



**HELSINKI UNIVERSITY OF TECHNOLOGY**  
Department of Electrical and Communications Engineering  
Communications Laboratory

# **Performance Testing of Rate Adaptation Algorithms in WLANs**

**Muhammad Sohail Khan**

Masters Thesis submitted in partial fulfilment of the requirements for the  
Degree of Master of Science in Technology

Espoo, June 2008

Supervisor: Professor Riku Jäntti

Instructor: Lic. Sc (Tech). Kalle Ruttik

**Author:** Muhammad Sohail Khan

**Title:** Performance Testing of Rate Adaptation Algorithms in WLAN

**Date:** June 2008

**Number of pages:** 73

**Department:** Department of Electrical and Communications Engineering

**Professorship:** S-72 Communications Engineering

**Supervisor:** Professor Riku Jäntti

**Instructor:** Lic. Sc (Tech). Kalle Ruttik

**Abstract**

In this thesis we test the performance of Rate Adaptation mechanisms in 802.11 Wireless Local Area Networks. We consider three different Rate Adaptation Algorithms i.e. SampleRate, Onoe and AMRR (Adaptive Multi Rate Retry) available on Atheros Chipsets for which open source MadWiFi (Multi-band Atheros Driver for WiFi) drivers are available.

The performance testing is carried out in a realistic wireless environment realized by using a wireless channel simulator and a set of wireless nodes working in ad-hoc mode. A set of wireless channels proposed by 802.11 Task Group are used. The channels can be configured to provide identical environment for testing the different algorithms. We measure the throughput of different traffic workloads over each algorithm to evaluate and compare how the three algorithms adapt the PHY data rates to the varying wireless channel conditions.

The analysis of the test results helps in better understanding the Rate Adaptation mechanisms and the possible advantages and drawbacks of the different design approaches currently in practice.

**Keywords:** WLAN, Rate Adaptation , Wireless Channels, 802.11, MadWiFi, Ad-hoc Networks

# Acknowledgements

I would like to seize this opportunity to thank Kalle Ruttik, my thesis instructor, for helping me throughout the thesis work and pushing me till the end and being so patient while I was writing the thesis. I would also like to thank Professor Riku Jäntti for giving me the opportunity and his guidance throughout.

Special thanks to Victor for helping me in the Lab and giving his invaluable guidance in understanding the different testing equipment.

My greatest gratitude to my family who has always being a source of motivation throughout my life, especially my parents who are the guiding force of my existence. I am especially indebted to my wife and daughter for being so patient for the last two years. I am grateful to the Almighty for his blessings.

In the end my friends also deserve some credit for their ever cheerful presence; Aamir, Zeek, Majid, Naveed and Ijazz.

Espoo, June 2008

Muhammad Sohail Khan

# Table of Contents

|   |      |
|---|------|
| Abstract .....  | ii   |
| Acknowledgements .....  | iii  |
| Table of Contents.....  | iv   |
| List of Figures .....   | vi   |
| List of Tables .....  | vii  |
| List of Abbreviations.....  | viii |
| 1 CHAPTER 1 Introduction.....   | 1    |
| 1.1 Motivation .....  | 1    |
| 1.2 Problem Formulation and Contribution.....                               | 2    |
| 1.3 Thesis Outline.....   | 3    |
| 2 CHAPTER 2 IEEE 802.11 Wireless Networks.....                              | 4    |
| 2.1 Basic Architecture .....  | 4    |
| 2.2 802.11 Medium Access Control (MAC).....                                 | 6    |
| 2.2.1 Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)..... | 6    |
| 2.2.2 Basic Access Method .....   | 7    |
| 2.2.3 RTS/CTS Access Method .....   | 8    |
| 2.2.4 Error Recovery and Backoff.....                                       | 9    |
| 2.3 Physical Layer (PHY).....   | 10   |
| 2.3.1 IEEE 802.11 a/b/g .....   | 10   |
| 2.3.2 Performance of PHY .....  | 12   |
| 3 CHAPTER 3 Rate Adaptation in WLAN.....                                    | 14   |
| 3.1 Rate Adaptation .....   | 14   |
| 3.2 Classification of Rate Adaptation Mechanisms.....                       | 15   |
| 3.2.1 Error/Loss based Adaptation .....                                     | 15   |
| 3.2.2 Through-Put Based adaptation .....                                    | 17   |
| 3.2.3 SNR based adaptation .....  | 17   |
| 3.2.4 Hybrid Automatic Rate Control .....                                   | 18   |
| 3.3 Shortcomings of Current Design Approach.....                            | 18   |
| 3.3.1 Misinterpretation of Frame Loss .....                                 | 18   |
| 3.3.2 SNR value as Channel Estimator .....                                  | 19   |
| 4 CHAPTER 4 Algorithms Under Test.....                                      | 20   |
| 4.1 Onoe.....   | 20   |
| 4.2 SampleRate .....  | 22   |
| 4.3 AMRR .....  | 24   |
| 5 CHAPTER 5 Experimental Setup .....  | 26   |
| 5.1 Performance Metrics and Measured Parameters .....                       | 27   |
| 5.2 Experimental Test Bed Hardware Equipment .....                          | 27   |
| 5.2.1 Wireless Nodes.....   | 28   |
| 5.2.2 Wireless LAN Adaptors .....   | 28   |
| 5.2.3 Channel Simulator.....  | 28   |
| 5.2.4 Signal Generator.....   | 31   |
| 5.3 Traffic Generation Tool.....  | 31   |
| 5.4 Experimental Setup Calibration.....                                     | 32   |
| 5.5 Connection Setup.....   | 33   |
| 5.6 Channel Models.....   | 34   |
| 6 CHAPTER 6 Performance Evaluation Test Scenarios.....                      | 36   |

|       |   |    |
|-------|---|----|
| 6.1   | Test Setup Configuration.....   | 37 |
| 6.2   | Wireless Channel Simulation Parameters.....                                 | 37 |
| 6.3   | Test Case.....  | 38 |
| 6.3.1 | CBR/UDP Traffic.....  | 38 |
| 6.3.2 | TCP Traffic .....   | 42 |
| 7     | CHAPTER 7 Conclusion and Future Work .....                                  | 47 |
| 7.1   | Conclusion .....  | 47 |
| 7.2   | Future Work.....  | 48 |
|       | BIBLIOGRAPHY .....  | 49 |
|       | APPENDIX A.....   | 52 |
| A.1   | Channel Model A for NLOS conditions with 50 ns rms delay spread.....        | 52 |
| A.2   | Channel Model B for NLOS conditions with 100 ns rms delay spread .....      | 53 |
| A.3   | Channel Model C for NLOS conditions with 150 ns rms delay spread .....      | 54 |
| A.4   | Channel Model D for LOS conditions with 140 ns rms delay spread.....        | 55 |
|       | APPENDIX B .....  | 56 |
| B.1   | Average Throughput vs. Pathloss for UDP traffic with Channel Model A .....  | 56 |
| B.2   | Average Throughput vs. Pathloss for UDP traffic with Channel Model B.....   | 56 |
| B.3   | Average Throughput vs. Pathloss for UDP traffic with Channel Model C.....   | 57 |
| B.4   | Average Throughput vs. Pathloss for UDP traffic across Channel Model D..... | 57 |
| B.5:  | Average Throughput vs. Pathloss for TCP traffic with Channel Model A .....  | 58 |
| B.6   | Average Throughput vs. Pathloss for TCP traffic with Channel Model B .....  | 58 |
| B.7   | Average Throughput vs. Pathloss for TCP traffic with Channel Model C .....  | 59 |
| B.8   | Average Throughput vs. Pathloss for TCP traffic with Channel Model D .....  | 59 |
|       | APPENDIX C.....   | 60 |
| C.1   | MadWiFi Driver Installation on Linux Family of OS.....                      | 60 |
| C.2   | Changing the Rate Adaptation Algorithm.....                                 | 61 |
|       | APPENDIX D .....  | 62 |
| D.1:  | Experimental Setup calibration via Network Analyzer.....                    | 62 |

# List of Figures

|   |    |
|---|----|
| Figure 2-1: Ad-hoc Based Network Configuration.....   | 5  |
| Figure 2-2: Infrastructure Mode.....  | 5  |
| Figure 2-3: Basic Access Method [1].....  | 8  |
| Figure 2-4 : RTS/CTS Access Method and NAV setting [1].....   | 9  |
| Figure 2-5: Theoretical BER vs. SNR for several modulation schemes in AWGN [3].....   | 13 |
| Figure 5-1: PROPSim C2 Channel Simulator.....   | 29 |
| Figure 5-2: SMJ 100A Vector Signal Generator.....   | 31 |
| Figure 5-3: Calibration of Connection Setup.....  | 33 |
| Figure 5-4: Testbed Setup.....  | 34 |
| Figure 6-1: Average Throughput Vs Pathloss value for UDP Traffic across Channel Model A (50ns rms delay spread NLOS).....   | 39 |
| Figure 6-2: Average Throughput Vs Pathloss value for UDP Traffic across Channel Model B (100 ns rms delay spread NLOS)..... | 39 |
| Figure 6-3: Average Throughput Vs Pathloss value for UDP Traffic across Channel Model C (150 ns rms delay spread NLOS)..... | 40 |
| Figure 6-4: Average Throughput Vs Pathloss value for UDP Traffic across Channel Model D (140 ns rms delay spread LOS).....  | 40 |
| Figure 6-5: Loss Ratio Vs Pathloss value for UDP Traffic across Channel Model C (150 ns rms delay spread NLOS).....         | 42 |
| Figure 6-6: Average Throughput Vs Pathloss value for TCP Traffic across Channel Model A (50 ns rms delay spread NLOS).....  | 43 |
| Figure 6-7: Average Throughput Vs Pathloss value for TCP Traffic across Channel Model B (100 ns rms delay spread NLOS)..... | 44 |
| Figure 6-8: Average Throughput Vs Pathloss value for TCP Traffic across Channel Model C (150 ns rms delay spread NLOS)..... | 44 |
| Figure 6-9: Average Throughput Vs Pathloss value for TCP Traffic across Channel Model D (140 ns rms delay spread LOS).....  | 45 |

# List of Tables

|  |    |
|--|----|
| Table 2-1: Modulation and Coding rate for OFDM 802.11a/g ..... | 12 |
| Table 6-1: WLAN configuration Parameters .....                 | 37 |

# List of Abbreviations

|                   |   |
|-------------------|---|
| AARF              | Adaptive Auto Rate Fallback                     |
| ACK               | ACKnowledgement                                 |
| AMRR              | Adaptive Multi Rate Retry                       |
| AP                | Access Point                                    |
| ARF               | Auto Rate Fallback                              |
| AWGN              | Additive White Gaussian Noise                   |
| BER               | Bit Error Rate                                  |
| BSA               | Basic Service Area                              |
| BEB               | Binary Exponential Backoff                      |
| BPSK              | Binary Phase Shift Keying                       |
| BSS               | Basic Service Set                               |
| CA                | Collision Avoidance                             |
| CARA              | Collision Aware Rate Adaptation                 |
| CBR               | Constant Bit Rate                               |
| CCA               | Clear Channel Assesment                         |
| CCK               | Complementary Code Keying                       |
| CSMA              | Carrier Sense Medium Access                     |
| CSMA/CA           | Carrier Sense Medium Access/Collision Avoidance |
| CTS               | Clear To Send                                   |
| CW                | Contention Window size                          |
| CW <sub>max</sub> | Contention Window maximum                       |
| CW <sub>min</sub> | Contention Window minimum                       |
| DBPSK             | Differential Binary Phase Shift Keying          |



|          |   |
|----------|---|
| DCF      | Distributed Coordination Function                 |
| DCS      | Digital Cellular Systems                          |
| DIFS     | Distributed Inter Frame Space                     |
| DQPSK    | Differential Quadrature Phase Shift Keying        |
| DS       | Distribution System                               |
| DSSS     | Direct Sequence Spread Spectrum                   |
| ESS      | Extended Service Set                              |
| FEC      | Forward Error Correction                          |
| FHSS     | Frequency Hopping Spread Spectrum                 |
| GSM      | Global System for Mobile Communication            |
| IEEE     | Institute of Electrical and Electronics Engineers |
| IBSS     | Independent Basic Service Set                     |
| IFS      | Inter Frame Space                                 |
| IMT-2000 | International Mobile Telecommunication- 2000      |
| IP       | Internet Protocol                                 |
| IR       | Infra Red   |
| ISM      | Industrial Scientific and Medical                 |
| LOS      | Line Of Sight                                     |
| MAC      | Medium Access Control                             |
| MadWiFi  | Multiband Atheros Driver for WiFi                 |
| NLOS     | Non Line Of Sight                                 |
| NAV      | Network Allocation Vector                         |
| OFDM     | Orthogonal Frequency Division Multiplexing        |
| OAR      | Opportunistic Auto Rate                           |
| PCF      | Point Coordination Function                       |

|        |   |
|--------|---|
| PCS    | Personal Communication Service                  |
| PHY    | PHYSical  |
| PIFS   | PCF Inter Frame Space                           |
| PLCP   | Physical Layer Convergence Procedure            |
| PMD    | Physical Medium dependent                       |
| QAM    | Quadrature Amplitude Modulation                 |
| RBAR   | Receiver Based Auto Rate                        |
| RF     | Radio Frequency                                 |
| RFLO   | Radio Frequency Local Oscillator                |
| RSSI   | Received Signal Strength Indication             |
| RTS    | Ready To Send                                   |
| SIFS   | Short Inter Frame Space                         |
| SLRC   | Station Long Retry Count                        |
| SNR    | Signal to Noise Ratio                           |
| SSI    | Signal Strength Indicator                       |
| SSRC   | Station Short Retry Count                       |
| STA    | STAtion   |
| TCP    | Transmission Control Protocol                   |
| TCP/IP | Transmission Control Protocol/Internet Protocol |
| TG     | Task Group                                      |
| TGn    | Task Group n                                    |
| UDP    | User Datagram Protocol                          |
| UDP/IP | User Datagram Protocol/Internet Protocol        |
| UN-II  | Unlicensed National Information Infrastructure  |
| WCDMA  | Wideband Code Division Multiple Access          |

|        |                                    |
|--------|------------------------------------|
| WiFi   | Wireless Fidelity                  |
| WLAN   | Wireless Local Area Network        |
| 16-QAM | 16-Quadrature Amplitude Modulation |
| 64-QAM | 64-Quadrature Amplitude Modulation |

# CHAPTER 1

## Introduction

### 1.1 Motivation

Demand for Wireless Networks has increased manifolds in the past several years, evolving from novelty to necessity. The main advantage of wireless networks is its ability to provide access to the internet on the go as opposed to its wired counterpart. The demand has primarily being driven by users connecting their laptops computers to the networks at work and the internet at home, coffee shops, airports, hospitals, hotels and other public gathering places. The pervasiveness of the Wireless Local Area Networks is making it possible to extend the technology beyond the personal computers and into consumer electronic applications like internet telephony, music and multimedia streaming, gaming and other on demand services [22]. However, these media rich applications demands increased reliability and higher data rates.

