metrics considered we take three different set of results for 6, 12 and 18 antenna beams. We compare the performance of ODF and all directional flooding schemes under the following performance metrics:

- **Connectivity:** total number of nodes which received the broadcast packet, reflecting coverage;
- **Forwarding nodes:** total number of nodes which forward the broadcast packet, say \( F_n \) reflecting redundancy;
- **Transmitted packets:** total number of transmitted packets in the network, say \( T_p \) reflecting redundancy. Then the upper bound for \( T_p \) is \( m \times F_n \) where \( m \) is the number of antenna beams;
- **Latency:** time required for all the nodes to receive the broadcast packet, reflecting delay.

### 4.6.2 Simulation Results

Several interesting observations are in order from the graphs. Here we define sparse, medium and high node densities as 0-15 nodes/km\(^2\), 15-50 nodes/km\(^2\), and 50-100 nodes/km\(^2\) respectively. Figure 4.3, Figure 4.4, and Figure 4.5 show the variation of connectivity with node density for 6, 12 and 18 antenna models. We note that with 12 and 18 antenna beam models, the connectivity is almost 100% for all directional based schemes even in sparse density areas. In contrast, the omni directional scheme exhibits a higher connectivity for just dense regions. A direct interpretation of this is that directional antennas can overcome partitions in networks with their large coverage ranges. However, with a six antenna model, the performance of directional schemes is almost the same (in regions of medium and heavy density) and in some cases worse
(sparse density) than the omni directional scheme. The reason behind it is that in a six beam antenna model, the sweeping delay is not offset by the coverage gain. We also note that both NEDF and PRB consistently offer a better connectivity in comparison to SEDF and SRB.

Figure 4.3 Connectivity: 6 Antenna Beam
Figure 4.4 Connectivity: 12 Antenna Beam
Figure 4.5 Connectivity: 18 Antenna Beam
Figure 4.6 Forwarding Nodes: 6 Antenna

Figure 4.6, Figure 4.7, and Figure 4.8 and depicts the variation in the number of forwarding nodes with node density for 6, 12 and 18 antenna models. In regions of medium and heavy density, except SRB and PRB, all other schemes have nearly 99% nodes forwarding. As opposed to this, the number of forwarding nodes decreases with increase in density for both SRB and
PRB (except for 6 beam model where connectivity is low for sparse networks.). For maximum node density, as low as 15% and 36% nodes forward in case of SRB and PRB respectively as opposed to 99% for all remaining schemes. Next, we observe that ODF has surprisingly very low number of forwarding nodes for sparse areas. This is primarily because very few nodes get connected in this scenario and hence very few forward. Another observation is that, the numbers of forwarders for PRB is always greater than SRB. This is intuitive since there can be more than one forwarder in a beam in the case of PRB as opposed to SRB.

Figure 4.7 Forwarding Nodes: 12 Antenna
Figure 4.8 Forwarding Nodes: 18 Antenna
Figure 4.9 Latency: 6 Antenna Beam
Figure 4.10 Latency: 12 Antenna Beam
Figure 4.9, Figure 4.10, and Figure 4.11 measures the variation in 95%-latency with node density for 6, 12 and 18 antenna beams respectively. For 12 and 18 beam models, NEDF with its high coverage range and intelligent order of sweep exhibits the least latency. ODF and SRB display highest latencies, ODF because of its limited coverage range and SRB due to its minimal number of forwarders. PRB exhibits an average performance since it neither relies on just a single relay in each direction nor does it let all the nodes forward.
Lastly Figure 4.12, Figure 4.13, and Figure 4.14 measure the number of transmitted packets with changing node density for 6, 12 and 18 antenna beams respectively. In regions of low density, for a 6 beam model, we observe that very few packets get transmitted, but this is because of the fact that very few nodes actually receive the broadcast packet. For 12 and 18 beam models we observe that the number of transmitted packets decreases with increase in node density. However, SDF and ODF transmit a constant number of packets, SDF the maximum and ODF the minimum. This is due to the fact that SDF sweeps across all antenna beams, whereas ODF transmits a single packet for each forwarder. Also, intuitively the number of transmitted packets increases with increase in the number of antenna beams. Amongst the existing and proposed schemes SRB and SEDF provide the best and worst performances. NEDF fares better than SEDF with PRB again exhibiting average performance.

4.7 Discussion on Standard Scheme

From the simulation results we observe that there is a definite tradeoff in latency and redundancy. NEDF gives excellent results for latency while SRB fares well in controlling redundancy. We also notice that directional broadcasting schemes fare much better than the omni directional scheme in scenarios where nodes are equipped with 12 and 18 beam antennas. In such scenarios, the coverage gain beats the sweeping delay incurred over these models. This can be attributed to the optimized deferring employed to lower contention in case of directional antennas. For 12 and 18 beam antenna, all directional schemes exhibit almost 100% connectivity. Amongst these schemes, we find that PRB gives average results for both redundancy and latency. It has been shown that SRB fares badly in regions of high mobility [Hu03]. Though we do not consider mobility in our simulations, our conjecture is that PRB should fare better than SRB even in highly mobile scenarios. This should be so since we do not rely on just a single
node to forward in a direction. Moreover, we employ a more sophisticated neighbor discovery mechanism. The same explanation can be extended to assert that PRB should fare better in regions of high network traffic. On this basis, we propose PRB to be used as a standard broadcasting protocol for route discoveries and maintenance. In addition, for one-hop broadcasting where the broadcaster expects a direct reply from one of its neighbors, we propose the use of the one-hop hello broadcast protocol explained in section 4.5.

4.8 Conclusion

In this chapter we have indicated the shortcomings of existing schemes and proposed two novel approaches (NEDF and PRB) for exploiting broadcasting over directional antennas. NEDF provides best results for latencies and fares better than SEDF for reducing latencies. PRB gives the best average case performance and fares better than ODF, SDF and SRB in terms of latency. It is much more effective in reducing redundancy than SDF, SEDF and NEDF. As opposed to omni directional broadcasting, directional schemes are highly effective in sparse networks as they offer great connectivity and low latencies. However, there are greater redundancies too.

In the following chapter we shall discuss the routing issues associated with using a directional antenna system. We will then propose our cross-layered routing protocol, DRP. The insights obtained from this chapter shall be used while developing the route discovery module DRP.
Chapter 5

5. Routing over Directional Antennas

5.1 Introduction

The use of directional antennas in MANETs creates new types of hidden terminal problems and node deafness [Agrawal04, Ramanathan01, and Roy02a]. In addition, fundamental issues such as determination of a node’s neighbors have to be properly handled. Deafness is defined as the phenomenon when a node $X$ is unable to communicate with node $Y$, as $Y$ is presently beamformed in a different direction. In such an event, $X$ perceives $Y$ to have moved out of its range, thereby signaling its routing layer to take actions, hence affecting the network throughput. Deafness and hidden terminal issues have been extensively studied in a one hop scenario but no solution has been provided.

Although there is plethora of literature towards designing efficient directional MAC schemes, a complete design of a routing protocol tuned to the underlying directional environment still requires to be explored. Here we argue that a routing protocol specifically tuned to the underlying MAC layer can reap interesting performance benefits. For example, in directional routing, a source node can exploit the antenna beam information towards its destination for an
efficient route recovery. In this chapter we propose a Directional Routing Protocol (DRP)\(^1\) for MANETs. The salient contributions of this work can be summarized as follows:

I. A detailed study of issues related to routing in directional antenna systems. We also outline the behavior of different IEEE 802.11 system parameters in a multi-hop directional environment.

II. Designing a directional routing protocol called DRP. The main features of DRP include an efficient route discovery mechanism, establishment and maintenance of directional routing and directional neighbor tables (DRT and DNT respectively) and a novel directional route recovery mechanism among others.

III. Comprehensive simulation comparison of DRP with DSR over both omni-directional and directional antenna models. Our simulation compares both static and mobile scenarios.

The rest of this chapter is organized as follows. In Section 5.2, the major issues involved in the design of a directional routing protocol in MANETs are discussed. In Section 5.3 we discuss the existing works in the area of directional routing protocols. Dynamic Source Routing (DSR) is discussed in section 5.4. Section 5.5 outlines and discusses the key features of DRP. Our comprehensive simulation results comparing DRP with omni and directional DSR are presented in Section 5.6. Finally the chapter concludes in Section 5.7 highlighting some open problems and future research directions.

5.2 Directional Routing Issues

In this section, we will investigate different issues related to directional communication and their impact on directional routing. For the discussion below we assume an omni-directional...
DSR protocol running over a single switched beam directional antenna system. This will serve as a foundation for developing our DRP protocol.

5.2.1 Directional Broadcasting Overhead

As discussed in the previous chapter, in a single switched beam directional antenna systems, sweeping is needed across all antenna beams in order to cover a node’s one hop neighbors. Each forwarding node, in effect, transmits M (the number of antenna beams) packets into the network. For a single switched beam antenna system, this adds to both packet redundancy and delay. Since the Route Request (RREQ) packets are flooded throughout the network, an inefficient broadcasting strategy may negatively impact the quality of routes [Roy02] source node gets.

Hence, a careful route discovery is necessary to obtain optimal routes with minimal Route Discovery Latency (RDL) and redundancy. In DRP, we employ a novel directional broadcasting strategy (described in Section 5.5.3) aimed at reducing the broadcast redundancy and RDL. This broadcast protocol is inspired by NEDF, explained in chapter 4.

5.2.2 Address Resolution Protocol (ARP) in Directional Environment

Once a destination node receives a RREQ packet, it unicasts a route reply (RREP) back to the source\(^2\). However, before forwarding the RREP packet, the destination needs to do an Address Resolution Protocol Query (ARP-Query) to obtain the MAC address of its previous hop (this is also true for all intermediate nodes forwarding the RREP packet). Since the ARP-Query is a broadcast packet, nodes will do a sweeping to locate its previous hop. However, sending ARP-Query through sweeping has some potential issues in directional environment. To illustrate please refer to Figure 5.1, wherein node D is the final destination of a RREQ packet, and it needs to send the RREP back to the source through an intermediate node C.

\(^2\) Assuming a bi-directional link between the source and destination
As shown in Figure 5.1(a), after receiving the \textit{RREQ} (step 1), and before sending its \textit{RREP}, node D is required to sweep a ARP-Query to obtain the MAC address of node C. If at the instant when node D broadcasts its ARP-Query (step 2) towards node C, node C is beam-formed in other direction to sweep its \textit{RREQ} packet (\textit{RREQ} is a broadcast packet), the ARP-Query at node D will fail. A similar situation can also arise for ARP-Query and ARP-Response as shown in Figure 5.1(b), wherein node D can miss ARP-Response from C because of being beam formed in other direction to transmit ARP-Query.

It is to be noted that node D remains deaf for the entire duration of ARP-Query sweeping. Hence it may miss a \textit{RREQ} packet with a better route, received from some alternate path. This issue has been studied in [Roy02]. In order to increase the chances of obtaining optimal routes, we extend the idea of delayed route reply as proposed in [Roy02]. In DRP destinations delay sending the \textit{RREP} by a time duration $T$, calculated from the time it received the first \textit{RREQ}. This additional time will allow the destination node to pick the best amongst all routes which arrive within the time duration $T$. We take $T$ as the time required for a complete sweep.
5.2.3 Directional Antenna Beam Handoff

Another important issue to address in designing a directional routing protocol is the movement of a next-hop within the transmission range of a node. In a single switched beam directional antenna system (where a transmitter has no way to continuously track the location of its next-hop) special care needs to be taken for the case when a next-hop moves, and is reachable through a different antenna beam (direction) from the sender.

Some MAC layer approaches [Nasipuri00, Roy02] first try to reach the destination in its previous beam location, and after \( m \) such RTS failures \((m < 7)\) (e.g., for example, in ([Nasipuri00, Roy02], \( m \) is 5), the remaining RTS are sent to the all antenna beams (referred to as scanning).

However, in DRP we claim that relationship exists between the beam-width and the number of adjacent beams which should be scanned. For example, as shown in Figure 5.2(a), even if node \( B \) is a neighbor of \( A \) (after movement from beam 0 to beam 4), the link from \( B \) to \( C \) is already broken. In DRP, we take this into consideration and employ an efficient and novel algorithm for scanning adjacent beams as described in Section 5.5.5.

![Diagram](image.png)

Figure 5.2 Effect of Node Movement and Multi-hop Packet Forwarding
5.2.4 Deafness and Node Movement

Effects of deafness on route discovery are well understood in the literature [Roy02]. However, the relationship between deafness and node movement has not been adequately investigated. For example, even if the next-hop (receiver) has not moved out of the transmission range of the sender, an absence of CTS from the receiver due to deafness may be perceived as a link failure by the sender. This may ultimately result in a route error leading to a new route discovery. MAC layer of DRP has some provisions to distinguish deafness from node movement. We will revisit this issue in Section 5.5.2 and outline how our MAC differentiates between deafness and node movement.

5.2.5 Directional Neighbor Discovery

A node at any time needs to be aware on which antenna beam its one hop neighbors lie. This information is necessary for the sender’s MAC to resolve the beam in which it must initiate communication. A periodic exchange of hello packets amongst all the nodes is the most trivial solution to this problem. However, as discussed before, a directional broadcast (sweeping) amounts to a lot of control overhead. In addition to the large number of packet transmissions, the broadcasting node becomes deaf during the entire sweeping duration. To eliminate these, we propose a novel method of both establishing and maintaining a directional neighbor table. In DRP, a directional neighbor table (DNT) is established during the route discovery phase. To this end, DRP uses routing control packets such as RREQ and RREP to facilitate neighbor resolution. Any neighboring node Y which receives this packet on its beam “i”, can easily comprehend that it needs to send a packet on its “i-th” beam in order to communicate with X. Similarly, with the RREP packet, node X can also resolve node Y to one of its antenna beams. In addition, similar to [Nasirpuri00a] DRP employs MAC layer snooping for maintaining the DNT. By setting the
MAC in a promiscuous mode, a node overhears all the traffic in its vicinity. By simply overhearing these packets, a node updates its DNT as well as track new neighbors in its surroundings.

5.2.6 Anomaly of Directional Multi-hop Packet Forwarding

It has been shown that, directional communication suffers from an increased instance of hidden terminal problem in a linear scenario [Roy02a]. Unfortunately the route obtained for a specific destination, is more likely to have some linear path. For example, in Figure 5.2(b), let us assume that a part of the route to a destination follows a linear path [X, Y, Z]. Let us now analyze the effects of linearity of path on directional packet forwarding. Node X after a successful DATA transmission to Y will begin its post-backoff before its next transmission attempt. In the meantime, Y will attempt to forward the packet to Z. While Y is sending its RTS to Z, if the post-back-off of X expires and it has another packet for Y, it will initiate RTS towards Y. Several unwanted side effects may result because of this. First it may result into a packet collision at Z, and force Y to try again. Secondly, at X, the RTS will fail (deafness at Y) and it will increment its STA Short Retry Count (SSRC) 0.

In essence, the post-backoff period is designed for omni environment, where a node can listen about the transmission in its neighborhood. On the other hand, in a directional environment the post backoff period may not be sufficient for a node (Y in this case) to complete sweeping and inform all its neighbors about its transmission. It is to be noted that the presence of hidden terminal problem is more severe for the case when the nodes do not employ a circular sweep of RTS, which increases the chances of DATA packet collision because of their longer duration.

---

3 After successfully transmitting a DATA packet, each node in IEEE 802.11 starts a random backoff. This is termed as post-backoff which allows other nodes to access the channel.

4 Some directional MAC protocols (e.g. DMAC [7]) do not employ a circular transmission of RTS/CTS packet. Hence they suffer from increased instance of hidden terminal problem in multihop environment.
One way to handle this issue is to increase the post-backoff period for directional antenna systems so that its at least equal to the duration of one complete antenna sweep. In this manner \( Y \) can complete its circular sweep of RTS and inform \( X \) about its transmission to \( Z \). However pause and resume functions involved with the backoff mechanism may force the node to wait for long and hence can negatively affect channel sharing. An alternative way to handle this issue would be to introduce the concept of post-defer after each successful DATA transmission. However, designing an optimal post-backoff or post-defer period in directional environment is an interesting research topic and can be a future work of this dissertation.

5.3 Related Work

Nasipuri et al. [Nasirpuri00a] propose a scheme to estimate the direction of the destination relative to the source in order to confine the spread of route discovery packets using directional transmission. However, the strategy is more effective for route rediscovery at the source. In addition, the flooding overheads due to directional sweeping have not been addressed. In DRP we propose a novel mechanism for the direction estimation based on the route to the source. This mechanism is used both at source and intermediate nodes to provide a more robust mechanism to handle route failures.

To reduce the MAC layer contention among different flows, in [Roy03], notion of exploiting maximally zone disjoint routes has been introduced. Directional communication is shown to be effective in both route discovery and use of such routes. But, the proposed protocols require all nodes to be completely aware of the topology and ongoing neighborhood communications. A MAC and a proactive routing protocol over ESPAR antennas have been suggested in [Bandy01]. This is a complex MAC incurring considerable control overhead. In [Saha02], the authors illustrate the effectiveness of directional antennas by overcoming partitions introduced in the
network due to the mobility of nodes. They advocate the use of larger ranges of directional antennas only in the event of a link failure. In [Roy02], authors evaluate the performance of DSR over DMAC and omni directional antennas. Several issues ranging from directional route discovery to mobility management are explored in the context of directional communication, which is shown to be more effective when topologies are sparse and random. However, DMAC is susceptible to deafness and hidden node problems [Roy02a] which limits multi-hop routing performance in many scenarios.

In this chapter, we attempt to provide a complete routing solution over single switched beam antennas so that both the spatial reuse and larger ranges offered by this model can be exploited. Our simulation results validate the efficacy of our proposal.

### 5.4 Dynamic Source Routing (DSR) Protocol

DSR is a source routed reactive routing protocol [Johnson96]. The protocol consists of two major phases: route discovery and route maintenance. When a mobile node has a packet to send for some destination, it first consults its route cache to determine whether it already has a route to the destination. If no such route exists, it initiates a route discovery by broadcasting a Route Request (RREQ) packet. This route request contains the address of the destination, along with the source node’s address and a unique identification number. All nodes, except the destination node, rebroadcast this packet exactly once. While doing so, they append their own address to the route record field in the RREQ packet. Hence, the path followed by this route request gets included in the RREQ packet.