In recent years Wireless Local Area Networks (WLANs) based on IEEE 802.11 standard has become the most popular and widely accepted technology for broadband wireless access. Its popularity is due some characteristics such as low cost, easy deployment and availability of high data rates. The current 802.11 specifications

provide multiple data rates. The 802.11b standard specifies data rates up to 11 Mbps and the 802.11a/g provides data rates up to 54 Mbps.

## 1.2 Problem Formulation and Contribution

One of the key challenges to 802.11 communities is how to exploit the available data rates in an adaptive manner to maximize the system performance. The mechanism which adjusts the data rate in response to the varying wireless channel conditions is called rate adaptation. To exploit the multi rate capability of 802.11 the rate adaptation mechanism must select the best data rate and dynamically adapt its decision according to the varying wireless channel quality. The 802.11 standard does not specify any rate adaptation mechanism and it is up to the manufacturer of the Wireless LAN adaptors to implement its own. In recent years a number of algorithms have been proposed for rate adaptation [3][4][7][8][9][10][11][12]. This thesis evaluates the performance of three Rate Adaptation Algorithms i.e. SampleRate [3], Onoe [3][2] and AMRR [4] based on Atheros chipsets for which an open source MadWiFi [2] driver is available. Many recent studies have performed experimental evaluation of Rate Adaptation Algorithms [3][4][14][17][23]. Most of these experiments have been performed in uncontrolled wireless environments which lack the capability of reproducing and configuring wireless channels in a repetitive manner to provide the same conditions for comparison of the three Rate Adaptation Algorithms.

The main objective of the thesis is to provide a realistic wireless channel environment which can be reproduced and configured in a repetitive manner for the comparison of the performance of the Rate Adaptation Algorithms. It is realized by establishing a WLAN 802.11 test bed using a channel simulator. We will use different channel environments based on channel models proposed by 802.11 Task Group n (TGn) for our evaluation. The idea is to test how the three Rate Adaptation Algorithms adapts to the varying channel conditions by observing the application level throughput of traffic generated.

## 1.3 Thesis Outline

The rest of the Thesis is organized in the following way:

Chapter 2: This chapter gives an overview of the WLAN 802.11 standard with emphasis on the MAC protocol and PHY layer in relation with rate adaptation mechanisms.

Chapter 3: This chapter gives basic concepts about the rate adaptation mechanism and the work that has been previously done in published literature.

Chapter 4: This chapter explains the three Rate Adaptation Algorithms which are being tested.

Chapter 5: This chapter deals with the experimental setup being established for testing. It starts with the description of the software and hardware employed and then setting up the final test bed. It also explains the channel models being used.

Chapter 6: this chapter explains the actual test cases that are run to evaluate the performance of Rate Adaptation Algorithms. It also briefly analyses the results.

Chapter 7: This chapter gives the final conclusion about the test results and some guidelines for future work.

# CHAPTER 2

## IEEE 802.11 Wireless Networks

This chapter gives an overview of the IEEE 802.11 Wireless Networks. It covers the Physical Layer (PHY) and the Medium Access Control (MAC) layer standards most relevant to the thesis work.

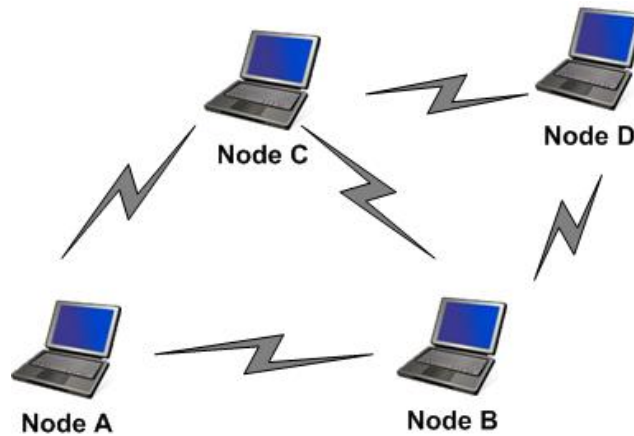
### 2.1 Basic Architecture

The IEEE 802.11 Wireless Local Area Networks (WLAN) consists of two types of basic architectures; the Infrastructure based and Infrastructure-less or Ad-hoc mode. The Basic Service Set (BSS) is the building block of the 802.11 WLAN. The coverage area of the BSS is referred to as the Basic Service Area (BSA). A station (STA) that is a member of the BSS within the BSA may communicate with other members of the BSS.

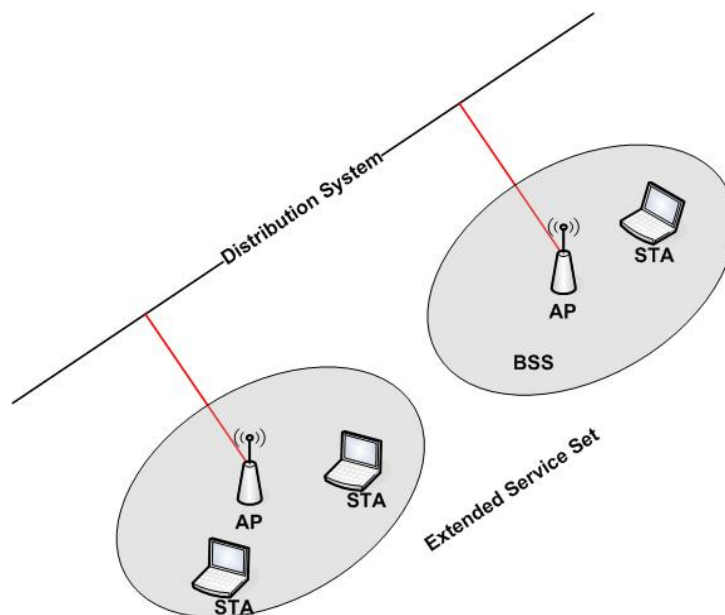
An Ad-hoc network mode deliberates on the grouping of stations into a BSS without the need for any infrastructure network. When operating in this mode stations are said to form an Independent Basic Service Set (IBSS) and the station directly communicate with each other. This type of WLAN is only deployed in dynamic

environments where people need to be temporarily interconnected without any infrastructure.

In the infrastructure mode the stations communicate through an Access Point (AP) as a common point of contact rather than communicating directly. The Access Point creates a range extension and obtains other services like Internet access from other Wired or Wireless Networks. Range extension can be realized through connection of one BSS with another BSS through a Distribution System (DS), which is referred to as Extended Service Set (ESS). Figure 2-1 and Figure 2-2 shows the Ad-hoc and the Infrastructure based network configurations.



**Figure 2-1:** Ad-hoc Based Network Configuration



**Figure 2-2:** Infrastructure Mode



## **2.2 802.11 Medium Access Control (MAC)**

Medium Access Control layer is mainly responsible for controlling the transmission of the data over the Physical Media. Its discussion is important because the Rate Adaptation Algorithms mainly uses the MAC for its operation.

The 802.11 standard specifies two different MAC schemes; the contention based, mandatory Distributed Coordination Function (DCF) and the contention free, optional Point Coordination Function (PCF). The DCF uses the distributed coordination for the access where all stations use the same method to control its access to the Physical media without the presence of any central coordinator. The PCF uses a Point Coordinator that polls stations, giving them the opportunity to transmit. This limits its use and can only be used with a dedicated Access Point. We will only discuss the DCF as the experimental evaluation is conducted in Ad-hoc mode.

### **2.2.1 Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA)**

The DCF is based on the Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) scheme. When a station is ready to transmit a frame, it checks the status of the media. If the media is busy, it waits until the end of the on-going transmission. This part makes the DCF the carrier sense Multiple access (CSMA) protocol. When the media becomes idle, instead of transmitting immediately the station selects a random backoff interval in order to reduce the collision probability. This makes the DCF a collision avoidance protocol.

Even with random backoff a transmitted frame may still collide with other frames when two or more stations finish the backoff simultaneously. Such frame collisions cannot be eliminated due to contention nature of DCF and the problem becomes worse as the number of contending stations increases. Besides collision, a frame transmission failure may be caused by wireless channel errors and the distinction between the two is an important parameter in the design of the Rate Adaptation Algorithms.

Carrier sensing can be performed on both the physical (PHY) and MAC layers. On the Physical layer carrier sensing is done by detecting any channel activity by other stations often called as Clear Channel Assessment (CCA). The CCA busy is declared when an energy level measured at the antenna front-end is above a threshold. In addition to the physical carrier sensing, virtual carrier sensing is achieved by using time fields in the frames, which indicate to the other stations the duration of the current transmission. All stations that hear the frame update their Network Allocation Vector (NAV) field based on the value of the duration in the received frame and refrain from any transmission for the duration. The stations use a countdown timer to count down from current NAV value to zero and assume the medium busy till NAV reaches zero.

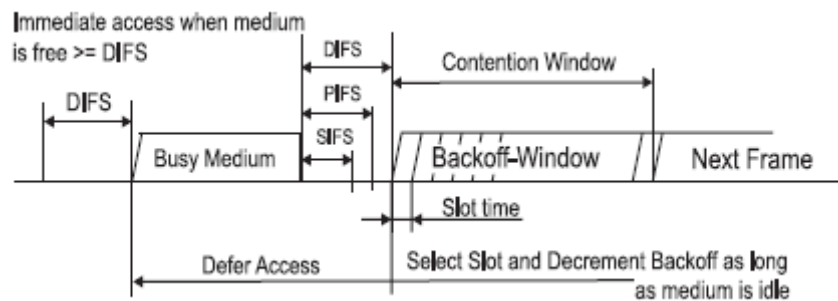
Under the DCF, data frames are transmitted via two methods. The essential method used in DCF is the basic access method. The 802.11 standard also provides an alternative way of transmitting data frames called the RTS/CTS method. The two access methods are discussed in following sections.

### **2.2.2 Basic Access Method**

In 802.11, priority access to the wireless medium is controlled by the use of the Inter Frame Space (IFS) time period between the transmissions of frames. The IFS defines the minimal time for a station to elapse after the end of a frame, before it may start transmitting a certain type of frame. Three different types of IFS intervals have been specified to provide various priority levels for access to the wireless medium; Short IFS (SIFS), PCF- IFS (PIFS) and the DCF-IFS (DIFS). The SIFS is the smallest followed by PIFS and then DIFS. After SIFS only acknowledgments, CTS and data frames may be sent. The use of PIFS and DIFS is to separate the PCF and DCF modes.

The station may proceed with its transmission if the medium is sensed to be idle for an interval larger than the DIFS. If the medium is busy, the station defers until a DIFS time has elapsed and then generate a random backoff period before transmitting. The backoff timer counter is decreased as long as the wireless medium is sensed idle,

frozen when sensed busy and resumed when the medium is sensed idle again for more than a DIFS. A station can initiate a transmission when the backoff timer reaches zero. After receiving correctly a frame in the destination station, an immediate positive acknowledgement (ACK) is sent to confirm the successful reception of the frame transmission after a time interval equal SIFS. Since the SIFS interval is shorter than the DIFS, the station sending an ACK attempts transmission before stations attempting to send data and hence takes priority. If the source station does not receive an ACK, the data frame is assumed to have been lost and a retransmission is scheduled. The basic access method is illustrated in Figure 2-3.

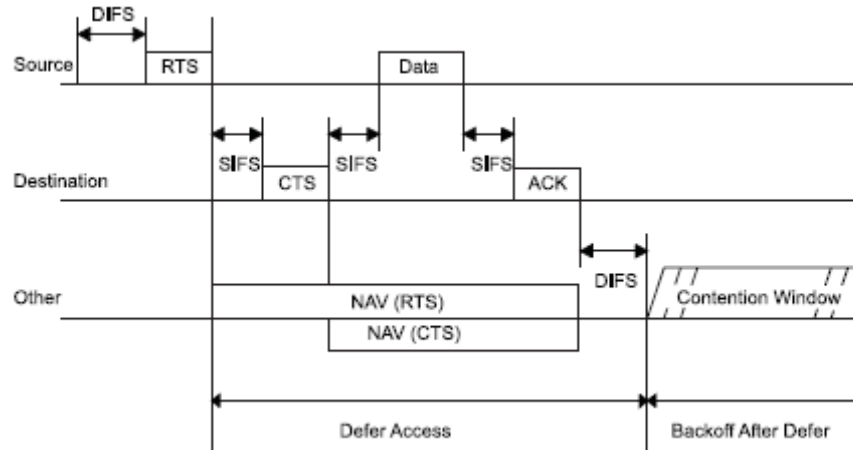


**Figure 2-3:** Basic Access Method [1]

### 2.2.3 RTS/CTS Access Method

In 802.11, DCF also provides an optional way of transmitting data frames that involve transmission of special short RTS/CTS frames prior to data frames. RTS frame is transmitted by a station that wants to transmit. After receiving the RTS frame the receiver send a CTS frame after SIFS interval. The source is allowed to transmit actual data frame after a SIFS interval if it receives the CTS correctly. The RTS/CTS access method is illustrated from Figure 2-4. It can be seen that other stations are capable of updating the NAV based on the RTS from source or CTS from destination. The RTS/CTS access method is used to address the Hidden Node problem. The problem occurs when two stations not in the carrier sensing range of each other try to transmit to a station, which is in the carrier sensing range of both the stations, at the same time. The RTS/CTS method is also used in cases where the MAC frame length is greater than an RTS\_Threshold value. The RTS\_threshold value is manageable

parameter. An RTS frame loss in case of collision incurs less bandwidth wastage as compared to a large MAC data frame.



**Figure 2-4 :** RTS/CTS Access Method and NAV setting [1]

#### 2.2.4 Error Recovery and Backoff

Every frame sent should be positively acknowledged. The receiver must send an acknowledgement of the frame it receives but it is up to the sender to detect error and recover from it. The sender assumes an error or loss of packet when it doesn't get an acknowledgment back. Errors may occur due to collision with another transmission or due to the wireless link conditions. Error recovery is attempted by retrying transmissions for the frame that the initiating station infers has failed. Retries shall continue, for each frame, until the transmission is successful or until the relevant retry limit is reached, whichever occurs first. Most of the current Rate Adaptation Algorithms use the statistics of retries count, failed transmission as the basic design parameter. Every station maintains a Station Short Retry Count (SSRC) as well as Station Long Retry Count (SLRC). The SSRC is associated with MAC frames shorter than or equal to  $RTS\_threshold$  and SLRC is associated with MAC frames larger than  $RTS\_threshold$ . The initial values are set to zero and incremented for every retransmission. Retries for failed transmission attempts continue until SSRC reaches a *ShortRetryLimit* and SLRC reaches *LongRetryLimit*. When either of these limits is reached, retry attempts are stopped and the frame discarded.

In order to minimize the probability of collisions, a random backoff mechanism is used to randomize moments at which stations are trying to transmit. This contention resolution technique is called binary exponential backoff (BEB). After DIFS time is over the contention window ( $CW$ ) starts. The station calculates the random backoff time. The contention window is divided into time slots and the length of time slots depends on the physical layer implementation. The backoff time is an integer multiple of the time slot and is chosen uniformly in the interval  $(0, CW-1)$ . The value of the  $CW$  depends on the number of failed transmissions of a frame. At the first transmission attempt the  $CW$  value is set to  $CW_{min}$  which is called minimum contention window. After each retransmission,  $CW$  is doubled upto a maximum value  $CW_{max}$ . The values of the  $CW_{min}$  and  $CW_{max}$  depend on the physical layer adopted. Once the contention window reaches its maximum size it is only reset if the frame is discarded or frame is successfully transmitted.