On receiving the RREQ packet, the destination node responds by sending a route reply (RREP) message to the source. Since a bi-directional link is assumed between all the nodes, the route reply is a unicast message, following a path obtained by reversing the route followed by the route
request. The route received at the source is cached for subsequent communication. DSR facilitates route maintenance through the use of Route Error (RERR) packets. In the event of a link failure at an intermediate node on a source route, a RERR packet is generated and sent to the source. All intermediate nodes including the source node remove all the routes in their caches which have that broken link. DSR also has provisions for nodes to learn and cache routes by overhearing RREPs or data packets containing source routes. Such cached routes can reduce flooding overheads in the network and reduce route discovery latencies.

5.5 Proposed Protocol

5.5.1 Directional Routing Protocol (DRP)

DRP is an on-demand directional routing protocol, and is inspired in large by omni-directional Dynamic Source Routing (DSR) [Johnson96] protocol used heavily in MANETs. DRP closely couples the routing layer with the MAC layer and assumes a cross-layer interaction between some of the modules. A pictorial view of the protocol stack of a DRP aware mobile node is shown in Figure 5.3(a). In DRP the Directional Routing Table (DRT) is local to routing layer and maintains the routing information to different destination. The Directional Neighbor Table (DNT) on the other hand is shared with MAC.

Unlike DSR which maintains only the index of the node ID in a forwarding path; DRP also maintains node indices and the beam IDs used by the nodes to receive a packet in the forwarding path. The beam ID stored in the DRT helps the source node to estimate the angular position of its destination relative to itself. Although a similar scheme of maintaining beam IDs has been suggested in 0, DRP uses the beam ID kept in the DRT to do an efficient route recovery as described in Section 5.5.3.
Figure 5.3(b) illustrates the basic working principle of DRP. In this figure, suppose node $A$ has a packet for node $E$. After a successful route discovery, suppose node $A$ finds a route to $E$ as \{B(2), D(3), E(3)\}. This implies that the route taken by a packet from $A$ to $E$, follows the path $B$, $D$, and $E$, and the antenna beams used by $B$, $D$, and $E$ to receive a packet from uplink is 2, 3, and 3 respectively. This information is stored in DRT of the source node $A$. Similarly the content of DNT at $A$ is as shown in Figure 5.3(b). This information in DNT is used during the sweeping of RTS-CTS.

In addition to the shared DNT, in DRP the network layer is aware of the different antenna beams at the MAC layer. The MAC, in turn, has separate buffers for each antenna beams. Accordingly, the link layer follows this approach by maintaining separate queues for each beam as shown in Figure 5.3(a). In order to place the packet in the correct link layer queue, the network layer determines the antenna beam which the MAC will use for transmission of the packet (through DNT), and puts the packet in the link layer queue corresponding to this antenna beam. It is to be noted that broadcast packets are kept in a separate dedicated queue.
5.5.2 **DRP Medium Access Control**

DRP is built on top of MAC protocol for Directional Antenna (MDA) [Agrawal04]. To minimize the effect of deafness and hidden node problems in directional environment, MAC layer of DRP (termed as MDA) employs a special form of sweeping of both RTS and CTS, namely, the Diametrically Opposite Directions (DOD) procedure. The DOD mechanism includes two major enhancements over sweeping, firstly, RTS and CTS packets are transmitted in DOD which ensures maximum coverage; secondly, these packets are only transmitted through the antenna beams with neighbors. In addition, the Enhanced Directional Network Allocation Vector (EDNAV) mechanism incorporated in MDA considerably improves performance by accurately differentiating between deafness and collision scenarios. EDNAV mechanism is an extension of Directional NAV (DNAV) scheme outlined in [Takai02] is a simple extension of IEEE 802.11 NAV concept and is employed to handle issues of hidden terminal problem in directional environment. Essentially, DNAV is a table that keeps track for each direction the time during which a node must not initiate a transmission through this direction. For a complete description of MDA please refer to [Agrawal04].

5.5.3 **DRP Route Discovery**

The route discovery mechanism in DRP works similar to DSR. For a given source $X$ and destination $Y$, if $Y$ is not in the DNT of $X$, $X$ floods a $RREQ$ packet in the network. DRP enforces a broadcast optimizations proposed in the chapter 4 to reduce packet redundancy and route discovery latency. Whenever a node receives a $RREQ$ packet it starts a delay timer. If the same $RREQ$ packet is received again before the expiration of this timer, the node makes a note of all the beams where that packet arrived from. The node forwards (or sweeps) the packet in only those beams/directions other than those in which the packet arrived. Amongst the selected
beams, DRP initiates a rebroadcast in the beams which are vertically opposite to the beams where the node received the broadcast packet. Next, the beams which are adjacent to these vertically opposite beams are chosen. This shall continue till all the selected beams are covered.

The IEEE 802.11 basic Carrier Sense Multiple Access (CSMA) is followed before transmitting in the first beam of a particular sweep. For subsequent beams of the same sweep, we simply carrier sense and transmit. However if a beam has been marked as busy (i.e., the Directional NAV is set in this direction), that beam is ignored and the next free beam amongst the selected beams is chosen. It should be noted that we do not wait for the beam to become free. Deferring in every beam would lead to an extremely high sweeping delay. Further, we do not post-backoff after successfully transmitting in a beam and before initiating transmission in another beam. A node post-back offs after completing a sweep of all the selected beams.

The sequence of hops taken by the route request packet as it propagates through the network during the route discovery phase is recorded in a data structure in the packet. It is termed as directional route record. The directional route record appends both the node indices of the intermediate nodes and the beam ID used by these nodes to receive the packets from uplink. For example, an intermediate node Z which forwards the route request packet, also adds the antenna beam at which it received the RREQ packet in addition to its own ID.

5.5.4 DRP Route Maintenance

The function of the route maintenance module is to monitor the operation of a route to a destination and inform the sender of any intermediate link failures or routing errors. In DRP, routes are generally associated with the antenna beam to be used to reach a particular next-hop. Hence any change in the location of the next-hop even within the transmission range, needs to be handled carefully.
Similar to DSR, in DRP when originating or forwarding a packet using a source route, each node transmitting the packet is responsible for confirming that data can flow over the link. A link layer acknowledgment as in IEEE 802.11 is used for this purpose. Typically, in IEEE 802.11 a sender sends an RTS to the receiver and waits for a CTS packet before it begins the DATA transmission. It retransmits the RTS packet again if it doesn’t receive a CTS packet within a stipulated time. The standard specifies seven retries, before the link layer informs the routing layer of a link failure. In DSR, the routing layer proceeds by generating a route error packet back to the original source so that all intermediate nodes can remove routes which contain the contaminated link. When the source needs to route another packet for the same destination, it initiates a fresh route discovery. In a scenario where node mobility is high, link breakages are frequent. The situation gets worse in the case of directional antennas where the direction (or beam) of the next-hop needs to be known before-hand. If for every such link failure, a route discovery is initiated, there would be considerable delays and routing overheads. DRP’s route maintenance phase attempts to avoid such recurring route discoveries at the source.

Upon sensing a link failure, DRP follows a three phase route recovery mechanism:

- **Location tracking:** The next-hop may have moved so that it may be reachable through some other antenna beam as opposed to the one suggested by the DNT. It therefore scans the adjacent beams depending on its beam-width.

- **Local Recovery Mechanism:** If the antenna beam handoff fails, DRP proceeds to discover a route for the next-to-next hop. It estimates the direction of this next-to-next hop based on the beam information and confines a route discovery in this region.

- **Zonal Recovery:** Now if the local recovery also fails, a route error packet is generated and sent back to the source. When another packet for the same destination arrives, the source first
tries to discover the route for the destination by confining the route query packet in a specified zone. The source perceives the destination to be within this zone. If no route reply is received, it resorts to a route discovery over the entire network.

5.5.5 Route Recovery Mechanisms

Location tracking and two-hop directional local recovery phase are local to the node which detects a link breakage. On the other hand, the Zonal Recovery Phase is at the source. We now provide a comprehensive description of each of these phases and how they inter-work.

5.5.5.1 Location Tracking Mechanism

Due to the continuous movement of the nodes, the antenna beam used by a node to reach its next-hop and vice versa may change. Several methods have been proposed in literature to track such movements. The approach in [Horneffer96] uses two antenna beams to continuously locate the position of a mobile node within the transmission range. This approach requires special hardware support which makes the cost of the overall system very expensive. The scheme discussed in [Yum92] uses the concept of tones and extensive network state information at each node to track the position of a mobile node. On the other hand, [Nasipuri00] uses the concept of circular directional transmission of both RTS and CTS packets which eliminates the need of any specific tracking mechanism within the transmission range of a node.

DRP employs a location tracking mechanism which scans a subset of antenna beams at the sender. Suppose node X is presently forwarding a packet to node Y. If the transmission of an RTS from X to Y’s previous location (say antenna beam “i”) fails for 3 consecutive attempts, node X tries to locate Y in its adjacent antenna beams for the remaining tries. Hence the 4th, 5th, 6th and 7th RTS is sent in a subset of adjacent antenna beams, including “i”. Clearly, the subset of beams which need to be scanned depends on the antenna beam-width. The reason behind
scanning adjacent antenna beams is obvious. If a node is not reachable through adjacent antenna beams, the validity of the old directional path to the same destination becomes questionable. Figure 5.4 (a) helps us to understand this point. Suppose an intermediate node Y has moved from X’s antenna beam 0 to antenna beam 4. For all practical purposes, the link between Y to Z is now broken. Hence scanning in beam 4 for node Y will be futile as a link breakage will then occur at the next hop.

Figure 5.4 (b) helps us to understand the antenna beams of X at which node Y can move, and may still have connectivity to Z. In addition it also helps us to estimate the number of antenna beams where an RTS packet should be sent to locate Y. The shaded region is the portion where Y can move and can still maintain a link with both nodes X and Z. If d is the separation between nodes X and Z and r is the communication radius, then the angular region where node X must scan for node Y is: \(2\cos^{-1}\left(\frac{d}{2r}\right)\). For \(d=1000\) and for an eight beam model, this angle is approximately equal to 67 degrees. This means that a node needs to just scan in the \((i-1)^{th}\) and \((i+1)^{th}\) beams. However, for a twelve beam model, this angle works to around 90 degrees. Hence a node needs to scan in the \((i-1)^{th}\), \((i-2)^{th}\), \((i+1)^{th}\), and \((i+2)^{th}\) beams. This approach of locating Y to only a subset of antenna beams is different than some of the existing schemes which either recommend searching in all the beams or simply scanning the adjacent beams [Roy02a]. Table 5.1 lists the adjacent beams to be ideally scanned with increasing beam width.

A node trying to scan for its next hop in its adjacent antenna beams transmits multiple RTS packets (a partial sweep). Therefore, the receiver node needs to be informed the additional time it must wait before initiating CTS towards the transmitting node. The failure to do so will either lead to a collision or a packet loss due to deafness. Consider Figure 5.5. Suppose node Y has moved and is now reachable through beam 1 of node X. Node X, equipped with an eight beam
antenna decides to scan for node Y in beams 1, 0 and 7. Say the order of this partial sweep is also beams 1, 0 and 7. Now when X transmits an RTS in beam 1, Y will receive it and immediately reply with a CTS packet. In the meantime X will be beam-formed towards beam 0, trying to complete its partial scan. It will, therefore, miss out on this CTS (deafness). Hence for a proper RTS-CTS handshake, RTS packets also include a variable “K” which is the remaining number of antenna beams through which the RTS packets have to be sent at node X. “K” helps in synchronization of RTS-CTS and the next-hop (node Y) sends its CTS after a duration of K*RTS_Transmission_Time. As mentioned earlier, we do not defer in each beam while sweeping. This synchronization is necessary to avoid loss of CTS due to RTS sweeping.

![Diagram](image)

(a) – Node movement within antenna beams.  
(b) – Region of intermediate node movement where the route is still maintained.

Figure 5.4 Intermediate Node Movement

Now if all the RTS retries fail, a broken link is reported to higher layer. This ends the location tracking phase in a node.
Table 5.1 Antenna Beams to be scanned for local route recovery

<table>
<thead>
<tr>
<th>Beam Width</th>
<th>Antenna Beams to be scanned about beam “i”</th>
</tr>
</thead>
<tbody>
<tr>
<td>90</td>
<td>“i-1”, “i”, “i+1”</td>
</tr>
<tr>
<td>45</td>
<td>“i-1”, “i”, “i+1”</td>
</tr>
<tr>
<td>30</td>
<td>“i-2”, “i-1”, “i”, “i+1”, “i+2”</td>
</tr>
<tr>
<td>20</td>
<td>“i-3”, “i-2”, “i-1”, “i”, “i+1”, “i+2”, “i+3”</td>
</tr>
</tbody>
</table>

Figure 5.5 Movement of node Y from beam 0 to 7

5.5.5.2 Local Route Recovery Mechanism

After the location tracking mechanism has failed, node X proceeds in the following manner:

i. It identifies the second next-hop in its path through the DRP packet header as a temporary destination (Z in this case). If the link breakage occurred at the last hop, the final destination will be the temporary destination.

ii. Node X uses the direction estimation algorithm (explained later) to determine the approximate direction of the temporary destination relative to itself. It then selects a set of beams which will cover this direction. We refer to this set as the damp beam set.

iii. Node X generates a directional RREQ packet to find the route to the temporary destination with maximum propagation limit set to 3.

iv. This RREQ packet is transmitted only towards the direction (damp beam set) of the temporary destination, and intermediate nodes which receive this packet are also supposed to forward it in the damp beam set.

v. After transmitting a directional RREQ, node X starts a timer for duration ‘β’ within which it expects a RREP. Node X maintains a local recovery table which records the temporary destination and the RREQ timeout value, ‘β’ in this case. The table also maintains a pointer to a
buffer where packets following the original route are buffered till a reply is received. This step is required to avoid un-necessary RTS retries by node X for node Y. Prior to forwarding a packet, DRP checks whether a local recovery is underway for the route in the packet. If it detects so (using the local recovery table), it buffers this packet and does not relay it to the MAC layer.

vi. If a reply is received within the duration ‘β’, node X stops its RREQ timer. At this point, two tasks are required: (i) use the route reply received to create another valid route to the destination and forward all the packets in the local recovery buffer using the new route and (ii) inform the source node about the new route. For the latter, Route Error (RERR) packets are used. The information about the contamination of a link is also included in these packets. We use a LOC_RERY flag in the RERR packet to indicate the success of the two-hop directional route recovery. If the LOC_RERY flag is set, the RERR packet includes the updated route. The source node after receiving a RERR packet with LOC_RERY flag set, updates its route to the corresponding destination. If node X fails to receive a route reply during ‘β’, it sends a RERR packet towards the source node with the LOC_RERY flag set to false.

Figure 5.6 illustrates the route before and after the two hop directional route recovery procedure. Node X initiates a local recovery and once it receives a reply from node Z, it must ensure that the new route is free of any loops.
Figure 5.6 Local Route Recovery Mechanism

Figure 5.7 Timer Interplay at Node X

Figure 5.7 explains the timing sequences at the node X. At instant $t(a)$, node X sends the route query for node Z. The interval between $t(a)$ and $t(c)$ corresponds to $\beta$. Let us assume that node X receives a reply from node Z at instant $t(b)$. Now while the route error packet (with the new route) is propagating back to the source, new packets from source S may arrive at node X. Node X needs to take special care for these packets. In DRP, once the local route reply is received, a timeout value (the interval between $t(b)$ and $t(d)$) is associated with the local reply. Till that timeout period, any packet which arrives at X (with final destination as D and next hop as Y), is modified so that it follows the new route. The failure to do so will result in another route query at X, when in reality a new route has already been discovered. The value of $\beta$ is approximated to:

$$6 \times (\text{Transmission\_Delay} + \text{Propogation\_Delay}) + \text{Time\_For\_A\_Complete\_Sweep} + \text{Random\_Value}.$$
5.5.5.3 Zonal Route Recovery Mechanism

After receiving a RERR packet with \textit{LOC\_RERY} flag set to false, the source node S first tries a zonal route discovery to find a route to its destination. The concept of a zonal route repair is to limit the zone in which the route request packet is propagated. For this purpose, the source first estimates the location of the destination node relative to itself using DRP’s \textit{direction estimation algorithm}. Consider Figure 5.2. Let us assume that node A requires to rediscover a route to E, and the previous route to E maintained at its route cache was \{B(2), D(3), E(3). Assuming all the nodes to be equipped with four beam antennas, A begins by approximating the relative position of B. Since B receives a packet from A in its antenna beam 2, by symmetry B will lie in the antenna beam 4 of A. If the average separation between the nodes is \(R\) (which is assumed to be half of the nodes transmission range), then B is assumed to lie at distance \(R\) on the angular bisector of antenna beam 4 of A. Hence the co-ordinates of B relative to A are \((R\cos x, R\sin x)\).

Next the co-ordinates of D and then E are estimated. Finally, A calculates the angular position of E relative to itself. A will then pad this angle by 45 degrees on either side. It will send the RREQ packets in only those beams which lie within this angle. Hence, in the aforementioned example, A shall send route request packet in beams 1 and 4. All the nodes receiving this route request packet are supposed to forward the route request in antenna beams 1 and 4 only. This limits the transmission zone of RREQ packets.

In case this zonal route repair packet fails, the source restarts a route discovery, flooding the route request throughout the network.
Figure 5.8 DRP Algorithm
5.6 Performance Evaluation

5.6.1 Simulation Model

We have implemented a complete directional antenna module in ns-2 (version 2.26) [ns226]. This module models most of the aspects of a directional antenna system including variable number of antenna beams, different gains for different number of antenna beams among others. The transmission range of an omni directional, 4, 8 and 12 beam antenna is assumed as 250, 370, 550, and 710 meters respectively [Roy02]. We have implemented MDA (a directional medium access control) and DRP (directional routing protocol with MDA as the MAC layer). We assume CBR traffic and a 2Mbps channel for all our scenarios. Simulation is run for 200secs and all results are averaged over 10 different seeds. We compare the performance of DRP with DSR over omni-directional antennas (referred simply as OMNI or DSR) and DSR over directional antennas (referred to as DDSR). It should be noted that DDSR also uses MDA as its underlying MAC. This is essential for a fair comparison which offsets any MAC layer benefits. We mention DRP and DDSR over an M beam model as DRP_M and DDSR_M respectively. DDSR also uses MDA at the MAC layer. In the following subsections we thoroughly evaluate all the modules of DRP under static and mobile scenarios.