## 2.3 Physical Layer (PHY)

The PHY is the interface between the MAC and the wireless medium, which transmits and receives data frames over the shared wireless media. The 802.11 physical layer has been divided into two sublayers. The Physical Layer Convergence Procedure (PLCP) sublayer and the Physical Medium Dependent (PMD) sublayer. The PLCP sublayer is responsible for the frame exchange between the MAC and PHY. The PMD sublayer has the responsibility of converting the frames received from the PLCP into bits for transmission over the physical media. The physical layer also has an additional function called the Clear Channel Assessment (CCA) function, which is the physical carrier sensing.

### 2.3.1 IEEE 802.11 a/b/g

IEEE 802.11 specifications define the MAC and PHY layers. The original 802.11 standard specifies a common MAC and three PHY layers [1] giving data rates of 1Mbps and 2 Mbps using the Direct sequence spread spectrum (DSSS) , Frequency Hopping Spread Spectrum (FHSS) and Infrared (IR) techniques. The 802.11 standard has subsequently been expanded with the MAC layer largely unchanged and three

PHY layer specifications added. Here, we will briefly describe the three PHY layers standards 802.11a/b/g.

802.11a PHY facilitates data rates up to 54 Mbps in the 5 GHz Unlicensed National Information Infrastructure (UN-II) band. Orthogonal Frequency Division Multiplexing (OFDM) has been selected as the modulation scheme which enables higher data rates. The basic principle of the OFDM is to divide the high-speed binary signal to be transmitted over a number of low data rate subcarriers. There are a total of 52 subcarriers out of which 48 subcarriers carry actual data and 4 subcarriers are pilots. 802.11a PHY gives 8 PHY modes with different modulation schemes and coding rates, making the idea of rate adaptation feasible and important. Subcarriers are modulated using DBPSK, DQPSK, 16-QAM and 64-QAM modulation schemes. Forward Error Correction (FEC) is performed by bit interleaving and rate 1/2, 2/3 and 3/4 convolutional coding. The support of data rates of 6, 12 and 24 Mbps are mandatory in the 802.11a PHY. Optional rates include 9, 18, 36, 48 and 54 Mbps.

802.11b is an extension of the original DSSS PHY. It operates in the 2.4 GHz Industrial, Scientific and Medical (ISM) band, providing 5.5 and 11 Mbps PHY rates in addition to the 1 and 2 Mbps rates supported by the original DSSS PHY. The 1 and 2 Mbps data rates are based on the BPSK and QPSK modulation schemes respectively. Both are encoded using DSSS based on 11-bit Barker code. The higher data rates of 5.5 and 11 Mbps are achieved using the Complementary Code Keying (CCK) modulation scheme. The 4 PHY modes provided by 802.11b makes the idea of rate adaptation feasible however with fewer options as compared to 802.11a and 802.11g.

While 802.11b uses only DSSS, 802.11g uses DSSS, OFDM or both at the 2.4 GHz ISM band to provide high data rates up to 54 Mbps. Thus, the modulation schemes used in 802.11g are OFDM for data rates of 6, 9, 12, 18, 24, 36, 48 and 54 Mbps and revert to CCK, like in 802.11b, for 5.5 and 11 Mbps and DBPSK/DQPSK for 1 and 2 Mbps. 802.11g is a promising standard due to the use of the same OFDM technology as employed in 802.11a, plus backward compatibility with 802.11b devices. Therefore 802.11g is an alternative for 802.11b users to upgrade their wireless Networks.

The table below shows the data rates, modulation, and coding rates used with OFDM in 802.11a/g.

**Table 2-1:** Modulation and Coding rate for OFDM 802.11a/g

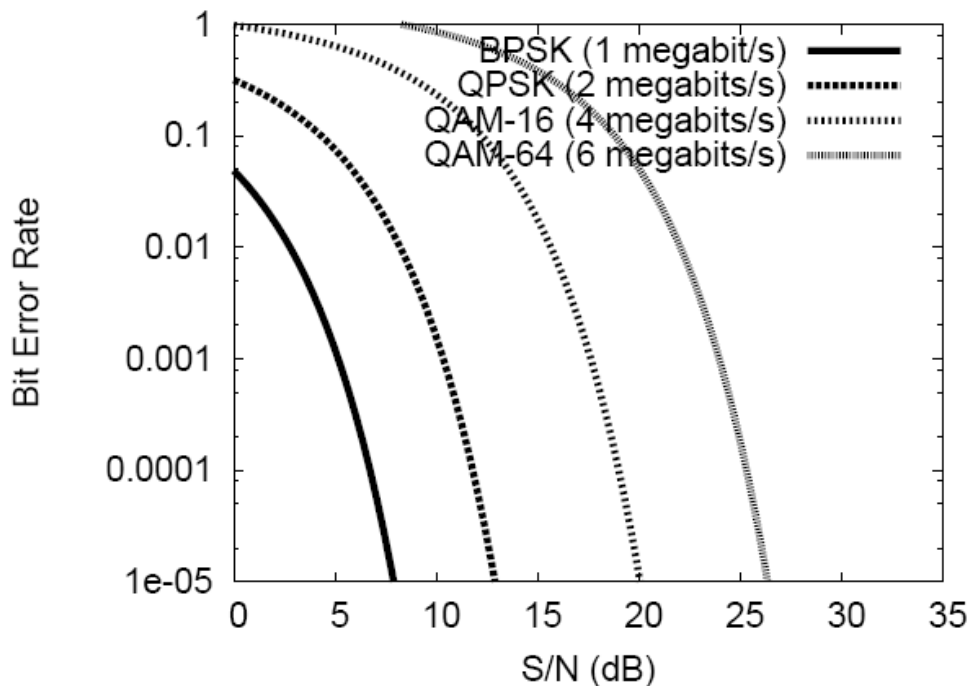
| <b>Data Rate (Mbps)</b> | <b>Modulation</b> | <b>Coding Rate</b> | <b>Data Bits per Symbol</b> |
|-------------------------|-------------------|--------------------|-----------------------------|
| 6                       | BPSK              | 1/2                | 24                          |
| 9                       | BPSK              | 3/4                | 36                          |
| 12                      | QPSK              | 1/2                | 48                          |
| 18                      | QPSK              | 3/4                | 72                          |
| 24                      | 16-QAM            | 1/2                | 96                          |
| 36                      | 16-QAM            | 3/4                | 144                         |
| 48                      | 64-QAM            | 2/3                | 192                         |
| 54                      | 64-QAM            | 3/4                | 216                         |

### 2.3.2 Performance of PHY

The wireless networks allow different modulation techniques for different link qualities. This allows the links to choose the modulation technique that best suites the conditions and thereby optimize the throughput as the link quality could vary. Each data rate uses a particular modulation scheme. Higher data rates are commonly achieved by more complex modulation schemes as is evident from Table 2-1. Modulation is a process of translating an outgoing data stream into a form suitable for transmission on the physical medium. This involves translating the data stream into a sequence of symbols. Each symbol may encode a certain number of bits, depending on the modulation scheme used. The symbol sequence is then transmitted at a certain rate, the symbol rate. So for a given symbol rate, the data rate is determined by the number of encoded bits per symbol.

The amount of information received over a specific link is dependant on the number of bits successfully decoded by the receiver. Wireless Channel can experience different types of interference and noise and the performance of different modulation

schemes is determined by its ability to preserve the accuracy of the encoded data. Interference, pathloss and fading in the wireless channel cause variation in the received signal to noise ratio SNR. High rate schemes typically use denser modulation encoding and therefore require higher SNR to ensure they can decode the bits successfully. Variation in the SNR also cause variations in the bit error rate BER because the lower the SNR the more difficult it is for the modulation scheme to decode the receiver signal. Figure 2-5 [3] plots the theoretical BER against the SNR for some of the modulation schemes assuming only Additive Gaussian White Noise (AWGN). There is generally a tradeoff between the data rate and the BER; the higher the data rate the higher the BER. The throughput in correct bits received per second is the product of symbol rate, bits per symbol and the probability that a bit is received correctly. For a system to cope with the variations of SNR it must be able to switch between multiple modulation schemes. The present WLAN devices support many data rates at different coding and modulation schemes. According to the Figure 2-5 as the link condition changes the SNR, if the modulation and coding is changed properly it is possible to get the best capacity from the system.



**Figure 2-5:** Theoretical BER vs. SNR for several modulation schemes in AWGN [3]



# CHAPTER 3

## Rate Adaptation in WLAN

### 3.1 Rate Adaptation

IEEE 802.11 a/b/g PHY [1] support multiple data rate transmission in wireless networks depending on the different Modulation and Coding schemes used. The IEEE standard does not define any rate adaptation schemes and it is open to the manufactures to create their own.

Rate adaptation is the process of dynamically switching between the available data rates depending on the channel conditions to achieve the rate that will give the optimum performance. The capability of WLAN to support multiple data rates enables the stations to select appropriate transmission rate depending on the required quality of service and channel conditions. Higher data rates correspond to higher throughput but require higher signal to noise ratios (SNR) for correct demodulation. If the quality of channel is not good i.e. SNR-limited environments then transmitting at higher rates may cause higher errors which correspond to retransmissions and thus low throughput. In such situation a more robust but lower data rate is required to be able to communicate. However if the channel quality is good giving higher SNR values then transmitting at higher rate is desirable.

Rate adaptation is a two step process; estimation and rate selection. Channel quality estimation involves the process of measuring the channel for predicting the future quality. The authors of [11] have defined further the estimation process as a three step approach; Layer selection, message selection and the estimation. The estimation indicators depend on the layer from which the information is gathered i.e. if the PHY layer approach is used the indicators are SNR or other PHY metrics. If the Link layer is used the indicators are the frame transmission results i.e. the Frame errors, ACKs or throughput calculations. The link layer approach uses the data frames to assess the quality of the channel and the PHY layer approach uses the control frames to assess the channel quality. The PHY layer approach estimation indicators i.e. SNR are translated into a predefined rate mappings, whereas the Link layer approach uses information from the previous transmitted frames.

## **3.2 Classification of Rate Adaptation Mechanisms**

The rate adaptation mechanisms have not been distinctively classified into different categories in the literature. Different publications have categorized it based on their own perception. Based on whether the rate adaptation function is performed at the transmitter or receiver it can be classified into transmitter and receiver based mechanisms [26]. The transmitter based rate adaptation mechanisms make use of statistics available and maintained about the transmitted data and can further be divided into frame error based and throughput based mechanisms which are a true representation of the Link Layer approach as discussed [11]. We classify Rate Adaptation mechanisms as Error/Loss based, Throughput based and SNR based mechanisms as proposed by [24]. In [9], an approach is defined which makes use of both the statistics and the SNR called Hybrid Rate Control.

### **3.2.1 Error/Loss based Adaptation**

In IEEE 802.11 WLAN the receiver sends an ACK frame upon successful reception of data frame. The transmitter assumes a successful transmission only upon reception of ACK frame. If the ACK frame is not received or received in error the transmitter assumes failure of the corresponding data frame. Whenever there are severe losses or

errors the rate adaptation mechanism should decrease the data rate. The errors/losses are often detected by the large number of retry counts, frame error threshold and successive transmission failures. Similarly to increase the data rate probe packets (data frames at different rate) are used to investigate conditions at higher rates. Probe packets can be single or multiple depending on the algorithms. If the probe packet is successful, switch to that particular data rate. ARF, AARF, Onoe and AMRR are some of the most widely used rate adaptation algorithms that use this approach.

### ***3.2.1.1 Auto Rate Fallback Algorithm***

Auto Rate Fallback ARF [25] is one of the first rate adaptation algorithms to be published. Many subsequent algorithms of the transmitter based class which uses error rates and retry counts as the channel estimate indicator are variants of ARF.

ARF works on the principle of successful and failed transmission counts. It keeps track of the successful transmissions and a timing function. ARF starts by selecting the highest possible data rate, if there are 2 successive transmission failures i.e. missed ACKs the data rate of the retry transmission and the subsequent transmissions are made at the adjacent lower data rate and a timer is started. When the timer expires or the number of successful transmissions reaches 10 the data rate is switched to next higher rate and the timer is stopped. The first packet at the next higher data rate is called probing packet and if it fails the data rate is lowered again.

Adaptive Auto Rate Fallback AARF [4] a variant of the ARF was developed with the idea to cope with the long term channel condition changes. The step-up parameter i.e. 10 in case of ARF is doubled every time when the probing packet fails with a maximum step-up success parameter of 50. The parameter is reset to 10 if the data rate is lowered due to 2 successive transmission failures, which was meant to take care of short term channel condition changes.

### 3.2.2 Through-Put Based adaptation

Throughput based rate adaptation mechanisms constantly sends probe packets at adjacent rates i.e. immediate lower and higher rate for a small fraction of time. Then at the end of the specified time the performance of the three is compared by calculating the throughput of the probe packets in the specified time. The rate is switched to the one with highest throughput. SampleRate [3] makes use of this approach. SampleRate is further discussed in chapter 4 in detail.

### 3.2.3 SNR based adaptation

SNR based mechanisms are usually receiver based that make use of some predefined mapping tables of SNR against throughput or error ratio to decide on the PHY rate selection. The SNR based mechanisms rely on the control frames to estimate and feedback the channel condition. The Received signal strength RSSI of the RTS frame is used to decide on the data rate and the information is sent back on the CTS frame. However the published SNR based rate adaptation mechanisms require changes to standard and have not been widely implemented.

#### 3.2.3.1 Receiver Based Auto Rate Algorithm

The Receiver Based Auto Rate RBAR algorithm [8] is receiver based mechanism that makes use of SNR as the channel estimate indicator. RBAR makes use of the RTS/CTS frames to select the data rate for the next data frame. The receiver of the RTS frame calculates the SNR of the received RTS frame and based on a priori channel model calculates the highest data rate for that particular SNR. The receiver then sends back the data rate information in CTS frame and the transmitter sends the next data packet with that particular data rate. However, this algorithm requires modification in the RTS, CTS and data frames it is not practically implemented.

OAR Opportunistic Auto Rate [13], a medium access protocol, extends the concept of RBAR and takes advantage of the good channel conditions at higher data rates. The OAR gives preference to nodes with good channel conditions and allows to opportunistically sending multiple data transmissions. OAR uses the same RTS/CTS

scheme as used in RBAR for rate control. It ensures that all nodes are given access to media for the same time-share as achieved by single data rate. This greatly increases the network throughput.

### **3.2.4 Hybrid Automatic Rate Control**

Hybrid rate control algorithm [9] makes use of combined statistic and SNR based algorithm. It uses a core controller based on throughput based algorithm and an SNR based controller which bounds the acceptable range of Signal strength of each data rate. The idea is prevent data transmissions at too high rate when the channel conditions suddenly changes.

## **3.3 Shortcomings of Current Design Approach**

### **3.3.1 Misinterpretation of Frame Loss**

The loss of frames in IEEE 802.11 can be due to Link errors i.e. low SNR values for a particular modulation scheme or due to collision over the wireless link. A collision occurs if two simultaneously transmitted frames interfere at the receiver so that those frames are lost. The IEEE 802.11 DCF discussed employs the CSMA/CA to avoid the collisions, however simultaneous transmission and hence frame loss can occur due to a phenomena called hidden node problem. The current rate adaptation mechanisms i.e. ARF, AARF and Onoe infer severe packet loss as a result of worsening link condition without taking the collision into account. Upon detection of frame losses via retry counts, successive transmission failures the data rate is reduced. In the presence of hidden terminals there may be collisions and hence severe packet loss but decreasing the data rate will not help in reducing the collision but will certainly decrease the throughput. The lower data rate increases the transmission time of the data as well as the collision period. Collision Aware Rate Adaptation CARA [10] algorithm has been developed to differentiate between frame errors due to channel errors and due to collisions which make use of the RTS/CTS exchange and the Clear Channel Assessment functionality of the IEEE 802.11. It assumes the RTS frame failure as due to collision and transmission failure after successful RTS/CTS due to

channel errors. However it has not been practically deployed and tested in diverse conditions.

### 3.3.2 SNR value as Channel Estimator

In theory the SNR can be the best channel estimation descriptor. It is related to Bit error rate, Frame error rate and the throughput and hence is an essential feedback for rate adaptation. However there are some limitations in the current rate adaptation algorithms [8] [12]. The relationship between the SNR and the data rates for certain links can be highly variable. This is mainly due to the fact that the current rate adaptation mechanisms like RBAR uses apriori channel models to calculate the SNR versus data rate lookup tables. Secondly in most of the cheap WLAN cards the uncalibrated Signal Strength Indication (SSI) is given which is not a true indicative of the link condition. The most severe problem with these receiver based algorithms is the feedback mechanism to relay the information back to the transmitter. The current standards does not provide any mechanism and most of the current mechanisms require some sort of changes to the standard which are practically not feasible. Furthermore [3] has shown that SNR has no strong correlation with the delivery probability at particular rate and thus it is difficult to determine the SNR threshold for a particular data rate.

# CHAPTER 4

## Algorithms under Test

This chapter gives a brief introduction to the three Rate Adaptation Algorithms Onoe, SampleRate and AMRR which are implemented in the Atheros communication chipsets as part of the Multimode Atheros Driver (MadWiFi) open source driver for the wireless chipsets.

### 4.1 Onoe

Onoe is an error based rate adaptation algorithm which makes use of a credit system which it accumulates for the current bit rate for one particular destination. These credits are determined by the successful transmissions, erroneous transmissions and retransmissions during a fixed period. The aim of Onoe is to find the bit rate with less than 50 % loss rate. Onoe keeps a track of the credits for current bit rate and keep on incrementing the credits till a threshold value is reached upon which it will switch to the next higher bit rate if the transmission has less packet loss.

Initially the rate is set to 24 Mbps for 802.11a/g and 11 Mbps for 802.11b and the credit value to zero. It updates the credits in a fixed observation period of default one

second. If no transmission was successful or more than 10% of the packets needed a retransmission decrease the bit rate to the next lower level immediately. The credits are incremented by one if less than 10% of the packets needed a retransmission in the observation period and decremented if more than 10% needed a retransmission. When the credit count for the current bit rate reaches a threshold value of 10 the bit rate is switched to the next higher rate and the credit count is reset to zero. The Onoe takes at least 10 seconds with its default settings to increase the bit rate to next higher level.

The Onoe Algorithm Performs the following steps periodically in an observation period of default 1 second by starting at 24 Mbps for 802.11a/g and 11 Mbps for 802.11b [3]:

- If no packets were sent successfully, move to the next lowest Data Rate and return
- If 10 or more packets were sent and the average number of retransmissions per packet was more than one then switch to next immediate lower Data Rate and return
- If the number of retransmissions was greater than 10% then decrease the number of credits and return
- If the number of retransmissions for packets was less than 10%, increment the number of credits
- If the current Data Rate has a credit of 10 or more switch to the next higher Data Rate and return
- Keep transmitting at the current Data Rate



## 4.2 SampleRate

SampleRate is throughput based rate adaptation algorithm which minimizes the mean packet transmission time and hence increase the throughput by sending packets at a bit rate with the minimum average transmission time. SampleRate gathers information about other rates by periodically sending packets at those bit rates other than the current rate and reduces the sample space by removing the bit rates that cannot send at higher rates than the current one or which had experienced successive transmission failures.

Initially the SampleRate send packets at the highest possible bit rate i.e. 54 Mbps for 802.11 a/g and 11 Mbps for 802.11b and if it is not possible to send packets after four transmissions it will decrease until it finds a bit rate that is capable of sending packets. After finding a bit rate it will start sending packets at that rate and send a data packet every 10th packet on a randomly selected bit rate other than the current one. It will make a list of eligible bit rates to be sampled by removing the bit rates that experienced four consecutive transmission failures or whose transmission time without retries is more than the current one. SampleRate calculates the average transmission time for a bit rate by averaging the packet transmission time in an averaging time window of the previous 10 seconds. The bit rate is switched to the one with the minimum average transmission time. The transmission time of an N-Byte packet given B bit rate and R retries is calculated by the following equation [3].

$$Tx-Time = DIFS + Backoff(R) + (R+1) * (SIFS + ACK + Header + (N*8/B))$$

The *Header* in the equation above indicates the UDP/TCP header, IP header, MAC header, PLCP preamble and PLCP header time. *Backoff(R)* is the average backoff period after a certain number of retries, which is discussed in section 2.2.4. *DIFS* is 50  $\mu$ sec in 802.11b and 28  $\mu$ sec in 802.11a/g. *SIFS* is 10  $\mu$ sec in 802.11b and 9  $\mu$ sec in 802.11a/g.

The SampleRate algorithm is implemented in three different functions; the first function allocates a Data Rate to each packet. The second function keeps a list of the transmission statistics i.e. transmission time, average transmission time, successful

packets and successive failures for the Data Rates. The third function removes information from the list of transmission statistics that were obtained before 10 seconds. Three functions can be implemented and explained in the following steps [3]

The first function is called before packet transmission and performs the steps as:

- If no packets were successfully sent then select the Data Rate with less than 4 successive failures
- If the number of packets is multiple of 10, select a random Data Rate from list of Data Rates that have less than 4 successive failures and whose minimum transmission time is less than average transmission time of current Data Rate
- Send packet at Data Rate with minimum average transmission time

The second function is called immediately after packet transmission and updates information about transmission results:

- Calculate transmission time of Data Rate
- Add the transmission time to total transmission time for that Data Rate
- If transmission is successful increase the number of successful packets for that Data Rate
- If transmission failed increase number of successive failure for that Data Rate
- Calculate average transmission time for that Data Rate according to the sum of transmission times and successful packets
- Set the current Data Rate with least average transmission time
- Add the statistics for Data Rate to list of transmission results

The third function is called before packet is sent and it removes results from the transmission list that were obtained more than 10 seconds before:

- Remove transmission time from total transmission time for that Data Rate

- If the packet succeeded, decrease the number of successful packets at that Data Rate
- Recalculate the average transmission time for each Data Rate
- Set the current Data Rate to the one with least average transmission time

### 4.3 AMRR

Adaptive Multi Rate Retry AMRR is a retry based rate adaptation algorithm which adaptively raises the threshold for rate increase to prevent attempts at Data Rates higher than the optimal one.

Different transmission counters having a threshold value are associated with different Data Rates. If the number of transmission retries for a Data Rate exceeds the threshold value the data rate is decreased. AMRR makes use of an adaptive method to control the time after which the transmission counter and Data Rate pair values are changed. It uses Binary Exponential Backoff technique to adapt the length of time used to change the values of rate and counter pair values. It sends probe packets and depending on its transmission status changes the threshold time.

AMRR is based on MadWiFi algorithm which has four pair of rate and transmission count fields ( $r_0/c_0$ ,  $r_1/c_1$ ,  $r_2/c_2$  and  $r_3/c_3$ ). Initially the data rate is  $r_0$ , if the transmission fails it keeps on trying to send the data at  $r_0$ ,  $c_0-1$  times. If the transmission keeps on failing it tries rate  $r_1$ ,  $c_1$  times then  $r_2$ ,  $c_2$  times and then  $r_3$ ,  $c_3$  times. In AMRR  $c_0 = 1$ ,  $c_1 = 1$ ,  $c_2 = 1$  and  $c_3 = 1$ , so for a single retransmission failure the rate is changed. The rate  $r_3$  is always set to be the lowest rate available. The value of  $r_0$  is determined from the previous value of  $r_0$  and transmission results for the elapsed period. The initial value of  $r_0$  is 11 Mbps for 802.11b and 24 Mbps for 802.11a/g. The AMRR algorithm performs these steps to change the value of  $r_0$ :

- If less than 10% of transmission packets failed and the transmission of 10 distinct packets was attempted, increase the successful transmit counter

- If successful transmit counter reaches threshold value, send a probe packet at next higher Data Rate
- If probe packet was successful, reset the successful transmit counter and increase the Data Rate
- If probe packet failed double the successful transmit threshold value but keep it below the maximum threshold value
- If more then 33% of packets failed reset the successful transmit counter and decrease the Data Rate
- Reset the rate and counter pair value according to the value of  $r_0$

# CHAPTER 5

## Experimental Setup

This chapter describes the experimental setup that has been used to test the performance of the three Rate Adaptation Algorithms i.e. Onoe, AMRR and SampleRate. The experimental tests have been performed in the Laboratory of Communication Engineering at the Helsinki University of Technology (TKK).

Several recent studies have been directed towards the performance of the PHY 802.11 Rate Adaptation Algorithms through experimental study but most of them have been performed in uncontrolled environments with very little reliability in similarity of the wireless channel for different Algorithms going for the same experimental tests. The experimental tests performed for evaluation of the algorithms in this thesis have been performed using the Electrowire Wideband C2 Channel Simulator. This provides a realistic testbed for different wireless channels which can be used for the experimental tests in a repetitive controlled manner thus providing the same level field for comparison of the three Rate Adaptation Algorithms.

The laboratory setup involves the installation of hardware and software and the calibration of the connection setup.

## **5.1 Performance Metrics and Measured Parameters**

To be able to compare the performance of the three Algorithms we must define the performance metrics and parameters to be used in the experiment. In most of the WLANS performance testing for channel selection, roaming, mobility and interoperability with other networks the most common test parameters are the system delay, jitter, error ratios and throughput.

For the performance testing of rate adaptation algorithms the most suitable test parameter will be the throughput of the system which gives a fair representation of the PHY 802.11 rate selected under different channel conditions. In our case we will focus on the application level throughput for User Datagram Protocol UDP and Transmission Control Protocol TCP generated traffic at different pathloss values using different Wireless Channel Models defined by the 802.11 Task Group n (TGn) [5].

The Application level throughput for UDP and TCP traffic is a suitable representation of the performance metric for the algorithms under test. The three algorithms belong to the sender based mechanism which uses the throughput and error/loss count as the channel estimator. The throughput of the UDP traffic, which does not take into account the retransmission of loss packets, along with the packet loss ratio will give a good comparative study of how the algorithms behave under different channel conditions set by pathloss values. The TCP throughput, which has its own Backoff mechanism to counter errors, will be an interesting parameter to compare with the UDP packet loss ratio.

## **5.2 Experimental Test Bed Hardware Equipment**

The experimental setup for the performance evaluation of the RAA includes the following Hardware available at the Student Laboratory of Communication Engineering at Helsinki University of Technology (TKK).

### **5.2.1 Wireless Nodes**

The experiment was performed on two AMD Athlon Desktop PCs (1.4 GHz /512MB and 800 MHz/256MB) running Linux based Ubuntu 7.04, Gutsy Gibbon, kernel version 2.6.22-14-generic as the operating system. Linux as operating system for this experiment was the ideal choice in the context of the thesis as it comes with several open source tools and APIs to manipulate the configuration setup. The experiments require constant interchange between the Rate Adaptation Algorithms and Wireless Tools for Linux i.e. WT (Wireless tools) provided by HP [18] was an essential tool used for this purpose.

### **5.2.2 Wireless LAN Adaptors**

TPLink TL-WN651G Wireless LAN adaptors with an external antenna are installed on the Desktop PCs. These Adaptors comes with Atheros based chipsets for which open source Linux based drivers are available. The Multi-band Atheros Driver for WiFi (MadWiFi) which contain the three Rate Adaptation Algorithms i.e. SampleRate, Onoe and AMRR as separate kernel modules inside the main driver is used for this experiment. The list of WLAN adaptors which support the MadWiFi driver is available at [2]. The MadWiFi driver v0.9.3.3 used in the experiment has the following version of Rate adaptation algorithms. SampleRate(v 1.2), Onoe(v 1.0), and AMRR(v 0.1). The Installation procedure of the MadWiFi driver and changing the Rate Adaptation Algorithm can be seen at Appendix C.

### **5.2.3 Channel Simulator**

PROPSim C2 wideband channel simulator is used in the experimental setup to simulate real time radio channels models. The channel simulator gives the experiment reliability and consistency to perform the evaluation in a controlled wireless environment.



**Figure 5-1: PROPSim C2 Channel Simulator**

The PROPSim C2 is generic channel simulator which simulates only the radio channel excluding the transmitter and receiver, thus it is independent of the system or modulation. It supports 30 MHz RF bandwidth, 0.1 ns delay resolution and 24 fading Paths per channel. The channel simulator has a large number of inbuilt channel models to support a wide range of wireless systems like Global System for Mobile Communication GSM, Digital Cellular System DCS, Personal Communication Service PCS, Wideband Code Division Multiple Access WCDMA and International Mobile Telecommunication-2000 IMT-2000 and also support user defined channel models. The channel models defined by WLAN task group TGn used in this thesis can be easily configured with the integrated PC, touch screen and keyboard. PROPSim C2 gives the possibility of defining different fading profiles, selecting different shapes of Doppler spectrum and creating hopping or smoothly sliding delays for various paths. The following paths parameters can be adjusted to define the channel model.

**Delay:**

Delay indicates the delay of the signal path related to direct signal path between a transmitter and receiver in Nanoseconds with a range between 0 and 400000ns.



### **Delay function:**

Delay function indicates the function according to which the delay behaves, taking several values i.e. Fixed, Hopping, Linear and sinusoidal.

### **Amplitude:**

Amplitude indicates the amplitudes of the path in decibels relative to each other taking a value in the range between -60 and 0 dB.

### **Amplitude Distribution:**

Amplitude distribution defines the method according to which the amplitude behaves, having certain alternatives i.e. Constant, Rayleigh, Pure Doppler and Rice.

### **Distribution Parameters:**

This field changes according to the amplitude distribution selected with the following options.

- Constant: phase shift in degrees
- Rayleigh: No parameters
- Pure Doppler: Angle between the mobile motion and the incoming radio wave in degrees
- Rice: Angle between the mobile and the direct ray in degrees.

### **Doppler spectrum:**

Doppler spectrum indicates the shapes of the Doppler spread taking the values i.e. Classical, Flat or Rounded.

After defining the channel model the simulation parameters can be defined and changed depending on different experimental scenarios. The simulation parameters

include, setting the Mobile speed in Km/h, the average signal input level, and the centre frequency of the channel model in Mega Hertz with a range between 350 and 6000 MHz and the Output Gain in decibels which indicates how much the signal is attenuated at the output channel.