5.6.2 Route Discovery Latency

We assume a square grid of 64 nodes with inter-node spacing as 250 meters. We have implemented the delayed route reply optimization for both DDSR and OMNI so that the destination always responds by a single most optimal route. Figure 5.9 depicts the behavior of route discovery latency with varying source-destination separation. An eight beam model is used for both DSR and DDSR. DRP with its novel broadcasting approach is seen to consistently outperform both DSR and DDSR. The differences are more profound as the source-destination
separation increases. Moreover, both DRP and DDSR outperform DSR as the range of an 8 beam antenna is higher in comparison to an omni antenna.

![Figure 5.9 Route discovery latency](image1.png)

![Figure 5.10 Redundancy due to directional broadcast](image2.png)

![Figure 5.11 Hop length with inter-node separation](image3.png)

### 5.6.3 Routing Control Packet Redundancy

In this section we evaluate the broadcast redundancy of DRP as compared to DDSR which does not use any kind of broadcast optimization. For a fixed source-destination separation, we measure the average number of redundant route query packets received at each node during a route discovery. Figure 5.10 depicts the average number of route query packets at each node for DRP and DDSR against increasing beam-width. A beam-width of 360 corresponds to OMNI. While OMNI exhibits minimum redundancy, DRP again scores over DDSR in controlling redundancy in the network. For 8 (45 degrees) and 12 (30 degrees) beam models, there are almost 50% more packets in the case of DDSR in comparison with DRP. A general trend observed for both DRP and DDSR is that redundancy decreases with increasing beam-width. This is in tune with the fact that as the number of beams increases, greater number of packets gets transmitted into the network.
5.6.4 Hop Length – Route Optimality

To evaluate the benefits of directional transmission in getting a shorter route, we vary the source-destination separation between two nodes and calculate the hop length of the received routes at the source. We observed little or no variation in the quality of routes (in terms of hop-count) for DDSR and DRP. Therefore, in this section we just compare DRP and OMNI. For a given separation, we picked as many as 5 different source-destination pairs and calculated the hop length for each route. The hop length occurring with the highest frequency was assigned to that separation. Figure 5.11 plots this variation for OMNI and DRP_4, DRP_8 and DRP_12. For larger separations, we observe that DRP_8 and DRP_12 yield shorter routes than a DRP_4 and OMNI. We also observe that there is little or no variation in the hop length for DRP_4 and OMNI. The reasoning is that the range of DRP_4 does not lead to any extra nodes getting covered in our topology (nodes separated by a distance of 250 meters) in comparison to OMNI. The same reasoning can be extended to explain the same hop lengths obtained in the case of DRP_8 and DRP_12.

5.6.5 Throughput in Static Scenario

In this subsection we compare the throughput of DRP with OMNI in a static scenario for single and multiple flows. Once again, we have omitted DDSR from our comparison since DDSR performs almost similar to DRP in a static scenario. We use the grid topology as described in section 5.6.2, and select the separation between the source and the destination node to 1500 meters. Figure 5.12(a) gives the result for a single flow. As evident from Figure 5.12 (a), DRP_8 and DRP_12 outperform OMNI. A 5 hop route in the case of OMNI as opposed to a three hop route for DRP_8/DRP_12 explains the result. The poor performance in case of DRP_4 is mainly contributed by its limited transmission range. In addition, DRP_4 incurs MAC protocol
overheads (sweeping to avoid hidden terminal problem). DRP_8/DRP_12 also incurs the same; however their larger transmission range, which results into shorter routes, offset these limitations. We also observe that throughput saturates at different values for different protocols. For multiple flows, we created two parallel flows separated by 250 meters. The result (Figure 5.12(b)) is consistent with our findings for single flows and DRP_8/DRP_12 are found have better aggregate throughput as compared to OMNI.

![Figure 5.12 Throughput for Static Scenario](image)

**5.6.6 Route Recovery Mechanisms**

DRP uses a three phase mechanism to recover a route in the event of a link failure. In this subsection, we evaluate each of these mechanisms individually. We design special topologies to test our modules against DDSR. No comparisons are made with DSR since the idea here is to stress the need for special routing modules when directional antennas are to be used. However, we do compare all the three different protocols under random mobile scenarios in the later sections.
5.6.6.1 Efficiency of Direction Estimation Algorithm

To evaluate the accuracy of our direction estimation algorithm we created a random topology of 64 nodes in a 2250m by 2250m area. We then selected 10 different source destination pairs and a route was calculated between them over 4, 8 and 12 antenna beam models. For each such pair, the angle of the destination relative to the source was computed using our algorithm. Then the deviation from the actual angle (which is calculated based on actual locations of source and destination) was calculated. The results are listed in Table 5.2. Two observations can be made from this. Firstly, the algorithm performs better for lower beam widths. This can be easily explained as we approximate the next hop to lie on the angular bisector. With lower beam widths, the margin of error is reduced. Secondly, the algorithm estimates destinations to a reasonable degree of accuracy.

<table>
<thead>
<tr>
<th></th>
<th>4-Beam</th>
<th>8-Beam</th>
<th>12-Beam</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean Dev.</td>
<td>22.98</td>
<td>12.81</td>
<td>11.34</td>
</tr>
<tr>
<td>Standard Dev.</td>
<td>4.67</td>
<td>5.19</td>
<td>4.38</td>
</tr>
<tr>
<td>(from mean)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Max. Dev.</td>
<td>34.74</td>
<td>38.79</td>
<td>22.87</td>
</tr>
<tr>
<td>Min. Dev.</td>
<td>9.79</td>
<td>0.583</td>
<td>3.69</td>
</tr>
</tbody>
</table>

Table 5.2 Efficiency of Direction Estimation

5.6.6.2 Location Tracking Mechanism

To evaluate the performance of our antenna beam handoff module, we constructed a linear topology of 6 nodes with inter-node spacing equal to 500 meters. All nodes have an eight beam antenna system. It should be noted that the chances of a handoff are greater in an 8 beam model as compared to a four beam model. CBR traffic with data rate of 100Kbps was started between nodes 1 (source) and 6 (destination). Intermediate nodes were made to oscillate about their initial positions (Figure 5.13(a)) with varying speeds. Such a behavior causes rapid antenna handoffs. Our result (Figure 5.13(b)) indicates that DRP with its antenna handoff module is able to
maintain a fairly constant throughput with increasing node speeds. In contrast, frequent route failures in DDSR leads to a decreasing throughput.

![Topology to evaluate Location Tracking Module. Along the dotted arcs, nodes oscillate.](image1)

![Variation of throughput with mobility](image2)

**Figure 5.13** DRP Location Tracking

### 5.6.6.3 Local Route Recovery Mechanism

In this subsection we evaluate DRP’s *two-hop local recovery mechanism*. To this end, we use a grid topology of 64 nodes as shown in Figure 5.14(a). Nodes 24 and 28 (with four beam antenna system) are chosen as the source and destination nodes respectively. Before the start of the simulation, a linear optimal route of (24, 25, 26, 27, 28) is hard-coded in the route cache of the source node. This is done to offset the initial advantage DRP may have over DDSR due to its enhanced route discovery mechanism. Next, a timer is implemented at the routing layer which periodically fires at 20sec interval. The purpose of this timer is to artificially simulate route failures. Every time the timer fires, the current route are picked and an intermediate node is made inactive. For our entire simulation duration of 200secs, 10 such failures are generated. The antenna handoff module in DRP is disabled, so that the local route recovery module will be invoked whenever there is a link failure. We vary the packet generation rate from 10 packets per
second to 80 packets per sec. Three performance metrics are used to gauge the effectiveness of the local recovery mechanism:

- **Packet Delivery Ratio**: The ratio of the packets received to packets generated
- **Routing Overhead per received packet**: Ratio of the total number of routing control packets (including route requests, route replies and route errors) generated/forwarded to the data packets received correctly at the destination.
- **Average End to End packet delay**: The average end to end delay encountered by each data packet.

![Grid Topology used in simulations](image)

![Packet Delivery Ratio](image)

![Routing Overhead](image)

![Average Latency per Packet](image)

Figure 5.14 DRP Local Route Recovery
The variation of the packet delivery ratio with increasing data rate at the source is shown in Figure 5.14(b). For data rates of 50 packets per second, both DRP and DDSR have nearly the same packet delivery ratio. Beyond this point, the packet delivery ratio falls very steeply in the case of DDSR as compared to DRP. The frequent route failures force DDSR to initiate route discoveries at the source. The combined effect of a greater route discovery latency (DDSR) and increasing data rate at the source lead to a large loss of packets. DRP, on the other hand, buffers packets and quickly recovers routes to the destination. The dip in the packet delivery ratio for DRP can be explained for cases when the local recovery fails. Whenever the local recovery fails, a source rooted discovery follows. Hence for such scenarios, the performance is actually worse than DDSR. However, from our simulation results we conjecture that in most of the scenarios, DRP is able to locally repair the route.

Next, Figure 5.14(c) depicts the routing overhead per received packet at the destination node. For all data rates, the overhead is more in the case of DDSR as compared to DRP. This again is intuitive since DRP does not always resort to a route discovery at the source whenever a link fails. It should be noted that at high data rates (>50 packets per sec), the packet delivery ratio is very low in the case of DDSR. Inspite of this, the overhead at these rates is still more than DRP. This clearly demonstrates the efficacy of the local recovery module of DRP. Figure 5.14(d) depicts the end to end latency experienced by each received packet. Again for higher data rates, this latency is substantially greater in the case of DDSR. Packets wait for a route reply from the destination whenever a route-error occurs. As such, they encounter larger latencies.

5.6.6.4 Zonal Route Recovery Mechanism

Zonal recovery is initiated when both antenna handoff and local recovery fail in DRP. In order to evaluate the effectiveness of this module, we picked six different source-destination pairs in
our grid topology. In addition we disabled the handoff and local recovery module in DRP. As in
the above case, a node was disabled in an active route to generate a link failure. This prompted a
zonal recovery in the case of DRP and a source rooted route discovery in the case of DDSR.
Figure 5.15(a) and Figure 5.15(b) illustrates the comparison of the two protocols with respect to
both routing overhead and the route discovery latency. Route recovery is faster in the case of
DRP. In addition, the routing overhead is also significantly lesser than DDSR.

![Graphs showing redundant route requests and route discovery latency](image)

(a) – Redundant Route Requests  
(b) – Route Discovery Latency

**Figure 5.15 Zonal Route Recovery**

### 5.6.7 Mobile Random Scenario

In this section we compare the performance of DSR, DDSR and DRP in a random scenario.
We generate 64 randomly placed nodes in an area of 2000m by 2000m. Nodes follow the
random waypoint mobility model with varying speeds. Two pairs of source and destination
nodes were selected randomly and CBR traffic with a data rate of 100Kbps was started
between them. The packet delivery ratio, routing overhead per packet and the average end to
end latency per packet is calculated against increasing node speeds. For each topology, we
average the results over 10 different seeds. Figure 5.16(a), Figure 5.16(b), and Figure 5.16(c)
plot the above stated metrics for four and eight beam model. Several interesting observations
are in order. DRP has a very high packet delivery ratio in comparison to DDSR and DSR for both the four beams and eight beam directional antenna models. For an eight beam model, both DRP and DDSR have an almost 100% packet delivery ratio. This can be attributed to the larger range of these models, which result in very few hops. For a four beam model, both DDSR and DSR have an almost equal packet delivery ratio. This ratio drops to less than 50% in cases of high mobility. The trends observed for the routing overhead per received packet were similar for both four and eight beam antenna models. DRP exhibits the least overhead, and DSR the maximum. This is because of the greater number of link failures in the case of DSR as compared to DDSR because of the restricted range of omni-directional antennas. This results in frequent route discoveries leading to a greater overhead. Our results are again consistent for packet latency for both four beam and eight beam models. We observe that DRP has the least latency, whereas DDSR has the maximum latency. The route recovery mechanisms of DRP are responsible for quicker route repair and lesser end to end latencies. DDSR has higher latencies than DSR. This can be explained in context of the MAC overhead incurred in MDA. Intermediate nodes sweep MAC control packets to avoid hidden terminal and deafness problems in DDSR [Agrawal04]. DRP also incurs the same as it also uses MDA. However, since it doesn’t resort to source rooted route discovery at each link failure, its MAC overhead gets offset.

5.7 Conclusion

In this chapter, we have introduced a cross layered directional routing protocol (DRP) specifically tuned to the underlying directional antennas. DRP attempts to alleviate some of the inherent drawbacks involved in directional communications while exploiting the potential benefits such as increased coverage range and directionality. Our simulation results indicate that
DRP has a substantial decrease in route discovery latency as well as directional broadcasting overhead as compared to DDSR. The efficient route recovery mechanisms in DRP prevent any throughput degradation due to frequent movements of intermediate nodes. However, it is worthwhile to note that throughput gain in case of directional antenna systems depends on the topology under consideration.

The design and implementation of DRP finishes our entire directional protocol suite. The remainder of this dissertation focuses MAC and PHY cross layer designs for optimizing the IEEE 802.11 multi-rate networks. We begin by studying the link-delay characteristics of multi-rate networks in the following chapter.

Figure 5.16 Mobile Random Scenario
Chapter 6

6. Performance Evaluation of IEEE Multi-Rate Networks

6.1 Introduction

Performance evaluation of IEEE 802.11 DCF has been widely studied in [Bianchi00, Carvalho03, Carvalho04, and Sasaki04]. However, all these works assume a single transmission rate by all competing nodes. In practice, however, nodes typically use different transmission rates. Such networks are popularly termed as “multi-rate” networks. This leads to the performance anomaly problem which was discovered experimentally and studied with a simple analysis in [Heusse03]. It states that the individual throughput gets bounded by the slowest transmitting peer. However, neither a mathematical formulation of the problem nor mechanisms to remedy it were considered. This chapter provides a simple yet accurate analytical framework to study the performance of multi-rate networks. The remainder of this chapter is organized as followed. Section 6.2 describes an analytical model for multi-rate networks. The model is evaluated in section 6.3. The chapter concludes with some directions for future work in section 6.4.
6.2 Link Delay Characteristic of Multi-Rate Networks

6.2.1 Analytical Model for Link Delay Estimation

We begin by deriving the link delay characteristics for multi-rate networks following the IEEE 802.11 DCF as a MAC protocol. We consider an IEEE 802.11b wireless network which supports four transmission rates, 1, 2, 5.5, and 11Mbps. Each node is assumed to be in a saturated condition\(^5\). We begin by classifying the nodes according to their transmission rates and assign a class identifier \(i = 1, 2, 3,\) and 4 to each class. In general, the total number of classes can be \(N\) (\(N=4\) for 802.11b, \(N=7\) for 802.11a, \(N=9\) for 802.11g). Let \(n^i\) be the number of nodes in class \(i\) and \(L_{av}\) be the average packet length transmitted by all nodes. The minimum contention window, transmission and collision probabilities for each node in class \(i\) is denoted by \(CW_{\text{min}}^i, \tau^i\) and \(p^i\) respectively. Lastly, to simplify our analysis we enumerate all the classes in the order of increasing data rates (Figure 6.1). Class \(i\) and class \((i +1)\) comprises of nodes operating at a transmission rate of \(R^d(i)\) and \(R^d(i+1)\) respectively, such that \(R^d(i) < R^d(i+1)\).

In IEEE 802.11 DCF, if the channel is sensed idle by a node for duration of DIFS, it transmits the packet after completing a random backoff. If a collision is detected, this backoff time is doubled. In general, the node waits for \(r_j\) backoff slots after the \(j^{th}\) collision, where \(r_j\) is uniformly chosen as [Mukherjee05]:

\[
\begin{align*}
    r_j &= U(0, 2^{i} CW_{\text{min}}^i - 1) & j \leq m \\
    r_j &= U(0, 2^{m} CW_{\text{min}}^i - 1) & j > m
\end{align*}
\]  

(6.1)  

(6.2)

Here, \(U\) is the uniform distribution function.

\(^5\) A station which always has another frame waiting for transmission once a frame has been successfully transmitted is referred to be in a saturated condition.
Let $k$ be the number of collisions before a node in class $i$ successfully transmits a packet. Then, the delay $d^{(i)}$ experienced by the packet can be represented as:

$$d^{(i)} = kT_c^{(i)} + rT_b^{(i)} + T_s^{(i)}$$

(6.3)

where the value of the total number of backoff slots, $r$ is obtained as,

$$r = r_1 + r_2 + .... + r_k$$

(6.4)

and $T_c^{(i)}$, $T_b^{(i)}$ and $T_s^{(i)}$ are the average time for collision, backoff and transmission respectively. We note that $T_b^{(i)}$ is the average backoff duration and is different from the backoff slot time ($\sigma$) since the backoff timer freezes whenever any transmission or collision is detected on the network. It has been assumed in this derivation that the delays encountered during each collision are independent of each other. A similar assumption has been made in [Bianchi00] and past observations have shown this to be fairly accurate.

To obtain $T_b^{(i)}$, we observe that the backoff slot time comprises of three components. First is the mandatory time equal to $\sigma$. In addition to this, the backoff slot time may be incremented by $T_c^{(i)}$ or $T_s^{(i)}$ depending upon whether a collision or a successful transmission had occurred during that slot time respectively. Since the respective probabilities for these events are $p_c^{(i)}$ and $p_s^{(i)}$ for class $i$, we obtain $T_b^{(i)}$ as

$$T_b^{(i)} = \sigma + p_c T_c^{(i)} + p_s T_s^{(i)}$$

(6.5)

Here, the products $p_c T_c^{(i)}$ and $p_s T_s^{(i)}$ may be viewed as the expected intervals for a collision or successful transmission in the network when a node from class $i$ is in the backoff stage. The successful packet transmission time $T_s^{(i)}$ for a transmitting node in class $i$ including all the overhead and ACK packet time is given by:
\[ T_s^{(i)} = H_{phy} + \frac{L_{act} + H_{MAC}}{R^d(i)} \]  

\[ T_s^{(i)} = DIFS + T_s^{(i)} + SIFS + H_{phy} + \frac{112}{R^c(i)} \]  

Here, \( R^d(i) \) and \( R^c(i) \) are the data and control rates respectively for class \( i \). At this point, we observe that the duration of a collision depends on the data rate of the slowest node amongst all the colliding nodes. Therefore, the collision interval will equal \( T_s^{(i)} \) if no node in class \( j \) (\( j < i \)) transmits and at least two nodes in class \( k \) (\( k \geq i \)) transmit. Further, we define \( \tau^{(i)} \) and as the transmission probability in a given slot and \( p^{(i)} \) as the conditional collision probability of a node (belonging to class \( i \)). From [Bianchi00], we have:

\[ \tau^{(i)} = \frac{2}{1 + CW_{\min} + p^{(i)}CW_{\min} \sum_{j=0}^{\infty} \left( 2p^{(i)} \right)^j} \]  

\[ p^{(i)} = 1 - \left( \frac{1 - \tau^{(i)} p^{(i)}}{1 - \tau^{(i)}} \right) \prod_{k \neq i} \left( 1 - \tau^{(k)} \right)^{p^{(i)}} \]  

Now, \( p_s T_s^{(i)} \) is the expected time that a node in class \( i \) would wait if a valid transmission were to occur during one of its backoff slots. Also, with the assumption that all packets are of the same length, this duration will be equal to class \( k \)’s packet transmission time if no other nodes are transmitting.

To derive \( p_c T_c^{(i)} \), we observe that the length of this duration will be equal to class \( k \)’s packet transmission time if no other nodes belonging to class \( j(j < k) \) are transmitting. On the other hand, if nodes belonging to class \( j(j > k) \) transmit, the collision time will still be equal to \( p_c T_c^{(i)} \).

Thus, mathematically,
\[ p_i T_c^{(i)} = \sum_{k=1}^{N} T_s^{(k)} \left( \frac{1}{1} \right) \tau^{(k)} \left( \prod_{j=1}^{k} \left( 1 - \tau^{(j)} \right)^{m_{(j)}} \right) \left( \prod_{j=k+1}^{N} \left( 1 - \tau^{(j)} \right)^{m_{(j)}} \right) \]  

(6.10)

\[ p_i T_s^{(i)} = \sum_{k=1}^{N} T_s^{(k)} \left( \frac{1}{1} \right) \tau^{(k)} \left( 1 - \tau^{(k)} \right)^{m_{(k)}} \left( \prod_{j=k+1}^{N} \left( 1 - \tau^{(j)} \right)^{m_{(j)}} \right) \]  

(6.11)

We now estimate the value of the average collision time. The average collision duration for a node of class \( i \) can either be \( T_s^{(i)} \) or \( T_c^{(j)} \) (\( j<i \)). A packet from class \( i \) may collide with another packet of: (i) its same class, (ii) class \( j \) (\( j<i \)) or (iii) class \( k \) (\( k>i \)). In all the three cases, the collision duration for a packet of class \( i \) can never be less than the transmission time for that class. For this reason, we account the ACK time when the packet collides with a packet of the same or a class with nodes at higher transmission rates (since ACK timeout will occur) and neglect the time when the collision is with a packet of a slower class (assuming that the ACK timeout will occur within the collision interval). Hence, \( T_s^{(i)} \) can now be expressed as:

\[ T_s^{(i)} = T_s^{(i)} \left( 1 - \prod_{j=1}^{N} \left( 1 - \tau^{(j)} \right)^{m_{(j)}} \right) \left( \prod_{j=1}^{k} \left( 1 - \tau^{(j)} \right)^{m_{(j)}} \right) + \sum_{j=k+1}^{N} T_s^{(j)} \left( \prod_{j=1}^{k} \left( 1 - \tau^{(j)} \right)^{m_{(j)}} \right) \left( \prod_{j=k+1}^{N} \left( 1 - \tau^{(j)} \right)^{m_{(j)}} \right) \]  

(6.12)

The IEEE 802.11 standard specifies a long retry limit for control packets, e.g. RTS and a short retry limit (for DATA packets) of seven and four respectively, before a packet (control or data) is discarded. However, in our analysis, we consider unlimited retrials. While this assumption is an approximation, it does not skew the results significantly since the probability of more than \( m \) retrials is small, even for saturation. We also note that the majority of the research on the IEEE 802.11 MAC analysis considers infinite retrials [Bianchi00, Battiti04, Carvalho03, and Carvalho04].

The \( cdf \) for \( d^{(i)} \) [Mukherjee05] can now be obtained as:
\[ F_{d(i)}(x) = P(d^{(i)} \leq x) = \sum_{j=0}^{\infty} P(k^{(i)} = j)P(d^{(i)} \leq x | k^{(i)} = j) \]

\[ = \sum_{j=0}^{\infty} P(k^{(i)} = j)P\left(r \leq \frac{x - kT_c^{(i)} - T_s^{(i)}}{T_b^{(i)}} | k^{(i)} = j\right) \]  

\[ (6.13) \]

where \( k^{(i)} \) is the number of collisions in class \( i \).

We know that the \textit{pmf} of the sum of two uniform random distributions is obtained by the convolution of their distributions. Therefore, the \textit{pmf} for \( r \) given \( k \) collisions is obtained as:

\[ D_{k}(r) = \begin{cases} 
U_{\text{CW}_{\text{min}}} \otimes U_{2\text{CW}_{\text{min}}} \otimes \ldots \otimes U_{2^y\text{CW}_{\text{min}}} (r) & k \leq m \\
U_{\text{CW}_{\text{min}}} \otimes U_{2\text{CW}_{\text{min}}} \otimes \ldots \otimes U_{2^y\text{CW}_{\text{min}}} \otimes U_{2^{m+1}\text{CW}_{\text{min}}} (r) & k > m 
\end{cases} \]

\[ (6.14) \]

where \( \otimes \) is the convolution operator and \( U^Y_X \) represents \( Y \) convolutions of a uniform distribution between 0 to \( X-1 \). A similar formulation has been derived by Tikoo and Sikdar in [Tickoo04].

We can now easily derive the distribution for the number of backoff slots, \( r \):

\[ P(r \leq R | k = j) = F_r (R) = \sum_{r=0}^{R} D_j (r) \]

\[ (6.15) \]

Now substituting (15) in (13), and observing that \( P(k^{(i)} = j) = (1-p^{(i)}) \cdot p^{(i)^j} \) \([8]\) we can obtain the distribution for \( d^{(i)} \) as:

\[ F_{d(i)} = \sum_{j=0}^{\infty} (1-p^{(i)})p^{(i)^j}F_r\left(\frac{x - jT_c^{(i)} - T_s^{(i)}}{T_b^{(i)}}\right) \]

\[ (6.16) \]

The corresponding \textit{pmf} \( P_{d(i)}(x) \) is obtained as:

\[ P_{d(i)}(x) = \lim_{h \to 0} \frac{F_{d(i)}(x+h)-F_{d(i)}(x)}{h} \]

\[ (6.17) \]

The average throughput can now be calculated by:

\[ S_{av}^{(i)} = \frac{L_{av}}{\sum_{x=0}^{\infty} xP_{d(i)}(x)} \]

\[ (6.18) \]
6.3 Performance Evaluation

The first part of our performance evaluation presents results from our analytical model for multi-rate networks. We analytically demonstrate the rate-anomaly problem for such network scenarios and the dependence of throughput on the minimum contention window of the contending nodes.

We consider a network topology composed of two nodes in class 4 (11Mbps) and varying number of nodes in class 1 (1Mbps). We note that our analysis is for the basic IEEE DCF and hence the throughputs obtained are higher than that for an RTS-CTS mechanism. The number of nodes in class 1 ranges from 2 to 15. The payload size is fixed at 1500 bytes.

Table 6.1 summarizes the parameters of IEEE 802.11b used in the performance evaluation. We also assume that the network is overloaded, i.e. each node is in a saturated condition.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIFS</td>
<td>20 $\mu$ sec</td>
</tr>
<tr>
<td>DIFS</td>
<td>50 $\mu$ sec</td>
</tr>
<tr>
<td>$\sigma$</td>
<td>20 $\mu$ sec</td>
</tr>
<tr>
<td>ACK</td>
<td>14 bytes</td>
</tr>
<tr>
<td>MAC Header</td>
<td>28 bytes</td>
</tr>
<tr>
<td>PLCP Header Overhead</td>
<td>192 $\mu$ sec</td>
</tr>
<tr>
<td>Data Rates</td>
<td>[1, 2, 5.5, 11] Mbps</td>
</tr>
<tr>
<td>Control Packet Rates</td>
<td>[1, 2, 2, 2] Mbps</td>
</tr>
</tbody>
</table>

Table 6.1 IEEE 802.11b Parameters

Figure 6.1 Classification of nodes on basis of data rates
6.3.1 Effect of Varying Minimum Contention Window

Figure 6.2 and Figure 6.3 illustrate the improvement in the aggregate throughput through the differentiated minimum contention window. Class 4 nodes use the default the initial $CW_{min} = 32$ while class 1 (1 Mbps nodes) use larger values of $CW_{min}$ ranging from 32 to 352, while maintaining the maximum backoff window as 1024. The upper value of 352 represents a ratio of 11 between the $CW_{min}$ of class 1 and class 4 nodes. When the $CW_{min}$ are same for both the classes ($=32$), and there are two nodes each in both the classes, the throughputs of classes 1 and 4 are both 1.6Mbps. Therefore, the saturation throughput of each node in class 4 becomes 0.8Mbps (equal to the saturation throughput of each node in class 1) even though nodes in class 4 transmit data 11 times faster. This exhibits the behavior of performance anomaly in multi-rate networks. Moreover, we observe that the throughput of nodes in class 4 decreases as the number of nodes in class 1 increases.

The performance anomaly phenomenon starts diminishing as we increase the $CW_{min}$ of the nodes in class 1 from 32 to 352. Therefore with $CW_{min}$ as 32 and 352 respectively, and total number of nodes as 4 (2 in each class), we observe that the throughput for nodes in class 4 increases dramatically from 0.8Mbps to 4Mbps. The throughput for class 1 nodes, expectedly, drops down to 0.35Mbps. With differentiated minimum contention window sizes for different classes, the total throughput increases almost three times from 1.5 to 4.4Mbps.

The reasoning for this result is obvious. A higher contention window leads to fewer transmission opportunities and thereby increases the performance of nodes contending with smaller window size. We would expect the performance of class 4 nodes to keep increasing (and consequently class 1 nodes throughput to keep decreasing) with an increasing $CW_{min}$ of class 1 node. Therefore, the important question is to determine the optimal ratio of the minimum contention window, so that the throughput for both the flows is proportionally fair. Therefore,
with $CW_{\text{min}}$ of 256 (which is close to a ratio of 8), we observe that nodes in classes 1 and 4 achieve an individual throughput of 0.4 and 1.5 Mbps respectively, which is similar to baseline performance of both these flows (i.e. the throughput they each node will achieve if all other nodes are operating at the same rate). This ratio also corresponds to an air-time fair channel allocation. Ideally, we would like our proposed protocol to operate around this point.

![Figure 6.2 Throughput dependence on $CW_{\text{min}}$](image1)

![Figure 6.3 Aggregate throughput](image2)

### 6.3.2 Effect of Varying Packet Payload Sizes

This sub-section discusses the impact of packet payload. We fix the payload size for class 4 (11Mbps) nodes to 1500 bytes and vary the payload size of class 1 (1 Mbps) nodes from 256 to 1500 bytes. In addition, both the classes use the same value for the minimum contention window (i.e. 32).

The throughput of high rate nodes slightly increases as the size of packet payload decreases for the low rate nodes. This is depicted in Figure 6.4 and Figure 6.5. When there are four nodes in the network, the throughput of each node in class 4 increases from 0.8Mbps to 2.5Mbps as the payload size for class 1 decreases from 1500bytes to 256 bytes. Smaller payload sizes correspond to lesser air-time. Therefore, although nodes from both the classes have equal
transmission opportunities (since $CW_{\text{min}}$ is same), class 1 nodes occupy lesser air-time, which benefits nodes of class 4.

![Figure 6.4 Throughput versus Packet Size](image1)

![Figure 6.5 Aggregate Throughput](image2)

### 6.3.3 Link Delay Characteristics

Figure 6.6 and Figure 6.7 plot the c.d.f of the link delay for class 1 and class 4 nodes respectively. The number of nodes was fixed to five nodes in each class. We vary the $CW_{\text{min}}$ of class 1 nodes from 32 to 352, while maintaining the $CW_{\text{min}}$ of class 4 nodes at 32. We observe similar trends to those in section 6.3.1. As the $CW_{\text{min}}$ of class 1 increases, the link delay for class 4 nodes decreases. The delay characteristics are similar when the minimum contention window are same, illustrating the rate anomaly problem.

![Figure 6.6 Delay Characteristics (1Mbps)](image3)

![Figure 6.7 Delay Characteristics (11Mbps)](image4)
6.4 Conclusion

This chapter discusses an analytical model which provides the link delay characteristics and average throughput for multi-rate networks. It helps in providing a mathematical formulation for the rate anomaly problem. We observe that the performance in a multi-rate scenario is closely tied to the minimum contention window of nodes transmitting at lower data rates.

Using the insights obtained from the analytical model, the following chapter concentrates on developing a time-fair extension of IEEE 802.11, Time Fair CSMA (TFCSMA).
Chapter 7

7. Air Time Fairness to Mitigate Performance Anomaly Problem

7.1 Introduction

Over the last six years, the IEEE 802.11 working group has come up with physical layer enhancements to support data rates of the order of 54Mbps. Infact; the 802.11n working group is exploring transmission rates of upto 200Mbps. Further, the standard also supports Dynamic Rate Adaptation (DRA) which allows nodes to dynamically adapt the transmission data rate to their channel conditions. Propriety algorithms are frequently used by vendors to adaptively adjust the data rate with an intention to maximize throughput performance.

While intuitively there appears a direct relation between the data rate and the throughput, the recent seminal work by Heusse et al. [Heusse03] suggests otherwise. Surprisingly, under multi-rate networks\(^6\), the long term throughput of each node becomes largely independent of its own data rate; rather it gets bounded by the lowest data rate peer. An obvious implication of this phenomenon restricts the achievable benefits of any rate adaptation mechanism. In particular, the presence of a low data rate link down-equalizes the aggregate throughput, thereby restricting any

---

\(^6\) We define multi-rate networks as networks where different nodes may employ different transmission data rates.
benefits of higher bit rates by the peers. This phenomenon is better explained by the example topology in Figure 7.1. Node 1 has a two hop route to node 3, with node 2 as the intermediate forwarded. Both the links are assumed to be perfect, in the sense that they can transmit at the highest data rate of 11Mbps. Node 2 shares the channel with another link $R$ (node 4 to node 5). Figure 7.2 depicts the variation of the end to end throughput and the channel occupancy ratio (COR) for node 2 with the transmission rate of link $R$. Throughput decreases by 63% as the rate for link $R$ varies from 11Mbps to 1Mbps. We notice that both the links have approximately the same throughput even though node 2 always transmits at 11Mbps. This can be explained by the ratios of COR for the two competing nodes which is roughly proportional to their respective data rates. This clearly suggests that slower links occupy more channel air time to ultimately transmit the same number of packets as faster links.

This phenomenon has been quoted as the “rate-anomaly” problem of IEEE DCF networks. Its cause is rooted in the fairness philosophy of 802.11 which ensures long term equal channel access probability. This implies that if similar sized packets are used under similar channel conditions, then each node achieves roughly the same throughput irrespective of its own transmission rate. Now, if all the competing nodes employ similar data rates, 802.11 automatically guarantees equal time shares as well. However, under multi-rate networks, each node receives a time share which is roughly inversely proportional to their transmission data rate. This translates into faster nodes incurring a penalty since they invariably need to wait for their slower peers to complete their transmissions/retransmissions. This chapter addresses the behavior of nodes under multi-rate scenarios. It focuses on an extension of IEEE 802.11 DCF - TFCSMA: Time Fair CSMA, an online algorithm for ensuring long term air-time fairness, thereby mitigating the performance anomaly problem of the IEEE multi-rate networks.
The underlying idea behind TFCSMA is to allocate equal time shares to all competing nodes in a distributed manner such that node employing higher data rates receive greater transmission opportunities, thereby increasing the aggregate throughput. We propose a simple yet efficient mechanism by which each node can maintain a running estimate of its *packet error rate* (PER). The estimated PER and the employed transmission rate is used to dynamically determine a target throughput by each node. This calculation is based on the *baseline property* Tan04[Tan04] which guarantees air time fairness in the network if the target throughputs are exactly met by all the nodes. Every node attempts to meet its respective target by dynamically adjusting its minimum contention window (CWmin) and hence controlling its transmission opportunities. As opposed to the previous works [Tan04, Kim05] which assume equal packet lengths and scale the contention window in proportion to the data rate (relative to the absolute maximum possible data rate), TFCSMA directly alters the transmission probability based on the channel occupancy, thereby providing links which promise greater throughput, greater transmission opportunities. We thoroughly evaluate the performance of TFCSMA by means simulations and comparisons to existing approaches, showing better performance in terms of fairness and throughput.

The rest of the chapter is organized as follows. The fairness objectives for multi-rate networks are discussed in section 7.2. The inefficiencies in the existing solutions for rate anomaly problem are highlighted in section 7.3. Section 7.5 describes the functioning of TFCSMA in. We evaluate the performance and provide comparisons in section 7.6. The chapter concludes in section 7.7 with some interesting directions for future work.
Since the 802.11 MAC implicitly provides equal transmission opportunities, slower nodes tend to occupy more air-time as compared to faster nodes, provided equal sized packets are used by all nodes. Clearly, the max-min throughput fairness policy is not appropriate for such networks.

Throughput fairness fails to guarantee air-time fairness when multiple nodes with different data rates compete with each other. This neutralizes any advantage of employing higher rates and thereby adversely lowers the aggregate throughput. Maximizing throughput is also not a viable option since that will tantamount to allowing only the high rate nodes to transmit, thereby leading to gross unfairness and possibly starvation of the slower nodes.

Reference [Radunovi03] recommended proportional fairness as the objective of resource allocation to strike a balance between fairness and throughput for rate diverse networks. When multiple nodes contend for a single resource, proportional fairness tries to allocate the resource in proportion to a node's individual capacity. If channel bandwidth is taken as the resource in question, then proportional fairness shall strive to ensure that each node achieves an individual throughput proportional to its capacity.