The PROPSim C2 channel simulator has two one way physical Radio Frequency RF channels at the back panel. Since, the experimental setup need a two way communication channel the pair of one way channels can be used in combination with circulators and connectors to make a two way channel.

### 5.2.4 Signal Generator

In the PROPSim C2 channel simulator RF signal is converted to IF (Intermediate Frequency) and then to the basedband before going through the digital fading simulations. Therefore every RF interface needs an external RF local oscillator for conversion to IF. In this experimental setup Rhodes&Shwarz SMJ 100A Vector Signal generator is employed for generating the Radio Frequency Local Oscillator RFLO signal required by the C2 channel simulator.



**Figure 5-2: SMJ 100A Vector Signal Generator**

## 5.3 Traffic Generation Tool

The traffic generation tool used for generating traffic between the two wireless nodes is Iperf, supported by National Laboratory for Applied Network Research [19]. Iperf works in client and server mode and can be used to emulate both Transmission Control Protocol/Internet Protocol TCP/IP and User Datagram Protocol/Internet

Protocol UDP/IP applications and measure several network performance parameters such as throughput, jitter and datagram loss.

For the performance evaluation of the Rate Adaptation Algorithms Iperf has been used to generate both TCP and UDP data streams and measure the throughput and datagram loss.

## **5.4 Experimental Setup Calibration**

The PROPSim C2 channel simulator has two one way channels. To simulate a two way channel used in WLAN the two channels in the simulator can be used by splitting the WLAN channel between the two WLAN adaptors on the computers into uplink and downlink. This can be done by using circulators, connectors and a set of cables to separate the uplink and downlink and using the two channels of simulator separately for the two channels.

But before proceeding with the connection setup with channel simulator, the whole setup without the channel simulator needs to be calibrated for the attenuation present in the setup due to the connectors, circulators and RF cables. The main idea is to measure the exact attenuation of the connection setup and remove faulty devices that give unnecessary large attenuations.

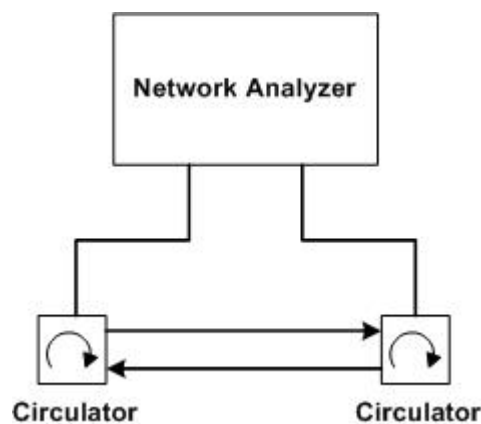
The measurement is done by using Ballman NWA S200 True Multiport Vector Network Analyzer. The network analyzer uses a full two port setup for generating and receiving a signal. The device under test is connected to the two ports and signal generated from one port and received via the other can be analyzed through a graphical interface.

The following parameters were set in the Network Analyzer for the devices under test.

Signal power = 0 dB

Frequency range = 2000 – 6000 MHz.

The attenuation values of the device under test are at a reference frequency of 2463 MHz. The following procedure was followed for the whole connection setup. First the RF cables are individually tested and the attenuation in each is measured. Then two pair of cables connected via connectors is separately tested. The circulators are tested for isolation between the three ports, which are then connected to the two pair of cables in such a way that two separate one way links are created. The connection setup is illustrated in the Figure 5-3. The final connection setup calibrated gives a 2 dB attenuation value and can be seen at Appendix D.

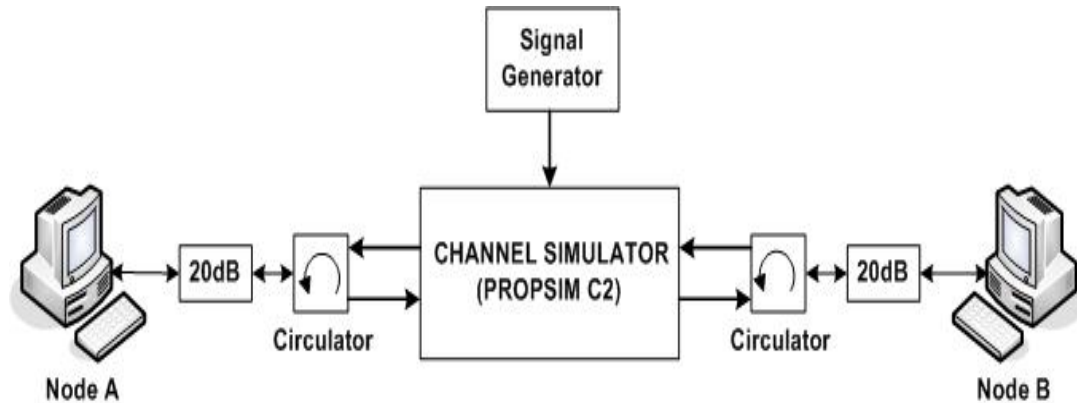


**Figure 5-3:** Calibration of Connection Setup

## 5.5 Connection Setup

After calibration of the connection setup without the PROPSim C2 simulator the final connection setup can be established by employing the PCs with WLAN adaptors, simulator, signal generator and fixed attenuators. The signal from the two WLAN adaptors is split into two using circulators and using the two Physical RF channels of the PROPSim C2 simulator in opposite directions a two way channel can be established. A fixed attenuation of 20 dB is connected between the PC and the circulator thus giving an overall attenuation of 40 dB fixed attenuation. The fixed attenuators serve two purposes; one it realizes pathloss values which is the basic parameter for the performance evaluation and second it safeguards the output port of the Physical RF channel from any leaked signal. The signal from SMJ 100A signal

generator is split into two using a splitter and fed to RFLO IN port of the each Physical RF channel. The final connection setup is illustrated in the Figure 5-4.



**Figure 5-4:** Testbed Setup

## 5.6 Channel Models

The comparative evaluation of the Rate Adaptation Algorithms in WLAN requires controlled experimental environment which can be realized by using PROPSim C2 simulator which gives possibility to emulate a wide range of channel conditions in repeatable manner. For the evaluation, the channel environment scenarios used are based on the channel models described by 802.11 Task Group n (TGn) [5].

The channel models developed by TGn are based on the set of five WLAN channel models developed by Medbo et al. [6]. The TGn has proposed six WLAN channel models using three channel models proposed by Medbo et al [6] and developed an additional three. We will use a subset of Channel models proposed by TGn and Medbo et al. The delay profile models for different environments proposed by the TGn and Medbo et al are described as follows.

**Model A:** representative of a typical office environment, non-line of sight (NLOS) conditions with 50 ns rms delay spread

**Model B:** representative of a large open space and office environment, NLOS conditions with 100 ns rms delay spread.

**Model C:** representative of large open space (indoor and outdoor), NLOS conditions with 150 ns rms delay spread.

**Model D:** representative of large open space (indoor and outdoor), same as Model C, but for LOS conditions. A 10 dB spike at zero delay with rms delay spread of 140 ns.

The delay profile of the channel models with channel coefficient (tap delays and corresponding powers) are defined in the form of Tables and can be seen at Appendix A.

The path loss model proposed by TGn for the above channel models consists of Free Space loss with a path loss exponent of 2 up to a breakpoint distance. The path loss exponent of 3.5 is used after the breakpoint distance given by the equation below

$$L(d) = L_{FS}(d) \quad d \leq d_{BP}$$

$$L(d) = L_{FS}(d_{BP}) + 35 \log_{10}(d/d_{BP}) \quad d > d_{BP}$$

Where  $d$  is the transmit-receive separation distance in meters. The break point distance  $d_{BP}$  for the channel models A, B and C are 10, 20 and 30 respectively.

## CHAPTER 6

# Performance Evaluation Test Scenarios

This chapter details the experimental test cases and scenarios that are used to evaluate the performance of the three Rate Adaptation Algorithms. The aim of these tests is to find out how the Algorithms perform under different wireless environments realized by using the C2 channel simulator. The evaluation of the algorithms is based on a set of realistic workloads consisting of CBR/UDP and TCP traffic generated by the traffic generation utility Iperf.

We will first define the pre-conditions for the test cases, which include the WLAN configuration of the Wireless nodes and the PROPSim C2 channel simulator configuration. This is followed by the description of the wireless channel simulation parameters and finally we will define the test case and evaluate and compare the Rate Adaptation Algorithms in light of the test results.

## 6.1 Test Setup Configuration

The experimental tests were performed by using the setup shown in Figure 5-4. The desktop PCs with the WLAN adaptors act as the end hosts for the wireless link in the test setup. The WLAN adaptors were configured in 802.11 Ad-hoc mode. 802.11g has been used in all the experiments with RTS/CTS disabled. The use of 802.11g gives us the opportunity to perform rate adaptation in the presence of the higher data rates based on OFDM and the lower data rates available in 802.11b. Earlier studies on the performance testing in realistic wireless environment [14] have been performed in 802.11b with a maximum data rate of 11Mbps, so these tests will give us further information on how the algorithms behave with the availability higher data rates. The wireless channel 11 which corresponds to 2462 MHz of frequency was used throughout the experiment. The test setup was calibrated for this particular frequency. Throughout the tests a fixed Transmit Power of 10 dBm was used at both sender and receiver.

**Table 6-1: WLAN configuration Parameters**

| <b>WLAN Parameter</b> | <b>Value</b>   |
|-----------------------|----------------|
| Mode                  | 802.11g Ad-hoc |
| Channel               | 11 (2462 MHz)  |
| Transmit Power        | 10 dBm         |
| Rate                  | 54 Mbps Auto   |

## 6.2 Wireless Channel Simulation Parameters

The experimental tests were performed for each algorithm using the channel models defined by TGn [5]. The channel models were configured in the C2 channel simulator according to the delay profile given at the Appendix A. All tests were performed at a Doppler speed of 3 Km/h and frequency 2462 MHz. for each of the channel models the tests were performed by varying the pathloss value. The pathloss value can be realized and calculated as given by the equation below.



$$\text{Pathloss} = \text{Cable loss} + \text{Connector Loss} + \text{Circulator loss} + \text{Fixed attenuation} + \text{Channel Output Gain}$$

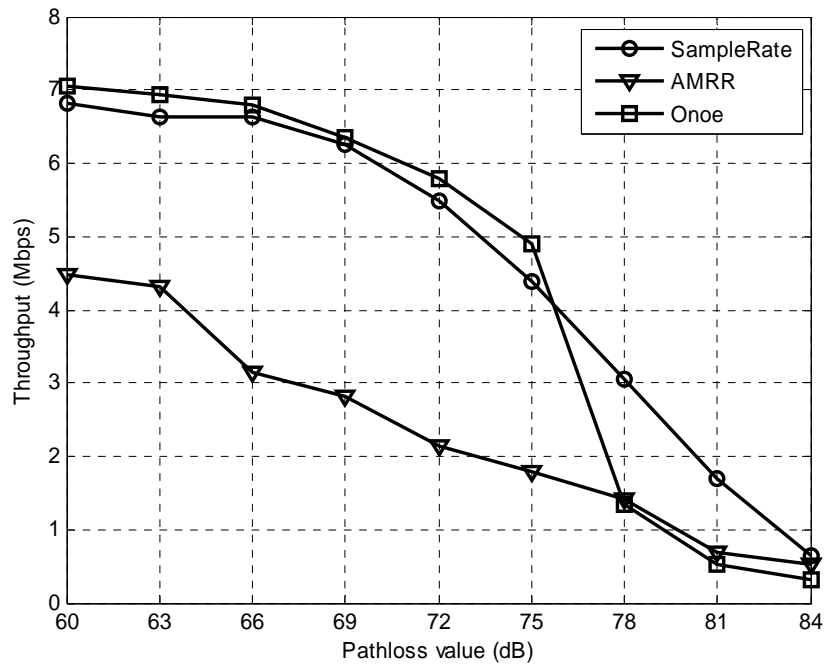
The pathloss value is varied by changing only the channel output gain parameter in the C2 channel simulator. The varying pathloss values give a good approximation of the distance between the sender and receiver, as proposed in [5] by the pathloss model, and how the rate adaptation algorithms behave under the varying channel conditions.

## 6.3 Test Case

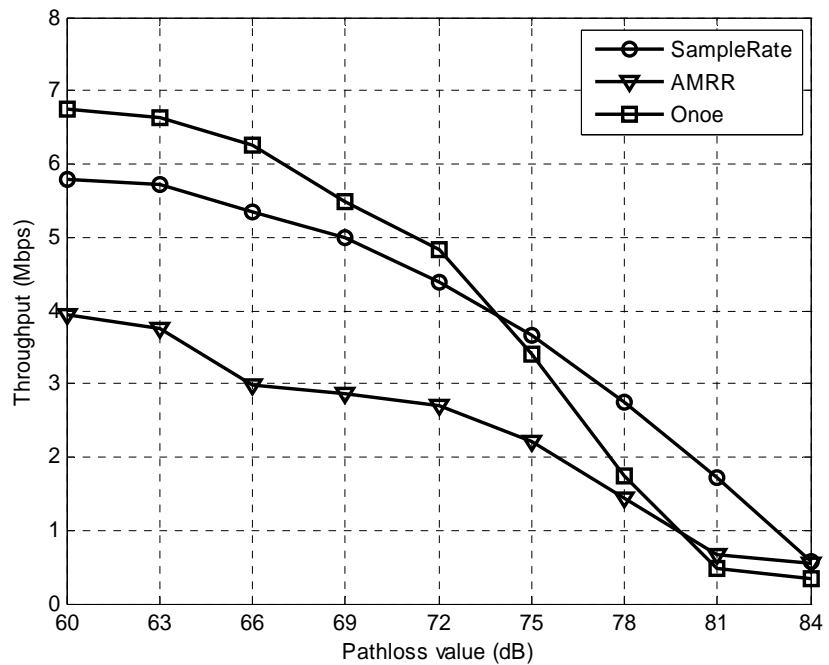
### 6.3.1 CBR/UDP Traffic

We will first examine the performance of the three rate adaptation algorithms using the CBR/UDP traffic. The UDP traffic considered is valuable in several respects. It is a representative of the real-time traffic, i.e. streaming media applications and also VoIP (Voice over IP) traffic. Furthermore, the unreliable nature of UDP gives us other important metrics like the loss ratio apart from the application level throughput.