This idea was further explored by Tan et. al. [Tan04, Tan05] in the context of time-based fairness. They suggested an interesting baseline property, the consequence of which is time...
fairness. The property states that under equal time shares for all nodes, the long term throughput a node achieves competing against $n$ nodes operating at varied data rates is identical to the throughput the node would achieve if it were competing against $n$ nodes all operating at its data rate. While the property was validated through simulations, no formal mathematical analysis or proof was provided for the same.

Li et al [Battiti04] demonstrated the equivalence of proportional and time based fairness for WLANs. They proved that air time fairness was a natural consequence of the more fundamental property of proportional fairness. Based on their analysis, they showed that under temporal fairness, the ratio of throughputs of two nodes $i$ and $j$ is proportional to their respective data rates $R_i$ and $R_j$ i.e.: 

$$\frac{S_i}{S_j} = \frac{R_i}{R_j}.$$ However, their analysis neglected protocol overheads and channel errors and hence the result stands as a weak approximation.

If the effective good channel time is denoted by $T$ and assuming that this time is equally shared by all $n$ nodes, then: $S_i \propto \frac{T}{nT_s}$, where $S_i$ is the throughput and $T_s$ is the same as in equation 7.1.

Therefore, we have, $\frac{S_i}{S_j} = \frac{T_{s_i}}{T_{s_j}}$. If the baseline property has to be true, then $S_i^p T_s = S_j^p T_s$, where $S_i^p$ is the throughput of node $i$ if all other nodes were operating at $R_i$. Using [Bianchi00], we can compute this ratio (say $r_{ij}$):

$$r_{ij} = \frac{S_i^p T_s}{S_j^p T_s} = \frac{(1-P_s)\sigma + P_s P_s + P_s (1-P_s)T_{s_j}}{(1-P_s)\sigma + P_s P_s + P_s (1-P_s)T_{s_j}}$$ (7.1)

Here, $P_s$ is the probability of at least one transmission in a slot and $P_s$ is the probability of a successful transmission in a slot. Clearly $r_{ij}$ must approach 1 for any $i$ and $j$ if the baseline
property has to hold. Figure 7.3 plots $r_{ij}$ for $i=1$Mbps and $j=11$Mbps respectively, against the total number of nodes, for different payload sizes. The choice of 11 and 1 Mbps is obvious since these rates reflect the maximum possible disparity. Several interesting observations are in order from this plot. First, we observe that that $r_{ij}$ increases with increasing payloads. This implies that the accuracy of the baseline property increases with data payload size. Second, we observe that for lesser number of nodes (20-40), $r_{ij}$ is close to 0.98 for almost all data payload sizes. Therefore, the baseline property is sufficiently correct for a contention region of around 40 nodes. This is an interesting observation, since a neighborhood cardinality of 40 is a reasonable assumption for a wireless network (both WLANs and Ad-Hoc). Lastly, we note that the ratio decreases with increasing nodes in the network. For a packet size of 1500 bytes and 200 nodes, $r_{ij}$ is observed to be around 0.93. Figure 7.4 plots $r_{ij}$ for $i=1$Mbps and $j=2$Mbps respectively. Similar trends are observed as stated above, except that a higher degree of accuracy is attained via the baseline property. For a neighborhood size of 200 nodes and a packet size of 1500bytes, $r_{ij}$ is observed to be around 0.99.

The baseline property is an interesting result for developing a time fair extension of 802.11. It allows deriving an expected performance benchmark for a node, provided $n$ is known or can be estimated. A simple control algorithm can then be developed to dynamically adjust the $CW_{min}$ (and hence the transmission probability) to achieve this target in a distributed way. Of course, such a mechanism would not ensure exact temporal fairness (as we showed that the baseline property is a strong approximation), it can nevertheless provide considerable performance improvements.
We now discuss the contention window scaling protocol proposed in [Kim05] to rectify the anomaly problem. The authors claim that the phenomena can be cleanly resolved via setting the initial contention window size inversely proportional to the bit rates. A similar idea is also proposed in [Heusse05]. Both these methods can be summarized by the following equation:

\[ \frac{CW}{CW_c} = \frac{R_{\text{max}}}{R} \]  

(7.2)

In this case, if a node transmits at the highest available rate \(R_{\text{max}}\), it uses a predefined minimum contention window \(CW_c\). If a node transmits at a lower rate \(R\), it modifies the minimum contention window by scaling it with respect to the maximum data rate. In case of [Heusse05], \(CW_c\) is computed using an algorithm which measures the number idle slots on the channel.

7.4 Shortcomings

7.4.1 Channel Underutilization

While CWSP is a very simple mechanism to resolve performance anomaly, it suffers from several shortcomings. Firstly, it can lead to a considerable underutilization of the channel. Consider a scenario where there are only nodes A and B operating at 1 and 2 Mbps respectively.
Such a scenario may exist when the channel conditions in a neighborhood are poor due to multi-path or reflections from the surrounding. If $C_W$ is taken as 31 for 11 Mbps (the maximum data rate), then nodes A and B shall have a $CW_{\text{min}}$ of 341 and 171 respectively. Notice that the anomaly could have been resolved with window sizes of 62 (node A) and 31 (node B). In other words, the contention window needs to be scaled with respect to the fastest transmitting peer, and not the absolute maximum data rate. This assumes greater importance especially when the ratio between the slowest and the fastest possible rate is sufficiently large. Table 7.1 compares the aggregate throughput for a 2 node network (described above) for different maximum allowed data rates. We notice that an Optimal Scaling Protocol (OSP: a theoretical protocol which adjusts the contention window with respect to the fastest peer) always outperforms CWSP. The aggregate throughput for CWSP drops by 28% (compared to IEEE 802.11 DCF) when the maximum allowable transmission rate is 54Mbps. The suboptimal values of $CW_{\text{min}}$ (1674 and 837 for node 1 and node 2 respectively) are primarily responsible for a large number of idle slots on network, bringing down the aggregate throughput. Further, we observe that for all the scenarios, OSP increases the throughput by 12% with respect to the IEEE 802.11.

<table>
<thead>
<tr>
<th></th>
<th>802.11</th>
<th>CWSP</th>
<th>OSP</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max Rate: 2 Mbps</td>
<td>1127.5</td>
<td>1253.8</td>
<td>1261.6</td>
</tr>
<tr>
<td>Max Rate 11 Mbps</td>
<td>1127.5</td>
<td>1142.4</td>
<td>1261.6</td>
</tr>
<tr>
<td>Max Rate 54 Mbps</td>
<td>1127.5</td>
<td>804.64</td>
<td>1261.6</td>
</tr>
</tbody>
</table>

Table 7.1  Aggregate Throughput (Kbps)

Figure 7.5  Three node example topology
7.4.2 Protocol Overhead Considerations

Secondly, we argue that the ratio in equation 20 should indeed be equal to $\frac{T_{\text{tx}}}{T_{\text{max}}}$ instead of $\frac{R_{\text{max}}}{R}$. The reasoning is intuitive. The ratio of the contention windows should take into account any protocol overheads (Physical layer, Acknowledgements etc) as well as payload sizes. This is better understood by Figure 7.6 which plots the throughput of each individual flow for the topology in Figure 7.5. Temporal fairness is achieved at a ratio of around 7.0 instead of 11. Notice that the throughput at this point is almost equal to the baseline throughputs, validating the exact accuracy of the property for smaller number of nodes.

![Figure 7.6 Throughput v/s minimum CW](image1)

![Figure 7.7 Dependence on Error Rates](image2)

7.4.3 Wireless Link Error Considerations

Thirdly, and perhaps the most important flaw is that CWSP ignores the time varying properties of a wireless channel. The quality of the received signal at a receiver may vary considerably over a short period of time due to noise, interference, multi-path or Doppler effects (due to mobility). This can lead to bit errors and possibly packet corruption at the receiver. The per or packet error probability increases with higher transmission rates since they demand a higher signal to noise ratio (SNR). Consider a scenario, where there are two transmitting nodes A and B operating at 1
and 11 Mbps respectively. Let $per_a$ and $per_b$ be the error rates at A and B’s respective receivers. It can be safely assumed that $per_a << per_b$. Now, CWSP attempts to achieve proportional throughput fairness by modifying the transmission probability of a node in proportion to its data rate. What it implicitly assumes is that the error probability is equal for all rates, which of course is not true for any practical scenario. Clearly, under such circumstances, scaling the contention window of node A 11 times with respect to node B is incorrect. Figure 7.7 depicts the dependence of time fairness on different error rates. To ensure a time-fair allocation of the channel, the ratio of the minimum contention windows for node A and B increases with the $per$ for the 11Mbps (node B) link. This is intuitive since the IEEE 802.11 DCF mandates an exponential backoff even when a packet loss is due to link bit errors. As a result, erroneous links demand greater transmission opportunities, and hence the ratio increases. Therefore, the idea of using a fixed ratio (as in CWSP) will clearly not meet the objective of temporal fairness when different links have different error rates.

7.5 Proposed Protocol - TFCSMA

7.5.1 Protocol Overview

This section describes our proposed protocol: Time Fair Carrier Sense Multiple Access (TFCSMA). TFCSMA is an online, totally distributed extension of IEEE 802.11 DCF for ensuring air time fairness under multi-rate scenarios. TFCSMA utilizes the baseline property to dynamically derive a performance benchmark (target throughput) for a node based on its error and transmission data rates. It then uses a simple control algorithm to dynamically adjust the $CW_{min}$ (and consequently the transmission probability) to achieve this target in a distributed way. The protocol is completely compatible with IEEE 802.11 and requires no changes to the basic access mechanism. Infact, the access procedure is exactly same as the IEEE 802.11. It however
mandates the use of RTS-CTS since that facilitates the calculation of Packet Error Rates (PER). However, it can still work with the basic CSMA/CA, provided an alternative method is used to estimate per, or its effect is ignored.

TFCSMA has three basic modules: (i) PER Estimator, (ii) Target Estimator and (iii) Minimum Contention Window Controller. The following sub-sections describe their roles in greater detail.

7.5.2 Packet Error Rate Estimator (PERE)

The Packet Error Rate Estimator maintains a moving average of the packet error rate. TFCSMA mandates the use of an RTS-CTS exchange before a data packet is transmitted. This is necessary to decouple collisions from packet errors due to bit errors in the channel. RTS-CTS control packets reserve the channel around the transmitter and the receiver respectively and thereby minimize the probability of collision of DATA packets. Therefore, a failure of an ACK reception after a data packet transmission following an RTS-CTS exchange can be assumed to be a result of link error. Of course, we note that this mechanism isn’t completely accurate. Collisions can still occur after an RTS-CTS exchange [Xu02], however the probability of its occurrence are minimal. Therefore, for the purpose of this work, we neglect any packet losses due to collisions after an RTS-CTS exchange.

PERE constantly updates a moving average of the packet error rate for different transmission rates and packet sizes. It maintains a two dimensional table \( T \), where \( T[i][j] \) indicates the \( \text{per} \) at rate index \( i \), and packet size index \( j \). PERE continuously counts the number of ACKs received

---

7 For simplicity, we assume a single receiver per transmitting station (as in WLANs) and hence propose a two dimensional table. In an ad-hoc scenario, a station may have multiple receivers. In that case, we would require a three dimensional table so that the \( \text{per} \) can be maintained for every receiver/link.

8 While maintaining tables, we denote different transmission rates and packet sizes by integers. For example rates 1, 2, 5.5 and 11Mbps are denoted by rate indices: 0, 1, 2 and 3 respectively.
within a window of transmitted packets ($\lambda$) (using the statistics report\(^9\)) and maintains $T$ by the following equation:

$$\text{per}(i, j) = \alpha \left(\frac{\beta}{\lambda}\right) + (1 - \alpha) \ast \text{per}^{\text{prev}}(i, j)$$  \hspace{1cm} (7.3)

Here ($\alpha$) denotes the forgetting factor for the moving average estimate. A typical PER table is depicted in Table 7.2.

<table>
<thead>
<tr>
<th></th>
<th>1Mbps</th>
<th>2Mbps</th>
<th>5.5Mbps</th>
<th>11Mbps</th>
</tr>
</thead>
<tbody>
<tr>
<td>256-512 (bytes)</td>
<td>0.99</td>
<td>0.99</td>
<td>0.99</td>
<td>0.99</td>
</tr>
<tr>
<td>512-1024 (bytes)</td>
<td>0.99</td>
<td>0.98</td>
<td>0.94</td>
<td>0.91</td>
</tr>
<tr>
<td>1024-1536 (bytes)</td>
<td>0.99</td>
<td>0.9</td>
<td>0.86</td>
<td>0.81</td>
</tr>
<tr>
<td>&gt; 1536 (bytes)</td>
<td>0.99</td>
<td>0.87</td>
<td>0.81</td>
<td>0.75</td>
</tr>
</tbody>
</table>

Table 7.2 PER Table

### 7.5.3 Target Throughput Estimator (TTE)

The TTE is a module which selects the target throughput, $S'$ for a node based on its current transmission rate, packet error rate and packet payload size. It maintains a three tuple table $T_{\text{baseline}}$, where $T_{\text{baseline}}[i][j][k]$ denotes the saturation throughput if there were $k$ competing nodes, all operating at data rate index $i$ and packet size index $j$. This table is computed offline using the expressions\(^{10}\) derived in [Li04]. Table 7.3 depicts a snapshot of a truncated offline table.

TTE exploits the baseline property for achieving air time fairness. Each node assumes that every other node in the network is operating at its own data rate, packet size and is also experiencing similar packet errors. Based on these three parameters, the TTE selects $S'$ from the table $T_{\text{baseline}}$. Under similar data rates and packet sizes, 802.11 guarantees both time and throughput fairness. Hence, if every node is able to achieve its target, then air time fairness is automatically guaranteed by virtue of the baseline property.

---

\(^9\) After every successful packet transmission (ACK reception) or after a packet has been dropped (its retransmission numbers exceed the retry limit), a statistic report is generated by the MAC. This provides complete information about the number of retrials, the end to end delay, backoff slot duration etc for the current packet transmission.

\(^{10}\) We have slightly modified the expressions to account for RTS-CTS exchange in the calculation of the throughput.
It is assumed that every node is aware of the number of competing nodes\textsuperscript{11} in the network. In addition, we also assume that each node operates in a \textit{saturated} traffic condition.

The TTE works in conjunction with the PERE for dynamically determining $S'$. Just before a DATA packet transmission, TTE verifies with the PERE, the current \textit{per} for the transmission rate and packet size in question. It then indexes into $T_{\text{baseline}}$ to compute the target throughput. Referring to Table 7.3, we note that $T_{\text{baseline}}$ stores the offline targets for a given range of the packet error rate (like 0 to 10\%). In other words, the offline throughput for a data rate of $R$ Mbps and $P$ packet size, remains the same for a \textit{per} of 0 to 10\% (computed using the mean value, i.e. 5\% in this example). Therefore, $S'$ doesn’t fluctuate unless the error rate changes substantially (or the node decides to change its transmission rate/packet payload). For the present implementation of TFCSMA, a variation of more than 0.1 in the \textit{per}, forces a new target throughput.

\textbf{7.5.4 Minimum Contention Window Controller (MCWC)}

A control algorithm is required to track the measured throughput and dynamically adjust the minimum contention window (and consequently the transmission probability) to make it converge to the target throughput in a distributed manner. The MCWC is responsible for guiding a node to its respective target throughput. The transmission probability is inversely proportional to the minimum contention window. Also, the measured throughput is proportional to the transmission probability. By combining these two ideas, we arrive at our control algorithm. The MCWC studies the statistic reports for a window of \textit{aSamplePackets}\textsuperscript{12} successful packet

\begin{footnotesize}
\textsuperscript{11}This is clearly the weakest assumption in this entire work. The authors agree that this estimation is not a trivial task, particularly in the context of multi-rate networks. It is one of our ongoing works to dynamically compute the number of active nodes in a typical multi-rate mobile scenario.

\textsuperscript{12}Measuring throughput on a per packet basis may result in erroneous fluctuations. Hence, the average throughput over a sample number of packets is used as the test criterion for the minimum contention window in use. For the
\end{footnotesize}
transmissions, and adjusts the $CW_{min}$ for the next window of transmissions by the following control equation:

$$CW_{min}^{new} = CW_{min}^{prev} * \frac{S^m}{S'}$$  \hspace{1cm} (7.4)$$

where, $S^m$ is the measured throughput, computed from the link delay obtained from the statistic reports.

<table>
<thead>
<tr>
<th>PER</th>
<th>1Mbps</th>
<th>2Mbps</th>
<th>5.5Mbps</th>
<th>11Mbps</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 – 10%</td>
<td>165.7806</td>
<td>309.3475</td>
<td>663.1980</td>
<td>985.1680</td>
</tr>
<tr>
<td>10 – 20%</td>
<td>140.8819</td>
<td>262.7264</td>
<td>562.7127</td>
<td>835.1757</td>
</tr>
<tr>
<td>20 – 30%</td>
<td>124.2104</td>
<td>231.4084</td>
<td>494.6385</td>
<td>732.8020</td>
</tr>
<tr>
<td>30 - 40%</td>
<td>90.5331</td>
<td>167.6853</td>
<td>353.6547</td>
<td>517.6961</td>
</tr>
<tr>
<td>40 – 50%</td>
<td>82.0205</td>
<td>151.4574</td>
<td>317.1596</td>
<td>461.3809</td>
</tr>
<tr>
<td>50 – 60%</td>
<td>56.3594</td>
<td>102.457</td>
<td>206.9250</td>
<td>291.9967</td>
</tr>
<tr>
<td>60 – 70%</td>
<td>39.4344</td>
<td>70.5022</td>
<td>137.2705</td>
<td>188.1919</td>
</tr>
</tbody>
</table>

Table 7.3 Snapshot of Offline Throughput Table

If the measured throughput is observed to be greater in comparison to the target throughput, a node increases its $CW_{min}$. What it actually does is that it reduces its transmission probability and consequently pushes down the throughput. Now, if $S^m < S'$, the contention window is decreased appropriately, and the measured throughput increased.

---

The present implementation of TFCSMA, $aSamplePackets$ was taken as 10. The test criterion is abandoned (and a new test started) if the target throughput changes within this test window.
7.6 Performance Evaluation

To evaluate the performance of our protocol, we have extended the IEEE 802.11 implementation in OPNET [Opnet] that implements TFCSMA and CWSP. We present results for an IEEE 802.11b MAC which uses Direct Sequence Spread Spectrum (DSSS) as the physical layer technology. The parameters for TFCSMA are the following: $\lambda = 5$ and $a\text{SamplePackets}=10$. A simulation runs for duration of 300 seconds and each value if averaged over five different runs. Each node operates under saturated traffic load conditions.