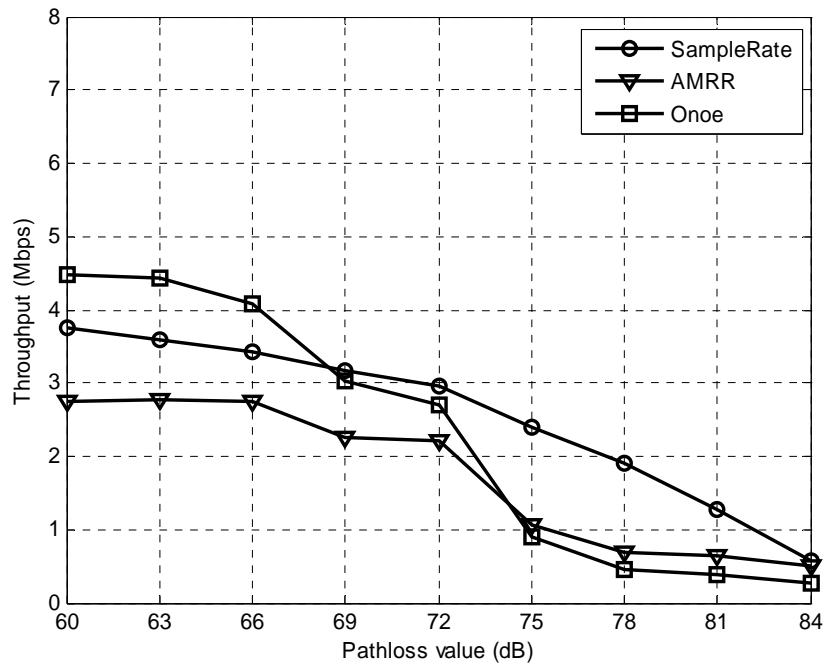
We will generate the UDP traffic from one wireless node to the other in Ad-hoc mode using the Iperf utility and observe the average throughput. The test will be run for each of the rate adaptation algorithm i.e. SampleRate, Onoe and AMRR respectively by varying the pathloss values. The tests are run for saturated traffic load while keeping the packet size at 1470 Bytes to avoid fragmentation. The following figures show the relative performance of the algorithms in terms of the average throughput for the different channel models. Each value corresponds to an average throughput value taken over a period of 180 seconds.



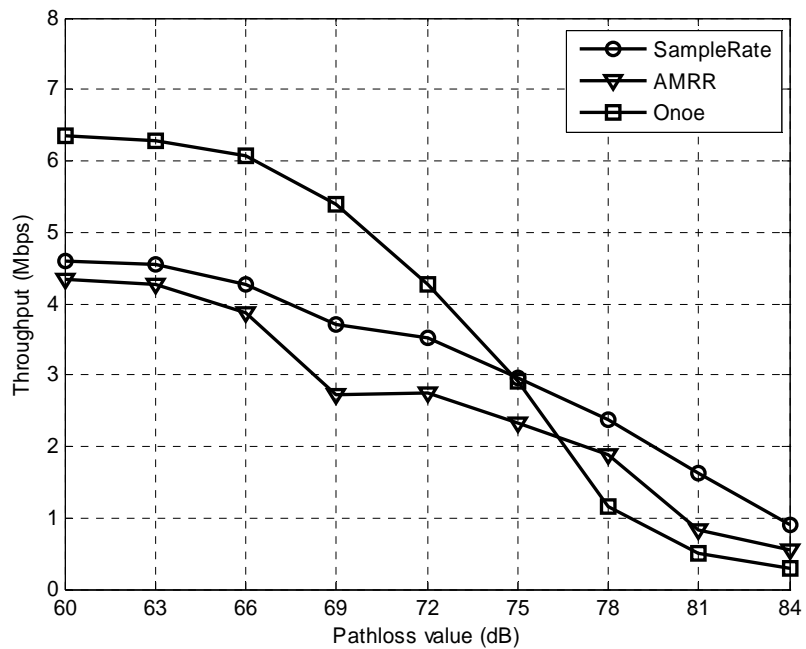
**Figure 6-1:** Average Throughput Vs Pathloss value for UDP Traffic across Channel Model A (50ns rms delay spread NLOS)



**Figure 6-2:** Average Throughput Vs Pathloss value for UDP Traffic across Channel Model B (100 ns rms delay spread NLOS)



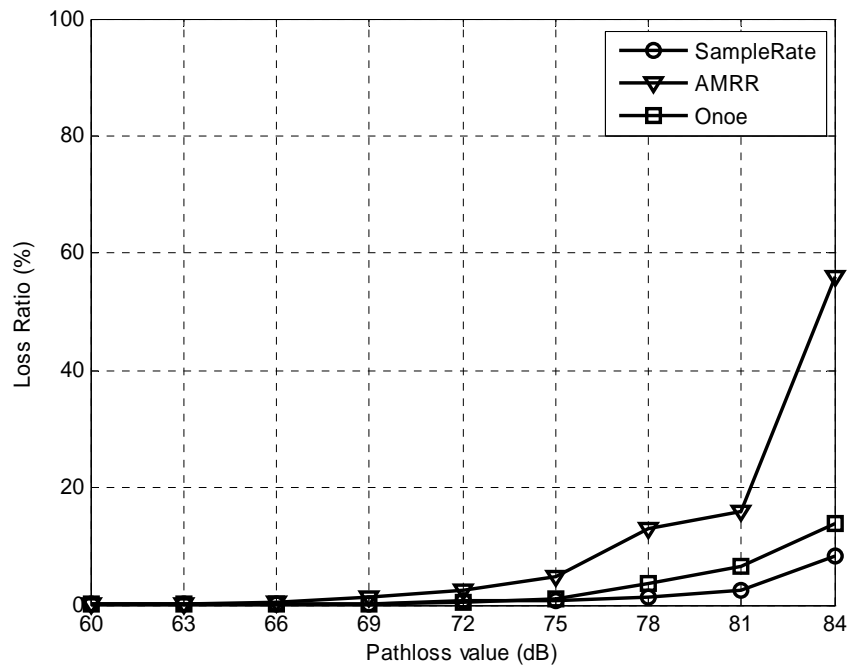
**Figure 6-3:** Average Throughput Vs Pathloss value for UDP Traffic across Channel Model C (150 ns rms delay spread NLOS)



**Figure 6-4:** Average Throughput Vs Pathloss value for UDP Traffic across Channel Model D (140 ns rms delay spread LOS)

It can be seen from the figures that Onoe has relatively better performance than the SampleRate at lower pathloss values as we go from Channel Model A to D. Though, SampleRate shows better performance at higher pathloss values. AMRR performs poorly compared to SampleRate and Onoe throughout. The throughput difference can be attributed to their design differences. The performance of AMRR can be explained by the fact that it tries to act quickly to the short term variations of the channel models by setting the retransmission parameter to one thus trying too many data rates which effectively lowers its average throughput. The better performance of Onoe at lower pathloss values is evident from the fact that it is quite conservative in nature and it stays at a particular rate for longer duration and thus has a higher average throughput than the SampleRate which though, is a throughput based algorithm that probes randomly every 10th packet to find a rate with less average transmission time, yet has less average throughput. Furthermore, Onoe is most likely to find the highest data rate with less loss ratio in these conditions. On the other hand SampleRate will send probes every 10th packet and in case where it is on a lower data rate there is a chance that the probe packet will fail which will effectively increase the average transmission time of the higher sampled rate. It will take longer time to reduce the average transmission time and switch to that rate. The performance of SampleRate at higher pathloss values is better than the Onoe and AMRR. In situations where the link conditions deteriorate the Onoe will switch to lowest PHY data rate and stay there if the higher data rates give more than 10 % loss for the probing period after it has accumulated the required credits. In such conditions all the data rates will encounter high packet losses but it cannot be concluded that higher data rates will perform poorly just because lower data rates perform poorly which has been investigated in [3] that in lossy conditions higher data rates still provide the highest throughput.

For further analysis if we look at the loss ratio in Figure 6-5 it is evident that at lower Pathloss values the loss ratio is lower for the three algorithms and increases as the Pathloss value increases with AMRR performing poorly. The SampleRate will perform relatively better than Onoe at the higher Pathloss values.



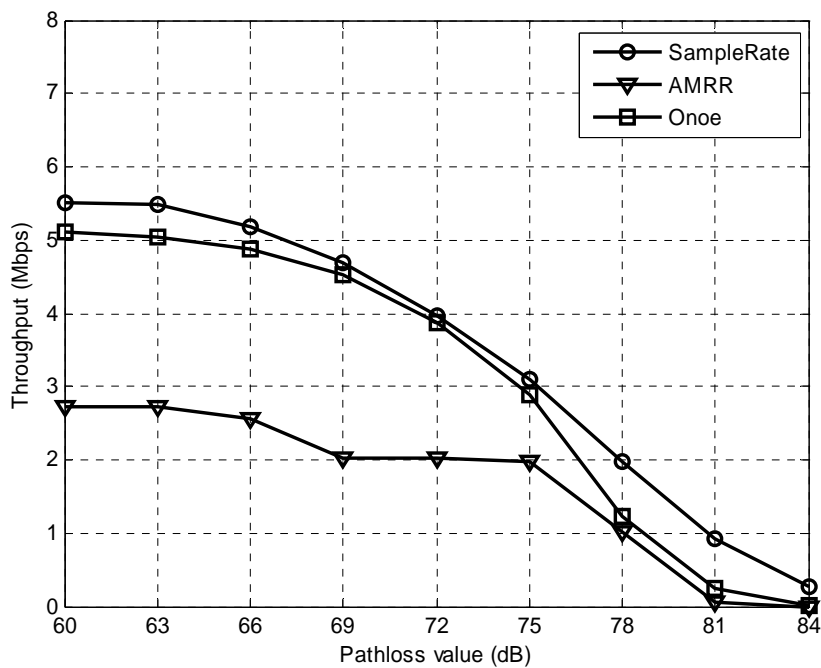
**Figure 6-5:** Loss Ratio Vs Pathloss value for UDP Traffic across Channel Model C (150 ns rms delay spread NLOS)

The transition of Onoe to lower data rate value in situations where a higher data rate has a credit value close to the threshold value of 10 can take relatively more time and once it falls to lower data rate failure of the initial transmissions can further reduce its data rate although the intermediate data rate may perform better. On the other hand SampleRate will most likely find the data rate with higher throughput at higher pathloss values. AMRR is the most aggressive of the three algorithms and tries to closely follow the short term variations of the channel [4] thus it will change the rate more frequently within the rate and retransmission counter pairs.

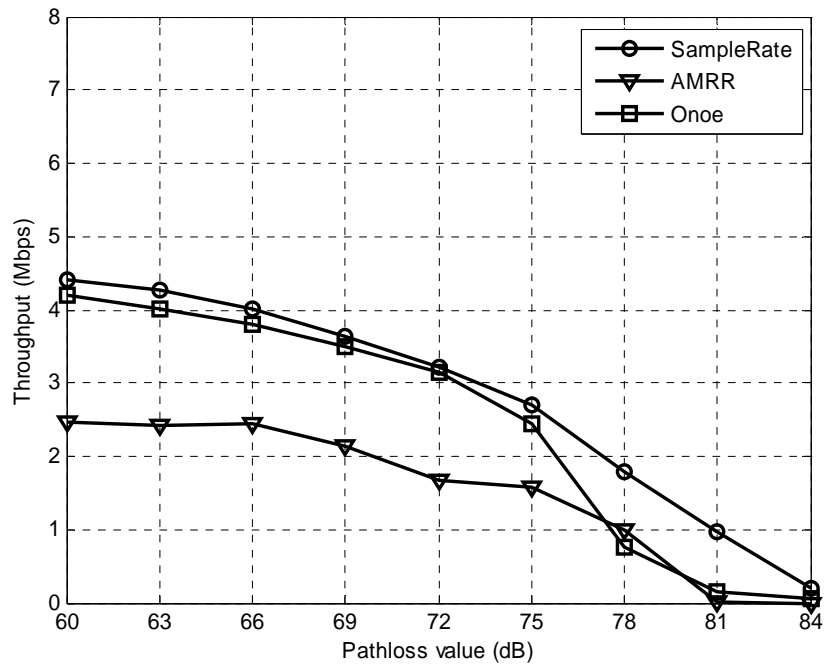
### 6.3.2 TCP Traffic

We will examine the performance of the three rate adaptation algorithms using the TCP traffic. The TCP traffic will be a good representative of the applications that run over the more reliable transport protocol like the TCP e.g. File Transfer Application and Web traffic. A key characteristic of the TCP is retransmissions in case of transmission failures and delays greater than estimated round trip time RTT.

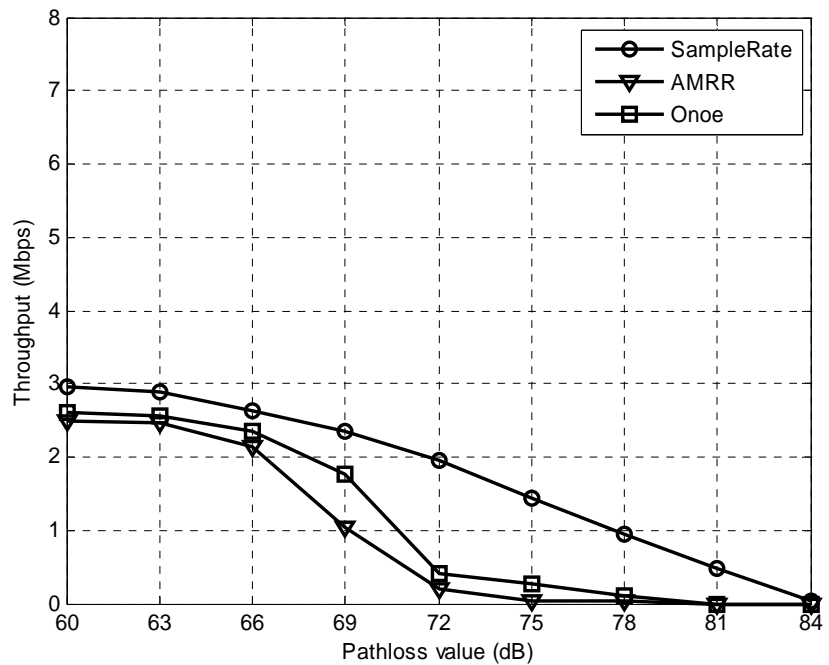
We will generate the TCP traffic from one wireless node to the other in Ad-hoc mode using the Iperf utility and observe the average throughput. The test will be run for each of the rate adaptation algorithm i.e. SampleRate, Onoe and AMRR respectively by varying the Pathloss values. The following figures show the relative performance of the algorithms in terms of the average throughput for the different channel models. Each value corresponds to an average throughput value taken over a period of 180 seconds.