7.6.1 Throughput

The goal of TFCSMA is to maximize throughput while providing air-time fairness. In this part, we evaluate the throughput of our proposed protocol, but we do not expect an extra-ordinary increase. The network topology comprises of two nodes in class 4 and varying number of nodes in class 1 (from 2-14). In addition, the $per$ for nodes in class 4 is varied from 0 to 0.75 in steps of 0.25. We compare the throughput achieved by the 802.11b, CWSP and TFCSMA. Unless stated otherwise, throughput of a class refers to the individual throughput of a node in that particular class.
The throughput of class 4 nodes decreases with an increase in the number of nodes in class 1 for all the three protocols. Consequently, a reverse trend is observed for class 4 throughputs. This is obvious; since the throughput of class 1 nodes is bound to increase with an increase in their total number.

We observe that TFCSMA and CWSP consistently outperform IEEE 802.11b as far as the throughput for class 4 nodes are concerned. This difference increases as the \( \text{per} \) varies from 0 to 0.75. When there are two nodes each in both the classes and the \( \text{per} \) is 0.5, TFCSMA (and CWSP) provide a throughput of 1.02 Mbps as opposed to 0.18 Mbps by 802.11. The rate anomaly phenomenon is responsible for this behavior.

Next, we observe for all packet error rates (except \( \text{per} = 0.75 \)) that CWSP provides greater class 4 throughput in comparison to TFCSMA. Further, the class 1 throughput of CWSP is consistently lower than TFCSMA (except \( \text{per} = 0.75 \)). Both these results are related. CWSP uses a fixed contention window ratio of 11 (between class 1 and class 4). Hence, CWSP ends up favoring class 4 flows, thereby increasing their throughput and decreasing the throughput of class 1 nodes. However, this difference diminishes as the \( \text{per} \) varies from 0 to 0.5. Infact at a \( \text{per} \) of 0.75, we note that the class 4 throughput of TFCSMA surpasses that of CWSP. This can be attributed to the sub-optimal fixed \( CW_{\text{min}} \) ratio which fails to provide sufficient transmission opportunities to class 4 nodes (under extremely poor link conditions), thereby restricting their performance.
7.6.2 Fairness

As stated above, the primary objective of TFCSMA is to provide air-time fairness under multi-rate scenarios. To measure air time fairness, we define a fairness index (FI) as:

\[ FI = \frac{|S_{avg} - S'|}{S'} \]  

(7.5)

where \( S_{avg} \) is the average throughput of flow of a flow over the simulation duration. \( S' \) is the target or the baseline throughput for a flow. \( FI \) provides an extent of deviation from the target throughput and thereby gives a normalized measure of air time fairness in the network. A \( FI \) of close to 0 represents air time fair channel allocation whereas a value of near one (or greater than one) indicates gross unfairness in the network.
To evaluate and compare the fairness of TFCSMA, we use a similar topology described in section 10.6.1. Figure 7.13 to Figure 7.16 plot the fairness index for varying packet error rates. Table 7.4 denotes the target (or baseline) throughput for individual flows. Almost under all the scenarios, TFCSMA outperforms both 802.11 and CWSP. It consistently provides an $FI$ close to 0 for all the flows in the network. As expected, 802.11 is the least air time fair protocol amongst all the three protocols.
7.6.3 Channel Utilization

We discussed in section 7.2 that CWSP can lead to a considerable channel underutilization since it scales the $CW_{min}$ with respect to the absolute maximum data rate instead of the fastest transmitting peer. To evaluate the performance of TFCSMA under such scenarios, we consider a network topology composed of an equal number of nodes from both class 1 (1Mbps) and class 2 (2Mbps). We vary total nodes from 4 to 10.

<table>
<thead>
<tr>
<th>Node</th>
<th>4</th>
<th>6</th>
<th>8</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>802.11 (Kbps)</td>
<td>1082.69</td>
<td>1081</td>
<td>1074.51</td>
<td>1088</td>
</tr>
<tr>
<td>CWSP-2 (Kbps)</td>
<td>1213.26</td>
<td>1208.8</td>
<td>1213.23</td>
<td>1222</td>
</tr>
<tr>
<td>CWSP-11 (Kbps)</td>
<td>1164</td>
<td>1184.4</td>
<td>1192.43</td>
<td>1209</td>
</tr>
<tr>
<td>CWSP-54 (Kbps)</td>
<td>940</td>
<td>1012</td>
<td>1056.99</td>
<td>1095.3</td>
</tr>
<tr>
<td>TFCSMA (Kbps)</td>
<td>1238</td>
<td>1248</td>
<td>1251</td>
<td>1254</td>
</tr>
</tbody>
</table>

Table 7.5 Aggregate Throughput Results

<table>
<thead>
<tr>
<th>Station</th>
<th>4</th>
<th>6</th>
<th>8</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>802.11</td>
<td>0.2534</td>
<td>0.2624</td>
<td>0.2661</td>
<td>0.2686</td>
</tr>
<tr>
<td>CWSP-2</td>
<td>0.1233</td>
<td>0.1053</td>
<td>0.1282</td>
<td>0.1106</td>
</tr>
<tr>
<td>CWSP-11</td>
<td>0.1495</td>
<td>0.1361</td>
<td>0.1308</td>
<td>0.1129</td>
</tr>
<tr>
<td>CWSP-54</td>
<td>0.2671</td>
<td>0.1983</td>
<td>0.1766</td>
<td>0.1439</td>
</tr>
<tr>
<td>TFCSMA</td>
<td>0.0257</td>
<td>0.0376</td>
<td>0.0537</td>
<td>0.0518</td>
</tr>
</tbody>
</table>

(a) - Fairness Index (Class 1 flows)

<table>
<thead>
<tr>
<th>Station</th>
<th>4</th>
<th>6</th>
<th>8</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>802.11</td>
<td>0.3172</td>
<td>0.3192</td>
<td>0.3342</td>
<td>0.3096</td>
</tr>
<tr>
<td>CWSP-2</td>
<td>0.0476</td>
<td>0.0372</td>
<td>0.0501</td>
<td>0.0612</td>
</tr>
<tr>
<td>CWSP-11</td>
<td>0.0021</td>
<td>0.0231</td>
<td>0.0255</td>
<td>0.0465</td>
</tr>
<tr>
<td>CWSP-54</td>
<td>0.2171</td>
<td>0.1598</td>
<td>0.1192</td>
<td>0.0799</td>
</tr>
<tr>
<td>TFCSMA</td>
<td>0.0013</td>
<td>0.0101</td>
<td>0</td>
<td>0.0121</td>
</tr>
</tbody>
</table>

(b) - Fairness Index (Class 2 flows)

Table 7.6 Fairness Index

Table 7.5 presents the comparison of the aggregate throughput between 802.11, CWSP-2, CWSP-11, CWSP-54\(^\text{13}\) and TFCSMA. TFCSMA outperforms all other protocols and provides the maximum aggregate throughput under all scenarios. Table 7.6(a) and Table 7.6(b) compare the air-time fairness for both the flows. We can see that TFCSMA provides much better fairness in comparison to all other protocols. 802.11 and CWSP-54 are the least fair, exhibiting gross unfairness.

\(^{13}\) CWSP-\(k\) refers to a network where the maximum permissible data rate is \(K\) Mbps. This implies that class 1 nodes use a $CW_{min} = 32*K$, while class 2 nodes use $CW_{min} = 16*K$. 

95
7.6.4 Convergence

The convergence speed of the TFCSMA protocol is the next aspect to evaluate. It is of paramount importance that TFCSMA quickly restore air time fairness whenever network conditions change. We consider the following scenario: At the beginning, there are two nodes, STA-1 and STA-2. STA-1 uses a data rate of 1 Mbps and has a \( \text{per} \) of 0. We conduct two different experiments. In the first experiment, the \( \text{per} \) for STA-2 is varied, while maintaining a transmission rate of 11 Mbps. The second test focuses on varying the transmission rate of STA-2, while maintaining a \( \text{per} \) of 0. Table 7.7 summarizes the experiment setup.

<table>
<thead>
<tr>
<th>Time (s)</th>
<th>( \text{per} ) Variation (STA-2)</th>
<th>Data Rate Variation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0-40</td>
<td>0</td>
<td>11</td>
</tr>
<tr>
<td>40-80</td>
<td>0.25</td>
<td>5.5</td>
</tr>
<tr>
<td>80-120</td>
<td>0.5</td>
<td>2.0</td>
</tr>
<tr>
<td>120-160</td>
<td>0.75</td>
<td>1.0</td>
</tr>
<tr>
<td>160-200</td>
<td>0</td>
<td>11.0</td>
</tr>
<tr>
<td>200-240</td>
<td>0.25</td>
<td>5.0</td>
</tr>
<tr>
<td>240-280</td>
<td>0.5</td>
<td>2.0</td>
</tr>
</tbody>
</table>

Table 7.7 Simulation Setup

It can be observed from figure Figure 7.17 that TFCSMA rapidly adapts to the changing network conditions. An average convergence period of less than 1.5 seconds was observed for both the experiments. Figure 7.18 shows the variation of the \( CW_{\text{min}} \) with time for the first experiment. We can observe that STA-1 steadily changes its window with varying error rates for STA-2. Whenever the error rate increases for STA-2, it experiences more backoffs. Therefore, STA-1 starts measuring a greater throughput than its required target (because of the relative greater transmission opportunities). In response, the MCWC of STA-1 increases the \( CW_{\text{min}} \), thereby indirectly helping STA-2 to achieve its target throughput.
Finally, we evaluate TFCSMA under unsaturated traffic load conditions. We again consider two flows in the network, one at 1 Mbps (STA-1) and the other at 11 Mbps (STA-2). Figure 7.19 plots the throughput of each flow with the offered load.

Several interesting observations are in order from the plot. For an offered load of upto 600 Kbps, both CWSP and TFCSMA provide an equivalent throughput. When the offered load varies from 600 to 800Kbps, TFCSMA exhibits a dip in the performance of the 1 Mbps flow. Since the STA-1 flow is also unsaturated, we would expect a greater throughput than 425 Kbps (the baseline throughput). However, TFCSMA computes its target with the assumption that all its contending nodes also operate under saturation. Therefore, the MCWC of STA-1 pushes down any incremental increase that would have been otherwise possible.
The benefits of TFCSMA surface again when the offered load for the STA-2 is increased beyond 2 Mbps. Again, the fixed contention window ratio in the case of CWSP begins favoring the 11 Mbps flow, thereby decreasing the performance of the 1 Mbps flow below its baseline value (425 Kbps). As usual, IEEE 802.11 continues to provide throughput fairness for both the flows, thereby severely impacting the performance of the 11 Mbps flow.
7.7 Conclusion and Future Directions

We have presented Time Fair Carrier Sense Medium Access (TFCSMA), an online extension of the IEEE 802.11 DCF that dynamically adapts the minimum contention window of contending nodes to provide air time fairness. This also mitigates the rate anomaly phenomenon observed in multi-rate networks.

Our method relies on the baseline property to dynamically derive a target throughput for a node based on its error and transmission data rates. It uses a simple control algorithm to adjust the minimum contention window of nodes to achieve this target. It is fully distributed and does not mandate the need of a centralized point of co-ordination. Moreover, the protocol is completely compatible with the IEEE 802.11 and does not require any changes in the basic access procedure. Our simulations results on OPNET show very encouraging results in terms of throughput and fairness.
Chapter 8

8. SARA: Stochastic Automata Rate Adaptation for IEEE 802.11 Networks

8.1 Introduction

In any typical wireless network, the quality of the received signal may vary considerably over a short period of time due to noise, interference, multi-path or Doppler effects (due to mobility). This can lead to bit errors and packet corruption at the receiver. As a consequence of this volatile nature of the channel, the performance of a wireless link is highly sensitive to several link layer parameters like packet size, modulation order (transmission rate), transmit power, and carrier sensitivity. Obviously, for different channel states, there exists a unique parameter set which provides optimal performance. Therefore, wireless systems often employ Link Layer Adaptation Algorithms which adjust the transmission parameter set appropriately to channel conditions in order to boost performance.

This chapter focuses on Rate Adaptation for IEEE 802.11 networks. Rate Adaptation is the process of dynamically adjusting the transmission rate to match the channel conditions, with the end objective of maximizing throughput. The present IEEE 802.11 compatible WLAN cards support several modulation types and data rates. For example, the IEEE 802.11g provides 12
different data rates ranging from 1Mbps to 54Mbps. The standard also provisions for Dynamic Rate Adaptation (DRA), allowing vendors to run propriety rate adaptation algorithms.

Existing rate adaptation algorithms can be broadly classified under two different categories: (a) Signal to Noise Ratio (SNR) measurement based, and (b) Statistical Count based. Examples of the former include RBAR [Holland02], Distributed Cooperative Rate Adaptation algorithm [Wang02a], OMAR [Wang04] etc, while ARF [Kamerman97], AARF [Lacage04], CARA [Kim06] etc are some popular examples of the latter.

Both of these approaches suffer from obvious shortcomings. While the SNR has been traditionally treated as an index for performance, recent studies indicate [Aguayo04] that both SNR and distance offer little predictive value in determining the packet loss characteristics for a wireless link. Experimental results attribute majority of loss rates to multi-path fading rather than attenuation or interference. Since fast fading can result in large fluctuations (upto 20dB) in the SNR over a few microseconds, any adaptation based on the received packet SNR measurement is bound to be inefficient.

As opposed to the SNR measurement mechanisms, Statistics Count based schemes require no explicit feedback from the receiver other than ACK or NACK (absence of ACK). Typically, recent feedback history (count of ACKs/NACKs) gathered for a particular rate is used for rate adjustments. Since these methods require minimal changes to the MAC protocol, they are widely used by different WLAN vendors. While the statistical count approach offers a very practical direction towards rate adaptation, it suffers from several inadequacies in their present embodiment. To begin with, most of these methods employ pre-defined thresholds to adjust the rate dynamically. For example, ARF increments the rate index if 10 consecutive ACKs are received. Likewise, the rate index is decremented on receiving two consecutive NACKs. This
one size fits all strategy restricts the versatility of these schemes over varied channel scenarios. Moreover the fixed thresholds are also responsible for the slow convergence of these methods specifically under non-stationary channel conditions [Qiao05][Lacage04][Kim05a].

This chapter proposes a rate adaptation mechanism which aims at providing both higher throughput and faster convergence. The novel contributions of this work are:

i. A comprehensive analysis of the impact of transmission rate on the performance of a wireless link. The findings highlight that no specific trends can be generalized for the behavior of performance versus rate.

ii. Stochastic Learning Automata\(^{14}\) (SLA) for link adaptation in IEEE 802.11 networks.

iii. SARA: Stochastic Automatic Rate Adaptation Algorithm, an online algorithm for dynamic rate adaptation to maximize throughput for IEEE 802.11. The algorithm is totally distributed in nature and completely compatible with the IEEE 802.11 DCF.

SARA treats rate adaptation as a learning automaton [Vasilakos94][Aranzulla02] that interacts with a random environment (wireless channel) and attempts to learn the optimal action (transmission rate) through an iterative learning process. SARA randomly chooses one of the offered data rates according to a probability vector, which at every instant maintains the selection probability of each rate. The outcome of the selected rate is qualified by means of an ACK/NACK from the receiver and is indicative of the stochastic characteristics of the chosen rate. The learning automaton utilizes this feedback to update its probability vector via a reward function. The goal of this process is to learn the rate which provides the maximum reward (throughput in our case) so that the automaton chooses this rate more frequently than other rates. Other salient features of SARA include a popular statistical method [Maxim] for estimating the packet error rate (PER) and also reducing the iterations during the learning process. As opposed

\(^{14}\) SLA is an existing machine learning technique used for adaptation in random environments.
to the previous work in this area, SARA is ideally suited for both stationary and non-stationary channel environments and is completely compatible with the existing IEEE 802.11 MAC standard. We compare the performance of SARA with ARF, AARF and other schemes. SARA outperforms the existing schemes in both stationary and non-stationary environments and displays fast convergence properties.

The rest of the chapter is organized as follows. Section 8.2 presents an overview of the IEEE 802.11 DCF, analyzes the effects of transmission rate on performance in wireless networks and discusses the objectives for a rate adaptation algorithm for IEEE 802.11 networks. The related work in the area of rate adaptation algorithms is discussed in Section 8.3. An overview of SLA and its relevance for wireless systems is provided in Section 8.4. Section 8.5 describes the functioning of SARA. We evaluate the performance and provide comparisons in Section 8.6. The chapter concludes in Section 8.7 with some interesting directions for future work.

8.2 Preliminaries

8.2.1 Effects of Transmission Rate on Performance

The performance of a wireless link is closely related to the transmission rate. In general, for a given channel condition, there exists a rate which maximizes the throughput. If a rate higher than the optimal is chosen, then the throughput decreases due to an increased number of retransmission attempts. This is because the packet error rate (PER) increases with rate if the SNR is fixed. On the other hand, a very conservative rate limits the throughput since the channel time gets poorly utilized. In fact, as demonstrated later, a rate lower than the optimal may even increase the PER. This sub-section presents a very simple analytical model to understand these aspects better. We divide the time axis into time slots of constant size \( \tau \). Let \( p_j \) be the probability
of a time slot in deep fade\textsuperscript{15} when rate \( R_i \) is being used. The time taken by a packet of length \( L \) for transmission is thus \( T = \frac{L}{R_i \tau} \) time slots. Now, for the entire packet to be received correctly, none of \( T \) slots should be in a deep fade. Hence we can define the packet success rate (PSR = 1-PER) as:

\[
PSR = (1 - p_f) \frac{L}{R_i \tau} \tag{8.1}
\]

Typically, we would like a node to operate at the maximum data rate possible. Increasing the transmission rate can have two contrasting effects:

- The PER may increase since a greater SNR is required to support the higher rate.
- The PER may reduce since the packet now occupies lesser channel time and hence its probability of hitting into a deep fade reduces.

Figure 8.1 uses equation (8.1) to plot the variation of PSR with the fade probability for different data rates. A slot size of the order of 0.001 is chosen. The contrasting trends are clearly observed for an example case of 12Mbps and 54Mbps. At a fade probability (\( P_f \)) of 0.2, the PSR for 12Mbps is 0.8 as

\textsuperscript{15} In multipath environments, there are regions of deep fade in the SNR at the receiver. This causes bursty errors and consequently packet errors.
compared to 0.95 for 54Mbps. This is because a packet transmitted at 54Mbps occupies 4.5 times less channel time and hence has much lesser chances of running into a deep fade. Realistically speaking, compared to 12Mbps, 54Mbps should have a greater value of $P_f$ because of the higher SNR requirements. It is observed that even for a $P_f$ ranging from 0.2 to 0.6 for 54Mbps, the PSR is greater as compared to a $P_f$ of 0.2 for 12Mbps. However, for a $P_f$ greater than 0.6, the PSR for 54Mbps starts falling below the PSR for 12Mbps at a $P_f$ of 0.2. Clearly, these observations indicate that no generalizations can be assumed about the relationship between the PSR and the data rate. Therefore, from the perspective of an adaptation algorithm, pre-conceived notions of reducing or increasing modulation when PSR drops or increases are incorrect. Figure 8.3 and Figure 8.4 provide simulation examples of above phenomena.

![Figure 8.3](image1.png)  
**Figure 8.3** PSR drops with increasing Rate

![Figure 8.4](image2.png)  
**Figure 8.4** PSR increases with Rate

### 8.2.2 Rate Adaptation Objectives for 802.11 Networks

A rate adaptation algorithm needs to continuously monitor the performance of the wireless link and consequently take actions to adapt correctly. A popular metric for measuring performance is the PSR (ratio of successful packets to total packets transmitted).

While intuitively there appears a direct relationship between the PSR and the throughput, the exact dependence is contingent upon the fairness criterion of the underlying MAC protocol. For example, the IEEE 802.11 DCF provides equal transmission opportunities to all the competing
nodes provided the back-off penalty is incurred only for collisions\textsuperscript{16}. Hence, each node tries to choose a rate that ensures maximum successes, thereby maximizing the net PSR. Here, maximizing the PSR is the optimal strategy for achieving the best possible throughput. On the other hand, if a time-fair MAC protocol is used, each node receives an equal portion of the channel time. Under these circumstances, maximizing PSR may not necessarily imply maximizing throughput. Consider for example that the channel has a PSR of 0.2 and 0.99 for 54Mbps and 1Mbps respectively. Clearly, if a maximize PSR approach is followed, 1Mbps will get chosen more frequently, lending a throughput of approximately 0.99Mbps. If instead, the rate adaptation algorithm decided to stick with 54Mbps, an approximate throughput of 10.2Mbps could have been possible. Therefore, under a time-fair MAC, the product of PSR and data-rate should be maximized instead of just the PSR.

Since the 802.11 MAC implicitly provides equal transmission opportunities, the throughput of each node becomes largely independent of its own data rate; rather it gets bounded by the lowest data rate peer [Heusse03]. As this phenomenon restricts the achievable benefits of any rate adaptation mechanism, researchers have proposed several time-fair extensions to IEEE 802.11 [Joshi06a][Kim05]. In this chapter, we present and analyze our generic adaptation mechanism which works equally well under both throughput and time fair 802.11 MAC models.

\textbf{8.3 Related Work}

Most of the existing rate adaptation mechanisms can be broadly classified as: (a) SNR measurement based schemes, (b) Statistics count based schemes, and (c) Hybrid Schemes

\textsuperscript{16} Equal transmission opportunity is only possible if there is no back-off penalty for packet losses due to link errors. Since the IEEE 802.11 DCF doesn’t distinguish between collisions and errors, researchers have proposed optimizations where no back-off is incurred after a successful RTS-CTS handshake. This roughly avoids back-off penalties for link errors.
(measurements and statistics based schemes). We discuss some of the most popular mechanisms proposed so far.

The Automatic Rate Fallback or ARF [Kamerman97] increments the rate index with 10 consecutive transmission successes (ACK threshold). The rate index gets decremented on 2 consecutive failures (NACK threshold). While such a simple approach definitely eases implementation, it restricts the versatility of ARF over varied channel scenarios. For a slowly varying channel, the retransmission attempts (and failures) increase with the ‘try higher rate’ paradigm of ARF. On the other hand, for a fast fading channel, ARF’s fixed thresholds (10 ACKs/2 NACKs) fail to track the ‘abrupt’ deep fades in the channel efficiently and thus the convergence is slow. Adaptive ARF (AARF) [Lacage04] attempts to reduce the unnecessary retransmissions by doubling the ACK threshold whenever a retransmission fails. Such a mechanism increases the time period between successive failures in a slow varying environment and thus improves throughput.

Both ARF and AARF attribute all packet losses to channel errors and ignore failures due to collisions. Therefore, both these protocols are liable to incorrectly decrement the rate even when channel conditions are good and losses are happening due to collisions. CARA [Kim06] addresses this problem by using the RTS/CTS mechanism to decouple collision from packet errors. A data frame is first transmitted without RTS/CTS support. If the transmission fails, the RTS/CTS exchange is activated for the next retransmission attempt, and the retransmission rate falls back if the transmission fails again. As a consequence, unnecessary rate decrements are completely avoided. The Loss-differentiating Adaptive Automatic Rate Fallback (LD-ARF) [Pang05] algorithm is another technique designed on the lines of CARA. This algorithm
performs the loss differentiation using the RTS/CTS mechanism followed by the data exchange and ACK/NAK.

The second class of rate adaptation schemes is based on the measurement of the SNR of the received packet. Vaidya et. al. [Holland02] proposed the first such rate adaptation mechanism: Receiver Based AutoRate (RBAR). In RBAR, receivers measure the channel quality using the request-to-send (RTS) message and select the rate on a per-packet basis during the RTS/CTS exchange, just prior to packet transmission. Sadeghi et. al propose OAR [Sadeghi04] which exploits multi-user diversity. OAR allows a node to transmit back to back packets successively with no contention for the medium when it experiences a good channel (indicated by the SNR).

While SNR based approaches provide faster convergence, they are not accurate metrics for channel state indication. In contrast, statistics based schemes improve performance, but suffer from slow convergence. Therefore, researchers proposed hybrid schemes which utilized a combination of SNR measurements and pre-defined statistical thresholds to boost performance. For example, HARF [Haratcherev04] adapts the rate by using the success/failure statistics along with the measured signal strength of the most recently received packet. Thus, this scheme increases the speed of rate adjustment and shows a good performance in the time varying channel conditions. Qiao et al [Qiao01] propose a dynamic programming approach towards rate adaptation. Their scheme is a table-driven approach which uses an offline table of PHY modes taking the data payload length, the receiver side SNR and the frame retry count into consideration. Each entry of the PHY mode table is chosen so as to maximize the expected effective good-put.

All the above approaches rely on either SNR measurements or utilize pre-defined thresholds. Recent studies indicate [Aguayo04] that there is little correlation between the SNR and the loss
characteristics of a wireless link. Hence, rate adaptation based on the received packet SNR measurement may be highly erroneous under certain scenarios (e.g., mobile multi-path channels). On the other hand, statistical count-based methods rely on pre-defined thresholds to adjust the rate dynamically. Since no preset relationship exists between rate and PSR, these schemes in their present form are also unsuited for varied channel conditions.

Motivated by these observations, we present SARA, a stochastic learning rate adaptation mechanism in this chapter. SARA neither requires any SNR measurements, nor does it assume any relationship between rate and PSR. It is inspired by *Stochastic Learning Automata* (SLA), a machine learning technique for adaptation in random environments. SLA algorithms have been widely studied in the design of control systems. Surprisingly, their application in link adaptation for 802.11 networks has not been adequately investigated. The only notable works in this domain include [Haleem05]. SARA shares a lot of common features with this scheme as both are inspired by SLA [Vasilakos94].

### 8.4 Stochastic Learning Automata

This section provides a brief overview of stochastic learning automata. For a detailed description, the interested reader may refer to [Vasilakos94].

#### 8.4.1 Introduction to Stochastic Learning Automata

A SLA is a finite state machine that operates in a random environment and makes a selection from a number of available options. It uses the feedback from the environment to judge whether its selection was good or bad and thereby learns the stochastic characteristics of the environment it is interacting with. SLA is an iterative learning process which assumes no predetermined relationships between the actions and responses. It is therefore ideally suited for learning actions in environments characterized by a large number of random variables. Since a wireless channel
is best described by a random process, SLA algorithms are attractive candidates for designing link adaptation mechanisms for such systems.

### 8.4.2 General Approach

The goal of a learning automaton is to learn the optimal action under the current conditions. The automaton chooses one of finite possible actions according to a probability vector which at every instant maintains the probability of choosing each action. The environment responds to the action of the automaton by a response (typically 0/1) which is indicative of the stochastic characteristics of the chosen action. Based on this feedback, a transition function computes the estimated mean rewards of every action. The automaton then increases the selection probability of the most favorable action and thus attempts to assign the maximum selection probability to the optimal action. As a result, the optimal action gets selected more often the other available actions. Figure 8.5 shows a block diagram of a typical automaton.

### 8.4.3 Non Stationary Environment and Learning Automata

SLA algorithms often maintain a running estimate of the mean reward of each action. Transition functions use this information to adjust the probability vector. While this method works perfect for stationary environments, it may work too conservatively under non-stationary environments and deteriorate performance. Under such environments, the stochastic characteristics of the actions change with time. Hence an action $k$ which was optimal earlier may not be the best for the current conditions. Unfortunately, during the learning process, the probability of selection of $k$ gets maximized such that other actions get selected very infrequently. We explain this phenomenon in the context of a rate adaptation algorithm. Say for the current channel conditions, 1Mbps provides the optimal performance. We will expect the automaton to learn this and select 1Mbps more often than other rates. Now consider that the
channel improves and 54Mbps can now provide the same PSR as 1Mbps. Obviously, we will desire the automaton to adapt to this higher rate. However, 1Mbps still provides a high PSR even under the new conditions. Since the selection probability of 54Mbps was reduced considerably while learning 1Mbps, it doesn’t get tried and the automaton gets stuck at 1Mbps. This is commonly referred to as being stuck in an absorbing state.

![Block diagram of a SLA](image)

**Figure 8.5** Block diagram of a SLA. S is the probability vector

To circumvent the above problem, SLA algorithms require special provisions to avoid such absorbing states. Aging and biasing are two possible solutions to alleviate these problems. Aging updates the mean rewards for every action stochastically, such that the probabilities of actions which haven’t been used for a long period get artificially incremented. SELA [Vasilakos94] uses a normally distributed random variable whose variance is proportional to time elapsed between the last use of that action to implement aging. Biasing is another technique that prevents the selection probability of an action from dropping beyond a minimum bias probability. This allows every action to get tried once in a while. Both these methods require a careful implementation to avoid the automaton from being either too aggressive or too conservative.
8.5 Stochastic Automata for Rate Adaptation (SARA)

8.5.1 Algorithm Outline

This section describes our proposed algorithm: Stochastic Automatic Rate Adaptation (SARA). SARA is an online, totally distributed rate adaptation algorithm inspired by SELA. Once a node selects a receiver $i$, SARA randomly selects a rate from a set of $k$ available rates, following a probability vector $p_i^t = [p_i^t(1), p_i^t(2), ..., p_i^t(k)]$ of receiver $i$. For simplicity, we assume a constant payload size used by every transmitting node and hence propose a two dimensional probability vector. In the description of the algorithm to follow, $t$ is the index of the sequence of packet trial. We denote the set of available rates as $R : x=1,2,\ldots,k$ (bits/sec). SARA introduces Probabilistic Packet Success Rate Estimator (PPSRE), which serves three key functions: (a) Accurately measure PSR, (b) Reduce the computations during the learning process, and (c) Provide Rapid Convergence under non-stationary environments. A detailed description of PPSRE is available in section 8.5.2.

PPSRE uses offline tables to update the packet success probability vector: $psr_i^t = [psr_i^t(1), psr_i^t(2), ..., psr_i^t(k)]$. Under a time fair 802.11, the throughput achieved with a rate $R_x$ is approximated by:

$$D_i^t = R(x) \ast (psr_i^t(x)) , i = 1,2,\ldots,K; \quad x = 1,\ldots,k$$

(8.2)

As discussed above, the performance objective of a rate adaptor depends on the fairness criterion. Therefore, SARA proposes two different rate objectives (equations 8.3 and 8.4) for a time and throughput fair 802.11 DCF respectively. Hence, a transmitting node is required to find the best rate $o_i^t$ for receiver $i$, i.e.,

---

A node may in fact use several packet sizes. In that case, we would require a three dimensional vector so that the selection probabilities can be maintained for different payloads.
\[ \mu' = \arg \max_x D_i'(x) \]  
\[ \mu_i\mu' = \arg \max_x psr_i'(x) \]  

In the remainder of the paper, we assume a time-fair extension of 802.11 and explain the algorithm accordingly. We also assume that packet errors are not penalized by means of doubling the contention window. To this end, SARA mandates the use of RTS-CTS since that helps to decouple collisions from link errors.

The rate selection probability vector is frequently updated by an iterative learning process such that the probability of assigning the best rate is maximized. Initially, the probabilities \( p_i'(x): x = 1, 2, \ldots, k \) are all assigned equal values of \( 1/k \). Then, the rate selection and transmission proceeds until every rate has been selected at least \( W \) number of times, following which SARA begins updating \( p_i'(x) \) at required intervals. The parameter \( W \) implements a sliding window, a technique that eliminates the slow adaptation problem of non-stationary systems [Vasilakos94].

Once a rate \( R(x) \) has been selected, SARA dynamically derives a performance target for this rate. The target is qualified in terms of the maximum throughput in \( D_i'(x) \). Next, the minimum required PSR at \( R(x) \) is calculated which can guarantee the target throughput. This is given by:

\[ tPSR_i = \frac{\max_{x \in R}(D_i'(x))}{R(x)} \]  

Clearly, the idea here is that \( R(x) \) must perform at least as good as the best possible rate seen so far. Using \( tPSR_i \), the PPSRE module now consults its offline tables \( T_{trial} \) to compute \( (N, n) \). Here, \( N \) is the number of trials the rate needs to be used and \( n \) denotes the minimum successes required in \( N \) trials such that \( R(x) \) can guarantee a success rate of \( tPSR_i \) with a confidence \( C \). At this instant, SARA enters a trial phase and starts using \( R(x) \) for subsequent transmissions.
Following each trial, the receiver responds by an ACK or NACK signal, indicating the successful reception or failure of the data packet respectively. This feedback is used to update the sliding window which maintains the history of successes and failures of the last $W$ trials of a rate. Now, the trial phase is terminated in the event of $n$ successes or $(N-n)$ packets errors. Using $N$ continuous trials helps in avoiding the existence of old and invalid feedback information in $W$. It also helps in minimizing the trials for a rate which is inefficient for the present channel conditions. Lastly, it avoids the invocation of the selection procedure after every transmission. Therefore it reduces the computations in the learning process and makes the automaton faster.

Following the trial phase, SARA begins the updating phase. The last $W$ ACK/NACK signals for every rate are used to update the probability vector towards the optimum distribution. SARA proceeds by counting the number of successes $r$ for every rate in their last $W$ respective trials. The offline table $T_{psr}$ in PPSRE pre-compute statistically the success probability for every combination $(r, W)$ with an associated confidence level $C$. Using these tables $psr_i^t$ is updated as:

$$psr_i^t(x) = T_{psr}(r)*C$$  \hspace{1cm} (8.6)

Next, the throughput vector $D_i^t$ is refreshed using equation 8.2. Now, the index $o_i^t$ of the best rate is computed, and the probability vector is updated using a pursuit algorithm [Vasilakos94] which drives the automaton in the direction of best reward. SARA uses a Discrete Pursuit Reward-Inaction (DPRI) pursuit algorithm as it has been shown to have excellent convergence properties. The probabilities $p_i^t(x), x \neq o_i^t$ are decremented by $\Delta$, another tunable parameter and the probability of the estimated best rate $o_i^t$ is increased by $\Delta*(k-1)$.

If the channel conditions remain stable for long periods, the algorithm may increase the selection probability of the best rate to almost unity, thereby rendering the selection probabilities of all other rates to close to zero. While this may be desirable under a stationary environment, it
may lead to ‘absorbing states’ for time-varying channels. The adaptor may get stuck at a lower rate and not try any better rates at all. To avoid this, SARA uses a biasing mechanism which maintains a minimum selection probability $b$ (a tunable parameter again) for all rates.

We observe a probability vector to have attained a steady state when the best rate’s (say $r_b$) selection probability has been maximized and all other rates are at the bias probability. Now, at the steady state, the choice of the value of $b$ determines how frequently rates other than $r_b$ get tried. Under stationary conditions, we would ideally desire that rates other than the optimal be very rarely selected. Therefore, a very small value of $b$ is desirable in this case. On the other hand, for a time-varying channel, a periodic assessment of different rates is necessary in order to avoid absorbing states. Under such circumstances, a reasonable value of $b$ is required for sufficient experimentation.

We denote the selection probability of $r_b$ at steady state by $p_{max}$. Now, we need to determine $b$ such that in $z$ trials, each and every rate gets tried at least once with a confidence level of $CL$. Hence the selection of $b$ should satisfy equations 8.7 and 8.8.

\[
(k-1)b + p_{max} = 1 \quad \text{(8.7)}
\]

\[
b = 1 - \left(1 - CL_{req}^{(k-1)z}\right)^{-1} \quad \text{(8.8)}
\]

The IEEE 802.11g allows $k = 12$ data rates. Table 8.1 shows the bias probability ($b$) such that all rates are tried within $z$ trials with an associated confidence ($CL$).
### Table 8.1 Bias Probability.

The results assume a fixed $p_{max}$ of 0.85. If the expected time for a packet transmission (including back-off) by a node is ETT, then we expect that within duration of $z \times ETT$ a node tries all the rates at least once. Let the vector $csr_i = [csr_i(1), csr_i(2), ..., csr_i(k)]$ denote a channel’s packet success rates for different transmission rates at a given instant $t$. Say the distribution of success rates within $csr_i$ change every $T$ seconds and the network has $n^n$ nodes. Under such conditions, we would desire all rates to be experimented at least once within a period $T$ for the algorithm to have any chances of adapting to changing conditions. This leaves us with an ETT budget of:

$$ETT \leq \frac{T}{(z \times n^n)} \quad (8.9)$$

Figure 8.6 plots the variation of the ETT budget with $n^n$. We observe that for all trials, the ETT budget is a reasonable 1.25 and 5msec for $T=5$ and 20 sec respectively. Hence, for the purpose of this paper, we choose a bias equivalent to 0.013. Actual implementations can use offline tables to dynamically adjust the bias based on the number of network nodes and past channel history.