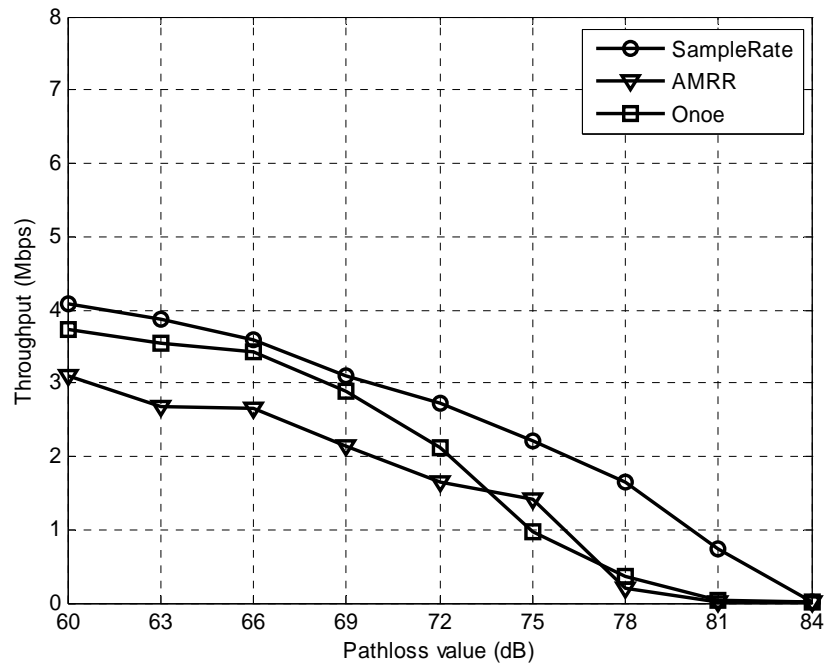
**Figure 6-6:** Average Throughput Vs Pathloss value for TCP Traffic across Channel Model A (50 ns rms delay spread NLOS)



**Figure 6-7:** Average Throughput Vs Pathloss value for TCP Traffic across Channel Model B (100 ns rms delay spread NLOS)



**Figure 6-8:** Average Throughput Vs Pathloss value for TCP Traffic across Channel Model C (150 ns rms delay spread NLOS)



**Figure 6-9:** Average Throughput Vs Pathloss value for TCP Traffic across Channel Model D (140 ns rms delay spread LOS)

From the figures above we can observe that AMRR still has the worst performance compared to SampleRate and Onoe. Interestingly, the SampleRate and Onoe have similar behavior at low Pathloss values with SampleRate relatively outperforming Onoe. Here the TCP level rate control has an impact on the throughput values as opposed to UDP traffic of saturation mode where there is packet to send every time the medium is ready to transmit. For TCP traffic, in conditions where there is negligible loss ratio the SampleRate and Onoe will perform relatively similarly as Onoe will always find the best data rate with less loss rate but the rate TCP rate control will have a major impact on the throughput.

At higher pathloss values SampleRate will always perform better than the AMRR and Onoe. This behavior can be related to the loss ratio in the UDP traffic. From the Figure 6-5 it can be seen that loss ratio considerably increases for AMRR and Onoe. This will have an impact on the TCP traffic as there will be retransmissions for every transmission failure at the application level which greatly reduces the throughput. An interesting insight here is that SampleRate should have higher loss ratio as compared



to Onoe as it regularly probes other data rates, but it is not the case as it will not probe at data rates that experience four successive failures thus decreasing the Sample space. Furthermore it is more stable than the other two algorithms in lossy conditions in terms of finding the data rate giving better throughput.

# CHAPTER 7

## Conclusion and Future Work

### 7.1 Conclusion

In this Thesis we have tested and compared the performance of three Rate Adaptation Algorithms available for Atheros based wireless chipsets using the Madwifi driver. The three Rate Adaptation Algorithms i.e. SampleRate, Onoe and AMRR were tested in realistic wireless channel conditions realized by the use of the PROPSim C2 Channel Simulator. The performance was measured, in varying wireless conditions through use of Channel Models provided by the Task Group for 802.11 TGn, for UDP and TCP traffic loads and the average throughput and loss ratio with the different Rate Adaptation Algorithms has been reported.

It has been observed that the performance of different Rate Adaptation Algorithms vary significantly owing to their design attributes in varying channel conditions. The channel quality estimator indicators can have a major impact on the performance. The algorithms studied here use statistics, such as retransmission counts for Onoe and AMRR and throughput calculation for SampleRate, for estimating the future quality of the channel to adapt the PHY data rates accordingly. It can be concluded that use of statistics based on probe packets can be misleading at times in estimating the state of

the wireless channel. Switching between too many data rates in case of AMRR due to single retransmission error can severely degrade the performance. Onoe performs better in low loss conditions as it is willing to stay at the higher data rates for longer duration, however it falls steeply to lower data rates in deteriorating channel conditions if the probe packets in the initial observation period suffer immediate loss. Performance of SampleRate can be relatively lower than Onoe in low loss conditions but improves considerably in deteriorating channel conditions.

## **7.2 Future Work**

The thesis work has mainly focused on the performance of Rate Adaptation Algorithms in Ad-hoc mode between a set of wireless nodes. It would be interesting to see the performance of the Rate Adaptation Algorithms in a congested environment where the wireless nodes contend for medium access. It will also be interesting how the Rate Adaptation Algorithms behave to frame losses due to collision rather than low channel quality. Furthermore a comparative study of SNR based rate adaptation algorithms and the statistics based algorithms can be valuable to evaluate the channel estimation indicators.

# BIBLIOGRAPHY

- [1] IEEE Std. 802.11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications 1999
- [2] “MadWiFi Project”, available online on May 12, 2008 at <http://www.madwifi.org>
- [3] J. C. Bicket, “Bit-rate Selection in Wireless Networks,” M.S Thesis, MIT, February 2005
- [4] M. Lacage, H. Manshaei and T. Turetletti, “IEEE 802.11 Rate Adaptation: A Practical Approach,” In Proceedings of the 7<sup>th</sup> ACM international symposium on Modeling, Analysis and Simulation of Wireless and Mobile Systems ACM MSWiM, October 2004, pp 126-134
- [5] IEEE 802.11 Wireless LANs Task Group n, “TGn Channel Models,” May 2004
- [6] J. Medbo and P. Schramm, “Channel Models for HIPERLAN/2,” ETSI/BRAN document no. 3ERI085B, March 1998
- [7] D. Qiao, S. Choi and K.G. Shin, “Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs,” IEEE Transactions on Mobile Computing TMC October 2002, Vol. 1, pp. 278-292
- [8] G. Holland, N. Vaidya and P. Bahl, “A Rate Adaptation MAC Protocol for Multi-Hop Wireless Networks,” In Proceedings of 7<sup>th</sup> Annual International Conference on Mobile Computing and Networking ACM MobiCom, July 2001, pp 236-251
- [9] I. Haratcherev, K. Langendoen, R. Lagendijk and H. Sips, “Hybrid Rate Control for IEEE 802.11,” In Proceeding of 2<sup>nd</sup> International Workshop on Mobility Management and Wireless Access Protocols ACM MobiWac October 2004, pp. 10-18
- [10] J. Kim , S. Kim, S. Choi and D. Qiao, “ CARA: Collision Aware Rate Adaptation for IEEE 802.11 WLANs,” In Proceedings of 25<sup>th</sup> IEEE International Conference on Computer Communications 2006 INFOCOM April 2006, pp. 1-11
- [11] S.H.Y.Wong, H. Yang, S. Lu and V. Bharghavan, “Robust Rate Adaptation for 802.11 Wireless Networks,” In Proceedings of the 12<sup>th</sup> International

- Conference on Mobile Computing and Networking ACM MobiCom September 2006, pp.146-157
- [12] J. P. Pavon, B. Manor and S. Choi, "Link Adaptation Strategy for 802.11 WLAN via Received Signal Strength Measurement," In Proceedings of IEEE International Conference on Communications ICC '03 May 2003, Vol.2, pp.1108-1113
- [13] B. Sadeghi, V. Kanodia, A. Sabharwal and E. Knightly, "Opportunistic Media Access for Multirate Ad Hoc Networks," In Proceedings of the 8<sup>th</sup> Annual Conference on Mobile Computing and Networking ACM MobiCom September 2002, pp. 24-35
- [14] Y. Yang, M. Marina and R. Bagrodia, "Experimental Evaluation of Application Performance with 802.11 PHY Rate Adaptation Mechanisms in Diverse Conditions," In Proceedings of IEEE Wireless Communication and Networking Conference WCNC April 2006, pp. 2273-2278
- [15] D. Qiao and S. Choi, "Fast Responsive Link Adaptation for IEEE 802.11 WLANs," In Proceedings of IEEE international Conference on Communications ICC May 2005, Vol. 5, pp. 3583-3588
- [16] Q. Pang, V. C. M. Leung and S. C. Liew, "A Rate Adaptation Algorithm for IEEE 802.11 WLANs Based on MAC-Layer Loss Differentiation," In Proceedings of 2<sup>nd</sup> International Conference on Broadband Networks ICBN October 2005, Vol.1, pp. 659-667
- [17] S. Pal, S. R. Kundu, K. Basu and S. K. Das, "IEEE 802.11 Rate Control Algorithms: Experimental and Performance Evaluation in Infrastructure Mode," In Proceedings of Passive and Active Measurement Conference PAM March 2006
- [18] "Wireless tools for Linux," Available online on May 12, 2008 at [http://www.hpl.hp.com/personal/Jean\\_Tourrilhes/Linux/Tools.html](http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html)
- [19] "Iperf v.2.0.2," Available online on May 12, 2008 at <http://dast.nlanr.net/Projects/Iperf>
- [20] "PROPSim C2 Channel Simulator," Available online on May 12, 2008 at <http://www.propsim.net>
- [21] G. Bianchi, "Performance analysis of the IEEE 802.11 distributed coordination function", IEEE Journal on Selected Areas in Communications March 2000, Vol. 18, No. 3, pp. 535-547

- [22] Whitepaper “802.11n: Next Generation Wireless LAN Technology,” Available online on May 12, 2008 at <http://www.broadcom.com>
- [23] Z. Wu, S. Ganu, I. Seskar and D. Raychaudhuri, “Experimental Investigation of PHY Layer Control and Frequency Selection in 802.11 based Ad-Hoc Networks,” In Proceedings of Workshop on Experimental Approaches to Wireless Network Design and Analysis SIGCOMM 2005, pp. 41-45
- [24] K. Ramachandran, H. Kremo, M. Gruteser, P. Spasojevic and I. Seskar, “Scalability Analysis of Rate Adaptation Techniques in Congested IEEE 802.11 Networks: An ORBIT Testbed Comparative Study,” In Proceedings of IEEE symposium on World of Wireless, Mobile and Multimedia Networks WoWMoM June 2007, pp. 18-21
- [25] A. Kamerman and L. Monteban, “WaveLAN-II: A High Performance Wireless LAN for Unlicensed Band,” Bell Labs Technical Journal March 1997, Vol. 2, No. 3, pp 118-133
- [26] Z. Li, A. K. Gupta and S. Nandi, “A Full Auto Rate (FAR) MAC Protocol for Wireless Ad Hoc Networks,” IEE Proceedings on Communications June 2005, Vol. 152, No. 3, pp.311-319

# APPENDIX A

## Channel Models

### A.1 Channel Model A for NLOS conditions with 50 ns rms delay spread

| Tap Number | Delay (ns) | Delay Function | Amplitude (dB) | Amplitude Distribution | Distribution Parameter | Doppler Spectrum |
|------------|------------|----------------|----------------|------------------------|------------------------|------------------|
| 1          | 0          | Constant       | 0.0            | Rayleigh               |                        | Classical        |
| 2          | 10         | Constant       | -0.9           | Rayleigh               |                        | Classical        |
| 3          | 20         | Constant       | -1.7           | Rayleigh               |                        | Classical        |
| 4          | 30         | Constant       | -2.6           | Rayleigh               |                        | Classical        |
| 5          | 40         | Constant       | -3.5           | Rayleigh               |                        | Classical        |
| 6          | 50         | Constant       | -4.3           | Rayleigh               |                        | Classical        |
| 7          | 60         | Constant       | -5.2           | Rayleigh               |                        | Classical        |
| 8          | 70         | Constant       | -6.1           | Rayleigh               |                        | Classical        |
| 9          | 80         | Constant       | -6.9           | Rayleigh               |                        | Classical        |
| 10         | 90         | Constant       | -7.8           | Rayleigh               |                        | Classical        |
| 11         | 110        | Constant       | -4.7           | Rayleigh               |                        | Classical        |
| 12         | 140        | Constant       | -7.3           | Rayleigh               |                        | Classical        |
| 13         | 170        | Constant       | -9.9           | Rayleigh               |                        | Classical        |
| 14         | 200        | Constant       | -12.5          | Rayleigh               |                        | Classical        |
| 15         | 240        | Constant       | -13.7          | Rayleigh               |                        | Classical        |
| 16         | 290        | Constant       | -18.0          | Rayleigh               |                        | Classical        |
| 17         | 340        | Constant       | -22.4          | Rayleigh               |                        | Classical        |
| 18         | 390        | Constant       | -26.7          | Rayleigh               |                        | Classical        |

**A.2** Channel Model B for NLOS conditions with 100 ns rms delay spread

| Tap Number | Delay (ns) | Delay Function | Amplitude (dB) | Amplitude Distribution | Distribution Parameter | Doppler Spectrum |
|------------|------------|----------------|----------------|------------------------|------------------------|------------------|
| 1          | 0          | Constant       | -2.6           | Rayleigh               |                        | Classical        |
| 2          | 10         | Constant       | -3.0           | Rayleigh               |                        | Classical        |
| 3          | 20         | Constant       | -3.5           | Rayleigh               |                        | Classical        |
| 4          | 30         | Constant       | -3.9           | Rayleigh               |                        | Classical        |
| 5          | 50         | Constant       | 0.0            | Rayleigh               |                        | Classical        |
| 6          | 80         | Constant       | -1.3           | Rayleigh               |                        | Classical        |
| 7          | 110        | Constant       | -2.6           | Rayleigh               |                        | Classical        |
| 8          | 140        | Constant       | -3.9           | Rayleigh               |                        | Classical        |
| 9          | 180        | Constant       | -3.4           | Rayleigh               |                        | Classical        |
| 10         | 230        | Constant       | -5.6           | Rayleigh               |                        | Classical        |
| 11         | 280        | Constant       | -7.7           | Rayleigh               |                        | Classical        |
| 12         | 330        | Constant       | -9.9           | Rayleigh               |                        | Classical        |
| 13         | 380        | Constant       | -12.1          | Rayleigh               |                        | Classical        |
| 14         | 430        | Constant       | -14.3          | Rayleigh               |                        | Classical        |
| 15         | 490        | Constant       | -15.4          | Rayleigh               |                        | Classical        |
| 16         | 560        | Constant       | -18.4          | Rayleigh               |                        | Classical        |
| 17         | 640        | Constant       | -20.7          | Rayleigh               |                        | Classical        |
| 18         | 730        | Constant       | -24.6          | Rayleigh               |                        | Classical        |



**A.3** Channel Model C for NLOS conditions with 150 ns rms delay spread

| Tap Number | Delay (ns) | Delay Function | Amplitude (dB) | Amplitude Distribution | Distribution Parameter | Doppler Spectrum |
|------------|------------|----------------|----------------|------------------------|------------------------|------------------|
| 1          | 0          | Constant       | -3.3           | Rayleigh               |                        | Classical        |
| 2          | 10         | Constant       | -3.6           | Rayleigh               |                        | Classical        |
| 3          | 20         | Constant       | -3.9           | Rayleigh               |                        | Classical        |
| 4          | 30         | Constant       | -4.2           | Rayleigh               |                        | Classical        |
| 5          | 50         | Constant       | 0.0            | Rayleigh               |                        | Classical        |
| 6          | 80         | Constant       | -0.9           | Rayleigh               |                        | Classical        |
| 7          | 110        | Constant       | -1.7           | Rayleigh               |                        | Classical        |
| 8          | 140        | Constant       | -2.6           | Rayleigh               |                        | Classical        |
| 9          | 180        | Constant       | -1.5           | Rayleigh               |                        | Classical        |
| 10         | 230        | Constant       | -3.0           | Rayleigh               |                        | Classical        |
| 11         | 280        | Constant       | -4.4           | Rayleigh               |                        | Classical        |
| 12         | 330        | Constant       | -5.9           | Rayleigh               |                        | Classical        |
| 13         | 400        | Constant       | -5.3           | Rayleigh               |                        | Classical        |
| 14         | 490        | Constant       | -7.9           | Rayleigh               |                        | Classical        |
| 15         | 600        | Constant       | -9.4           | Rayleigh               |                        | Classical        |
| 16         | 730        | Constant       | -13.2          | Rayleigh               |                        | Classical        |
| 17         | 880        | Constant       | -16.3          | Rayleigh               |                        | Classical        |
| 18         | 1050       | Constant       | -21.2          | Rayleigh               |                        | Classical        |