#### 8.5.2 Probabilistic Packet Success Rate Estimator (PPSRE)

In this sub-section we study the working of PPSRE, a key ingredient in the design of SARA. PPSRE is a statistical estimation mechanism for measuring PSR and also determining the trial length of a selected rate. It uses two pre-computed tables: $T_{pnr}$ and $T_{trial}$ respectively to accomplish both these tasks.
The law of large numbers requires infinite trials in order to ascertain the actual probability of an event. Since infinite tests require infinite time, statistical confidence levels (SCL) are often used for estimating the probability based on a small set of measurements.

SCL estimates state that the actual probability of an event is better than some specified level. When applied to PSR estimation, the definition of SCL can be paraphrased as: if there are \( n \) successes in \( N \) independent packet transmission trials at rate index \( j \), then with what probability \( C \) can we ascertain that \( csr_j \) is at least \( psr_h \)? The relationship between these parameters can be expressed by [Maxim]:

\[
N = -\ln(1-C) + \frac{\ln\left(\sum_{m=0}^{n} (N \cdot (1 - psr_h))^m \right)}{(1 - psr_h)} \tag{8.10}
\]

\( T_{psr} \) is computed and maintained offline using equation 10 and a few approximations. Table 8.2 depicts a typical \( T_{psr} \). A window size \( W=20 \) and \( C = 0.9 \) is assumed. As mentioned above, \( T_{psr} \) is consulted during the updation phase to compute the revised \( psr_i^j \) vector.
We now discuss the computations behind $T_{trial}$. This offline table returns a performance benchmark $(N,n)$ which guarantees a PSR of $tPSR^i$ for a rate $R(x)$, if $n$ successes are met in $N$ packet trials. PPSRE uses an interesting method for determining the trial length $N$ for $R(x)$. It maintains an oldness vector $oN_i = [oN_i(1), oN_i(2), \ldots, oN_i(k)]$ where $oN_i(x)$ denotes the number of trials elapsed since $R(x)$ was last used. In order to compute $N$, $T_{trial}$ uses:

$$N = \left\lfloor W \left(1 - e^{-a \cdot oN_i(x)}\right) \right\rfloor$$ \hspace{1cm} (8.11)

$$a = -\frac{\ln(0.01)}{MaxTrials}$$ \hspace{1cm} (8.12)

The parameter $a$ is the value for which $N=0.99W$ when $oN_i(x)$ equals $MaxTrials$. The idea behind using an exponential function is that the trial length for recently used rates should be lesser as compared to rates which haven’t been used for longer periods. This is critical for rapid convergence for non-stationary channels. A rate with a high value of $oN_i(x)$ gets tried longer, and hence the algorithm has a better chance of learning whether conditions have actually improved or not. The success benchmark of $n$ ensures that the rate gets abandoned quickly if either the conditions are same or the rate is not as good as the best available seen so far. This is calculated as an approximation:

$$n = \left\lfloor N \cdot tPSR^i \right\rfloor$$ \hspace{1cm} (8.13)

<table>
<thead>
<tr>
<th>$n$</th>
<th>PSR</th>
<th>$N$</th>
<th>PSR</th>
<th>$n$</th>
<th>PSR</th>
<th>$n$</th>
<th>PSR</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0.0511</td>
<td>5</td>
<td>0.1411</td>
<td>10</td>
<td>0.3408</td>
<td>15</td>
<td>0.6474</td>
</tr>
<tr>
<td>1</td>
<td>0.1212</td>
<td>6</td>
<td>0.1511</td>
<td>11</td>
<td>0.4008</td>
<td>16</td>
<td>0.7114</td>
</tr>
<tr>
<td>2</td>
<td>0.1213</td>
<td>7</td>
<td>0.1632</td>
<td>12</td>
<td>0.4614</td>
<td>17</td>
<td>0.7771</td>
</tr>
<tr>
<td>3</td>
<td>0.1214</td>
<td>8</td>
<td>0.222</td>
<td>13</td>
<td>0.5226</td>
<td>18</td>
<td>0.8451</td>
</tr>
<tr>
<td>4</td>
<td>0.1312</td>
<td>9</td>
<td>0.2812</td>
<td>14</td>
<td>0.5846</td>
<td>19</td>
<td>0.9165</td>
</tr>
</tbody>
</table>

Table 8.2 Offline Table: $T_{psr}$. 

<table>
<thead>
<tr>
<th>PSR</th>
<th>$n$</th>
<th>PSR</th>
<th>$n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0511</td>
<td>5</td>
<td>0.1411</td>
<td>10</td>
</tr>
<tr>
<td>0.1212</td>
<td>6</td>
<td>0.1511</td>
<td>11</td>
</tr>
<tr>
<td>0.1213</td>
<td>7</td>
<td>0.1632</td>
<td>12</td>
</tr>
<tr>
<td>0.1214</td>
<td>8</td>
<td>0.222</td>
<td>13</td>
</tr>
<tr>
<td>0.1312</td>
<td>9</td>
<td>0.2812</td>
<td>14</td>
</tr>
</tbody>
</table>
Figure 8.7 plots the relationship between $oN_i^t(x)$ and $N$ for different MaxTrials. We observe that the variation of $N$ with $oN_i^t(x)$ is more gradual for higher MaxTrials. For example, at $oN_i^t(x)=20$, $N=11$ and 19 for MaxTrials = 100 and 25 respectively. We use MaxTrials = 100 for this paper. Table 8.3 shows a snapshot of the trial table computed for a $tPSR_i^t=0.5$.

<table>
<thead>
<tr>
<th>$N$</th>
<th>$n$</th>
<th>$N$</th>
<th>$n$</th>
<th>$N$</th>
<th>$n$</th>
<th>$N$</th>
<th>$n$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>6</td>
<td>3</td>
<td>11</td>
<td>5</td>
<td>16</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>1</td>
<td>7</td>
<td>3</td>
<td>12</td>
<td>6</td>
<td>17</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td>8</td>
<td>4</td>
<td>13</td>
<td>6</td>
<td>18</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>2</td>
<td>9</td>
<td>4</td>
<td>14</td>
<td>7</td>
<td>19</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td>10</td>
<td>5</td>
<td>15</td>
<td>7</td>
<td>20</td>
<td>10</td>
</tr>
</tbody>
</table>

Table 8.3 Snapshot Trial Table: $T_{trial}.PSR=0.5$

8.6 Performance Evaluation

The performance evaluation section is divided into two sub sections: (i) Stationary Channel Analysis and (ii) Non-Stationary Channel Analysis. In the first part we assume a fixed $c_{sr_t}$ distribution so that the optimal performance is known beforehand. The purpose of this part is to gain a better insight into the working of SARA. We compare SARA with ARF, AARF and with the basic SLA protocol presented in [Haleem05]. For the remainder of this section, we refer to this protocol as Basic Stochastic Learning Automata BSLAP. In the next part, we periodically change the $c_{sr_t}$ distribution and evaluate the performance of SARA under non-stationary conditions.

<table>
<thead>
<tr>
<th>Access Mechanism</th>
<th>RTS/CTS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sliding Window (W)</td>
<td>20</td>
</tr>
<tr>
<td>Probability Step (Δ)</td>
<td>0.005</td>
</tr>
<tr>
<td>Data Rate (k)</td>
<td>12</td>
</tr>
<tr>
<td>Biasing Probability</td>
<td>0.013</td>
</tr>
<tr>
<td>Max Trials</td>
<td>100</td>
</tr>
</tbody>
</table>

Table 8.4 SARA Parameters.

<table>
<thead>
<tr>
<th></th>
<th>SIFS</th>
</tr>
</thead>
<tbody>
<tr>
<td>DIFS</td>
<td>28 μ sec</td>
</tr>
<tr>
<td>σ</td>
<td>9 μ sec</td>
</tr>
<tr>
<td>ACK</td>
<td>14 bytes</td>
</tr>
<tr>
<td>MAC Header</td>
<td>28 bytes</td>
</tr>
<tr>
<td>PLCP Overhead</td>
<td>20 μ sec</td>
</tr>
</tbody>
</table>

Table 8.5 IEEE 802.11g.
To evaluate the performance of our proposed protocol, we have simulated a wireless link and implemented ARF, AARF, SLAP, and SARA in MATLAB. The wireless link is simulated by means of a \( c_{sr} \) vector. Since the purpose of these simulations is to evaluate the performance of different rate adaptation algorithms, we ignore the presence of collisions and limit all results to a single transmitter-receiver link (wireless link). We present results for a MAC using IEEE 802.11g parameters. Table 8.4 lists the parameters of SARA used in all the simulations. Table 8.5 summarizes the IEEE 802.11g parameters.

### 8.6.1 Stationary Channel

In this section, we present the results for a stationary channel scenario. We define a stationary channel as one where the packet success rates for different rates doesn’t change with time. Table 8.6 shows the PSR distribution for rates used in this study. The optimal rate under this scenario is 48 Mbps capable of a maximum throughput of 26 Mbps.

We compare the performance of SARA, SLAP, ARF, and AARF under these conditions. The simulations are run for 10 seconds and the transmitting node is assumed to be in a saturated condition. Figure 8.8 to Figure 8.11 plot the histogram of the rates selected by SARA, SLAP, ARF, and AARF respectively. We observe that SARA’s rate selection is highly precise followed by SLAP. Both ARF and AARF perform miserably since they rely on pre-defined thresholds for adjusting rates. As soon as success rates begin to fall, ARF and AARF turn highly conservative and stick to low data rates. A consequence of this effect is highlighted in the throughput plot in Figure 8.12. Figure 8.14 compares the computation complexity of SARA and SLAP. This is measured in terms of the number of times an algorithm actually selects a new rate for an upcoming transmission. It is observed that the calls in SLAP are more than 6 times as compared
to those for SARA. This is attributed to the fact that SARA doesn’t select a rate on each transmission trials. Figure 8.13 and Figure 8.15 show the overall PSR and rate selection timeline.

<table>
<thead>
<tr>
<th>CSR</th>
<th>0.95</th>
<th>0.9</th>
<th>0.85</th>
<th>0.8</th>
<th>0.75</th>
<th>0.7</th>
<th>0.7</th>
<th>0.65</th>
<th>0.6</th>
<th>0.55</th>
<th>0.45</th>
</tr>
</thead>
<tbody>
<tr>
<td>Index</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
<td>9</td>
<td>10</td>
<td>11</td>
</tr>
<tr>
<td>Rate</td>
<td>1</td>
<td>2</td>
<td>5.5</td>
<td>6</td>
<td>9</td>
<td>11</td>
<td>12</td>
<td>18</td>
<td>24</td>
<td>36</td>
<td>48</td>
</tr>
</tbody>
</table>

Table 8.6 Channel Success Rates for data rates. Rates in Mbps.

8.6.2 Stationary Channel-Random Scenario

In this study, we generate randomly 10 different channel scenarios and compare the performance of SARA, SLAP, ARF, and AARF. Table 8.7 shows the throughput achieved in
each of the simulation runs for all algorithms. We observe that SARA outperforms all existing schemes.

Figure 8.12  Throughput Timeline.

![Throughput Timeline](image)

Figure 8.13  PCR Timeline

![PCR Timeline](image)

Figure 8.14  Iterations Timeline.

![Iterations Timeline](image)

Figure 8.15  Rate Selection Timeline

![Rate Selection Timeline](image)

<table>
<thead>
<tr>
<th>Simulation</th>
<th>SARA</th>
<th>SLAP</th>
<th>ARF</th>
<th>AARF</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>13322.12</td>
<td>9761.22</td>
<td>1037.19</td>
<td>967.24</td>
</tr>
<tr>
<td>2</td>
<td>18855.34</td>
<td>12480.21</td>
<td>825.23</td>
<td>2388.20</td>
</tr>
<tr>
<td>3</td>
<td>16758.56</td>
<td>12264.36</td>
<td>8609.45</td>
<td>22813.5</td>
</tr>
<tr>
<td>4</td>
<td>11706.78</td>
<td>8100.78</td>
<td>3963.98</td>
<td>1232.84</td>
</tr>
<tr>
<td>5</td>
<td>9228.33</td>
<td>6576.44</td>
<td>8178.48</td>
<td>8630.06</td>
</tr>
<tr>
<td>6</td>
<td>10338.12</td>
<td>8264.65</td>
<td>2406.85</td>
<td>6757.41</td>
</tr>
<tr>
<td>7</td>
<td>15282.11</td>
<td>11240.31</td>
<td>2020.84</td>
<td>1409.17</td>
</tr>
<tr>
<td>8</td>
<td>16945.67</td>
<td>11211.17</td>
<td>273.75</td>
<td>272.63</td>
</tr>
<tr>
<td>9</td>
<td>16048.86</td>
<td>11001.18</td>
<td>17813.74</td>
<td>18402.34</td>
</tr>
<tr>
<td>10</td>
<td>9290.34</td>
<td>6279.19</td>
<td>457.57</td>
<td>1176.144</td>
</tr>
</tbody>
</table>

Table 8.7  Throughput Comparison for Random Channels.
8.6.3 Non-Stationary Channel Conditions

<table>
<thead>
<tr>
<th>CSR</th>
<th>1.0</th>
<th>0.9</th>
<th>0.8</th>
<th>0.7</th>
<th>0.6</th>
<th>0.5</th>
<th>0.4</th>
<th>0.3</th>
<th>0.2</th>
<th>0.01</th>
<th>0.01</th>
<th>0.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSR</td>
<td>0.01</td>
<td>0.01</td>
<td>0.01</td>
<td>0.2</td>
<td>0.3</td>
<td>0.4</td>
<td>0.5</td>
<td>0.6</td>
<td>0.7</td>
<td>0.8</td>
<td>0.9</td>
<td>1.0</td>
</tr>
<tr>
<td>Rate</td>
<td>1</td>
<td>2</td>
<td>5.5</td>
<td>6</td>
<td>9</td>
<td>11</td>
<td>12</td>
<td>18</td>
<td>24</td>
<td>36</td>
<td>48</td>
<td>54</td>
</tr>
</tbody>
</table>

Table 8.8 Channel Success Rates for data rates. Rates in Mbps

In this section, we present the results for a non-stationary channel scenario. We define a non-stationary channel as one where the packet success rates for different rates change with time. Table 8.8 shows the PSR distribution for rates used in this study. The success rates are changed every 3 seconds. Each simulation runs for 12 seconds, thereby triggering 3 changes in the environment. The optimal rate under the two scenarios is 11 and 54 Mbps.
Figure 8.16 to Figure 8.19 plot the histogram of rate selection for all the protocols. It is observed that SARA has the most balanced distribution. Again, ARF and AARF show poor performance, since they implicitly assume that rate should be reduced in the event of packet errors. As a result, they are always stuck on the lower rates. Figure 8.20 shows the throughput comparison. Once again SARA provides the maximum throughput consistently. Figure 8.21 and Figure 8.23 show the PCR and rate selection timeline. The computation complexity is compared in Figure 8.22. Again, SARA selects rates 5 times less as compared to SLAP.
8.7 Conclusion

In this chapter we introduced a new rate adaptation mechanism for IEEE 802.11 networks. We call our proposal SARA. SARA is inspired by Stochastic Learning Automata, an area of machine learning. It is ideally suited for a random wireless environment. We observe SARA improves performance under both stationary and non-stationary channel conditions. In comparison to existing SLA mechanisms, SARA reduces the computation complexity of rate adaptation, thereby making it an attractive choice for a practical implementation.
Chapter 9

9. Conclusion and Future Directions

In this PhD dissertation, we have proposed cross layer optimizations between the MAC and Routing; and MAC and PHY layers to boost the performance of wireless networks. The first part of the dissertation addresses the use of directional antenna systems for MANETs. We elaborated in detail the issues related to directional broadcasting and routing in MANETs. We have proposed two novel broadcast protocols for nodes equipped with directional antennas. We then proposed a Directional Routing Protocol (DRP) specifically tuned for an underlying directional MAC and PHY. DRP assumes considerable cross layer interaction between the MAC and the routing layer. Through extensive simulations, we have observed that DRP performs better than DSR (DSR with an omni-directional antenna) and DDSR (DSR with a directional antenna system) in almost scenarios.

The second part of the dissertation focuses on IEEE 802.11 multi-rate networks, where different nodes use different transmission rates. We first derive an analytical model to study the link delay characteristics of such networks. Next, we investigate the impact of both throughput and time fairness on the performance of wireless networks. We then propose an online, totally
distributed time fair extension of IEEE 802.11 (TFCSMA). Through extensive simulations, we observe that TFCSMA helps to mitigate the performance anomaly problem in multi-rate networks and thereby improves the aggregate throughput of these networks. Finally, we study the problem of rate adaptation in context to IEEE 802.11. We propose Stochastic Automata Rate Adaptation (SARA), a rate adaptation algorithm inspired by Stochastic Learning Automata (SLA), a machine learning technique. We evaluate the performance of SARA with existing rate adaptation algorithms and observe that SARA provides higher throughput and faster convergence under both stationary and non-stationary environments.

The insights obtained from this dissertation open several interesting research issues. We mention three key future directions of this work: The behavior of TFCSMA in multi-hop scenarios should be investigated. An undesirable consequence of air time fairness under multi-hop networks can be an increased congestion at layer 2. This may happen when low rate link follows a high rate hop in a route. New route metrics, which pick routes with similar data rate links, may then be required for satisfactory performance.

Another research direction can be the study of the effect of hidden terminals in multi-rate scenarios. Transmission range decreases with an increase in data rates. This can lead to several asymmetric links in a multi-rate network. Therefore, it is our intuition that rate disparity can lead to an increase in the number of hidden terminals in a network.

Lastly, stochastic learning automate can be applied towards solving the joint power and rate adaptation problem. SLA algorithms can be developed for adaptive fragmentation and rate adaptation. Efficient design of such algorithms can help in improving throughput under mobile multi-path scenarios.
Bibliography


[Li04] Q. Li, T. Li, T. Turletti, and Yang Xiao, “Saturation Throughput Analysis of Error Prone 802.11 Wireless Networks,”