**A.4** Channel Model D for LOS conditions with 140 ns rms delay spread

| Tap Number | Delay (ns) | Delay Function | Amplitude (dB) | Amplitude Distribution | Distribution Parameter | Doppler Spectrum |
|------------|------------|----------------|----------------|------------------------|------------------------|------------------|
| 1          | 0          | Constant       | 0.0            | Rice                   | K = 10                 | Classical        |
| 2          | 10         | Constant       | -10.0          | Rayleigh               |                        | Classical        |
| 3          | 20         | Constant       | -10.3          | Rayleigh               |                        | Classical        |
| 4          | 30         | Constant       | -10.6          | Rayleigh               |                        | Classical        |
| 5          | 50         | Constant       | -6.4           | Rayleigh               |                        | Classical        |
| 6          | 80         | Constant       | -7.2           | Rayleigh               |                        | Classical        |
| 7          | 110        | Constant       | -8.1           | Rayleigh               |                        | Classical        |
| 8          | 140        | Constant       | -9.0           | Rayleigh               |                        | Classical        |
| 9          | 180        | Constant       | -7.9           | Rayleigh               |                        | Classical        |
| 10         | 230        | Constant       | -9.4           | Rayleigh               |                        | Classical        |
| 11         | 280        | Constant       | -10.8          | Rayleigh               |                        | Classical        |
| 12         | 330        | Constant       | -12.3          | Rayleigh               |                        | Classical        |
| 13         | 400        | Constant       | -11.7          | Rayleigh               |                        | Classical        |
| 14         | 490        | Constant       | -14.3          | Rayleigh               |                        | Classical        |
| 15         | 600        | Constant       | -15.8          | Rayleigh               |                        | Classical        |
| 16         | 730        | Constant       | -19.6          | Rayleigh               |                        | Classical        |
| 17         | 880        | Constant       | -22.7          | Rayleigh               |                        | Classical        |
| 18         | 1050       | Constant       | -27.6          | Rayleigh               |                        | Classical        |

# APPENDIX B

## Throughput Results

### B.1 Average Throughput vs. Pathloss for UDP traffic with Channel Model A

| Pathloss Value<br>(dB) | Average Throughput (Mbps) |      |      |
|------------------------|---------------------------|------|------|
|                        | SampleRate                | Onoe | AMRR |
| 60                     | 6.82                      | 7.05 | 4.47 |
| 63                     | 6.64                      | 6.93 | 4.32 |
| 66                     | 6.64                      | 6.79 | 3.14 |
| 69                     | 6.26                      | 6.35 | 2.82 |
| 72                     | 5.48                      | 5.78 | 2.15 |
| 75                     | 4.38                      | 4.90 | 1.80 |
| 78                     | 3.06                      | 1.35 | 1.42 |
| 81                     | 1.70                      | 0.53 | 0.69 |
| 84                     | 0.65                      | 0.32 | 0.53 |

### B.2 Average Throughput vs. Pathloss for UDP traffic with Channel Model B

| Pathloss Value<br>(dB) | Average Throughput (Mbps) |      |      |
|------------------------|---------------------------|------|------|
|                        | SampleRate                | Onoe | AMRR |
| 60                     | 5.80                      | 6.76 | 3.95 |
| 63                     | 5.72                      | 6.62 | 3.75 |
| 66                     | 5.34                      | 6.26 | 2.99 |
| 69                     | 4.99                      | 5.48 | 2.86 |
| 72                     | 4.39                      | 4.84 | 2.70 |
| 75                     | 3.65                      | 3.40 | 2.20 |
| 78                     | 2.76                      | 1.75 | 1.45 |
| 81                     | 1.71                      | 0.48 | 0.66 |
| 84                     | 0.58                      | 0.34 | 0.56 |

**B.3** Average Throughput vs. Pathloss for UDP traffic with Channel Model C

| Pathloss Value<br>(dB) | Average Throughput (Mbps) |      |      |
|------------------------|---------------------------|------|------|
|                        | SampleRate                | Onoe | AMRR |
| 60                     | 3.75                      | 4.49 | 2.74 |
| 63                     | 3.58                      | 4.44 | 2.77 |
| 66                     | 3.43                      | 4.08 | 2.75 |
| 69                     | 3.16                      | 3.04 | 2.25 |
| 72                     | 2.95                      | 2.70 | 2.22 |
| 75                     | 2.40                      | 0.91 | 1.07 |
| 78                     | 1.91                      | 0.46 | 0.69 |
| 81                     | 1.28                      | 0.38 | 0.64 |
| 84                     | 0.57                      | 0.27 | 0.51 |

**B.4** Average Throughput vs. Pathloss for UDP traffic across Channel Model D

| Pathloss Value<br>(dB) | Average Throughput (Mbps) |      |      |
|------------------------|---------------------------|------|------|
|                        | SampleRate                | Onoe | AMRR |
| 60                     | 4.60                      | 6.36 | 4.35 |
| 63                     | 4.55                      | 6.29 | 4.28 |
| 66                     | 4.28                      | 6.06 | 3.88 |
| 69                     | 3.70                      | 5.40 | 2.72 |
| 72                     | 3.53                      | 4.28 | 2.75 |
| 75                     | 2.96                      | 2.91 | 2.33 |
| 78                     | 2.37                      | 1.15 | 1.88 |
| 81                     | 1.63                      | 0.50 | 0.82 |
| 84                     | 0.91                      | 0.29 | 0.54 |

**B.5:** Average Throughput vs. Pathloss for TCP traffic with Channel Model A

| Pathloss Value<br>(dB) | Average Throughput (Mbps) |       |      |
|------------------------|---------------------------|-------|------|
|                        | SampleRate                | Onoe  | AMRR |
| 60                     | 5.45                      | 4.76  | 2.72 |
| 63                     | 5.24                      | 4.70  | 2.72 |
| 66                     | 5.17                      | 4.56  | 2.57 |
| 69                     | 4.66                      | 4.24  | 2.03 |
| 72                     | 4.15                      | 3.80  | 2.02 |
| 75                     | 3.38                      | 3.30  | 1.97 |
| 78                     | 2.19                      | 1.45  | 1.22 |
| 81                     | 1.14                      | 0.35  | 0.06 |
| 84                     | 0.54                      | 0.004 | 0    |

**B.6** Average Throughput vs. Pathloss for TCP traffic with Channel Model B

| Pathloss Value<br>(dB) | Average Throughput (Mbps) |      |      |
|------------------------|---------------------------|------|------|
|                        | SampleRate                | Onoe | AMRR |
| 60                     | 4.41                      | 4.19 | 2.47 |
| 63                     | 4.26                      | 4.02 | 2.41 |
| 66                     | 4.01                      | 3.79 | 2.44 |
| 69                     | 3.63                      | 3.49 | 2.13 |
| 72                     | 3.22                      | 3.14 | 1.67 |
| 75                     | 2.71                      | 2.45 | 1.59 |
| 78                     | 1.78                      | 0.75 | 0.99 |
| 81                     | 0.96                      | 0.15 | 0.01 |
| 84                     | 0.19                      | 0.05 | 0    |

**B.7** Average Throughput vs. Pathloss for TCP traffic with Channel Model C

| Pathloss Value<br>(dB) | Average Throughput (Mbps) |      |      |
|------------------------|---------------------------|------|------|
|                        | SampleRate                | Onoe | AMRR |
| 60                     | 2.95                      | 2.60 | 2.49 |
| 63                     | 2.88                      | 2.55 | 2.46 |
| 66                     | 2.64                      | 2.34 | 2.14 |
| 69                     | 2.34                      | 1.77 | 1.03 |
| 72                     | 1.95                      | 0.40 | 0.21 |
| 75                     | 1.44                      | 0.27 | 0.03 |
| 78                     | 0.94                      | 0.10 | 0.03 |
| 81                     | 0.49                      | 0    | 0    |
| 84                     | 0.03                      | 0    | 0    |

**B.8** Average Throughput vs. Pathloss for TCP traffic with Channel Model D

| Pathloss Value<br>(dB) | Average Throughput (Mbps) |      |       |
|------------------------|---------------------------|------|-------|
|                        | SampleRate                | Onoe | AMRR  |
| 60                     | 4.09                      | 3.73 | 3.10  |
| 63                     | 3.88                      | 3.54 | 2.67  |
| 66                     | 3.60                      | 3.42 | 2.66  |
| 69                     | 3.09                      | 2.88 | 2.15  |
| 72                     | 2.73                      | 2.11 | 1.64  |
| 75                     | 2.22                      | 0.98 | 1.42  |
| 78                     | 1.65                      | 0.37 | 0.19  |
| 81                     | 0.74                      | 0.04 | 0.002 |
| 84                     | 0.02                      | 0.01 | 0.001 |

# APPENDIX C

## MadWiFi Driver Details

### C.1 MadWiFi Driver Installation on Linux Family of OS

The following procedure is followed for installation of MadWiFi driver in Linux.

First of all download a MadWiFi driver release from the website and unpack it.

#### C.1.1 Remove the old modules

For removing the old modules one must be logged on as a root. First, set all the MadWiFi devices down

```
ifconfig ath0 down
```

```
ifconfig wifi0 down
```

After setting the devices down, run the following scripts to remove the current modules from the system and its memory. One must be in the MadWiFi directory to run the scripts

```
cd scripts
```

```
./Madwifi-unload.bash
```

```
./find-madwifi-modules.sh $(uname -r)
```

```
cd..
```

A prompt message asking if you really want to remove the modules will appear.

#### C.1.2 Compiling the MadWiFi code

For compiling the MadWiFi code into the actual driver, just type the following assuming one is in the MadWiFi directory

```
make
```

This will start the build process. After it finishes just check for any error messages. Make sure there are no error messages.

### **C.1.3 Installing the MadWiFi**

After the build process just type the following for installing the MadWiFi

```
make install
```

This will copy all the modules, tools and man pages to the correct directories.

### **C.1.4 Loading the MadWiFi Module**

Finally load the MadWiFi driver module into the running system which is essential for all the other softwares to talk to the MadWiFi hardware

```
modprobe ath_pci
```

## **C.2 Changing the Rate Adaptation Algorithm**

The following procedure is followed for changing the Rate Adaptation Algorithm in the MadWiFi driver. We will take an example of changing from SampleRate to AMRR which can be followed for changing according the requirements.

Make sure the devices are down

```
ifconfig ath0 down
```

### **C.2.1 Removing the Modules**

Remove the following modules related to SampleRate

```
rmmod ath_pci
```

```
rmmod ath_rate_sample
```

### **C.2.2 Loading the Module**

Load the following modules by just typing

```
modprobe ath_pci ratectl=amrr
```

List the modules loaded in the kernel by typing

```
lsmod
```

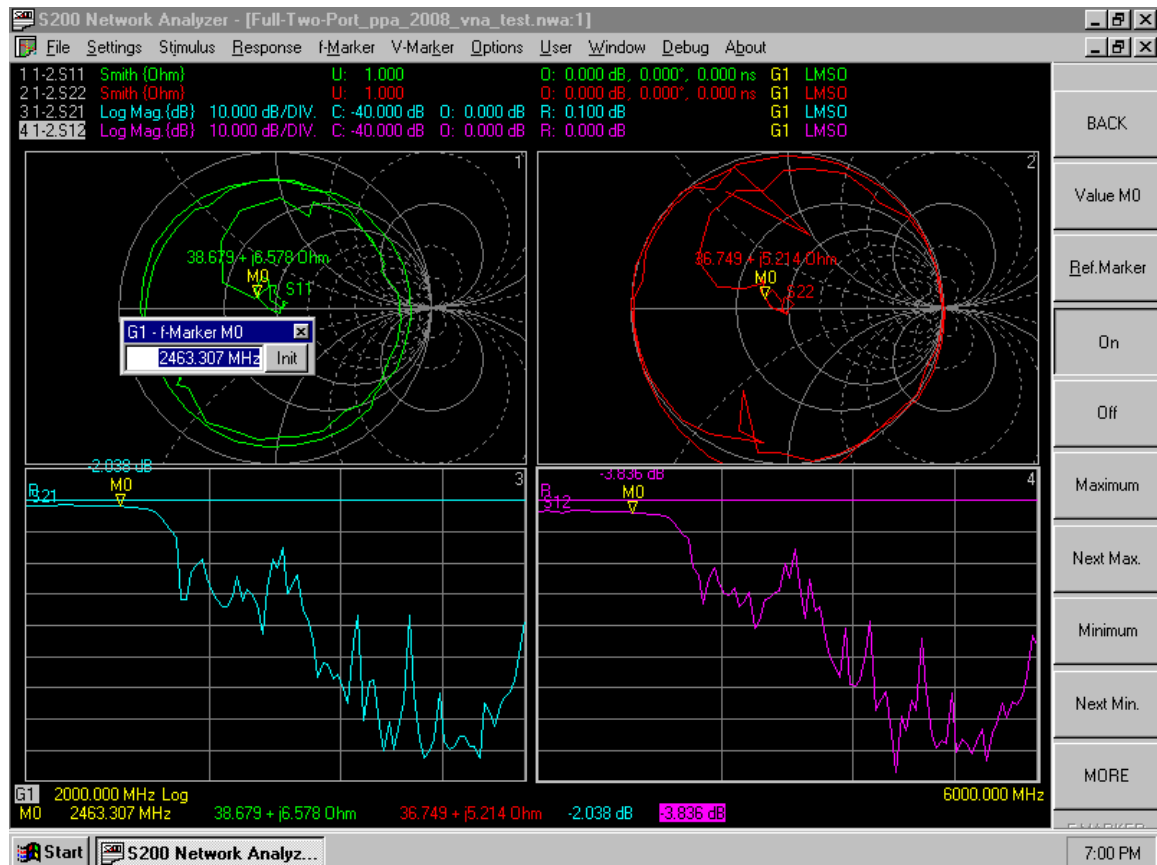
And make sure the new module is loaded i.e.

```
ath_rate_amrr
```



# APPENDIX D

## D.1: Experimental Setup calibration via Network Analyzer



The figure above shows the attenuation value of the connection setup including the RF cables, circulators and connectors. The test was performed as a full two port connection with signal generated and received at two ports. The S12 indicates signal generated at port 1 and received at port 2 and S21 indicates vice versa. The following values were set for testing

Signal Power = 0 dB

Frequency Range = 2000 MHz to 6000 MHz

The attenuation value at the Marker value of 2463.307 MHz is highlighted and gives the value as

S21 = -2.038 dB

S12 = -3.836 dB