Cross layer interactions for adaptive communications in IEEE 802.11 wireless LANs

Mohammad Hossein Manshaei

Université de Nice - Sophia Antipolis
École Doctorale STIC

THÈSE

Présentée pour obtenir le titre de :

Docteur en Sciences de l'Université de Nice - Sophia Antipolis

Spécialité : INFORMATIQUE

par

Mohammad Hossein MANSHAEI

Équipe d'accueil : Planète – INRIA Sophia Antipolis

CROSS LAYER INTERACTIONS FOR ADAPTIVE COMMUNICATIONS IN IEEE 802.11 WIRELESS LANS

Thèse dirigée par Walid DABBOUS et Thierry TURLETTI

Soutenue publiquement à l'INRIA le 14 Décembre 2005 à 10h00
devant le jury composé de :

Président :	Pierre	BERNHARD	UNSA/UFR ESSI, France
Co-directeurs :	Walid	DABBOUS	INRIA, France
	Thierry	TURLETTI	INRIA, France
Rapporteurs :	Mohamed	NAIMI	University of Cergy-Pontoise, France
	Michele	ZORZI	University of Padova, Italy
Examinateurs :	Isabelle	GUERIN-LASSOUS	INRIA, France
	Kavé	SALAMATIAN	Pierre & Mary Curie University/LIP6, France

Thèse

Mécanismes d'Interactions Intercouches pour Améliorer la Performance des Réseaux Locaux Sans Fil IEEE 802.11

Cross Layer Interactions for Adaptive Communications in IEEE 802.11 Wireless LANs

Mohammad Hossein Manshaei

December 2005

To Zeinab
and
My Parents

CROSS LAYER INTERACTIONS FOR ADAPTIVE COMMUNICATIONS IN IEEE 802.11 WIRELESS LANS

Mohammad Hossein Manshaei
Thesis Supervisors: Thierry Turletti et Walid Dabbous
Planète Project, INRIA Sophia Antipolis, France

ABSTRACT

Currently, the IEEE 802.11 is the de facto standard for *wireless local area networks* (WLAN). It specifies both the *medium access control* (MAC) and the *physical* (PHY) layers for WLANs. Regarding to cross layer mechanisms and adaptive communications over WLANs, there are many new and significant challenges with respect to wired and traditional wireless networks. In fact, as soon as we want to optimize data transmission according to both the characteristics of the data and to the varying channel conditions, a cross-layering approach becomes necessary. The main goal of this thesis is to propose efficient adaptive communication mechanisms using cross layer interactions in WLANs.

After a short overview of MAC and PHY layer protocols in the first part of the thesis, we present a detailed performance evaluation of IEEE 802.11a/b PHY layer transmission modes. While a lot of performance studies have been done for 802.11 MAC layer standards, very few analysis of the 802.11 transmission modes considering all physical layer building blocks are available so far. This PHY layer evaluation has been done using AWGN and Rayleigh fading channel models. We have taken into account the multipath receiver and Viterbi decoder at the receiver device as well. Based on simulation results, we show that the obtained performance is highly dependent on several factors such as, the modulation scheme, the signal detection algorithm, the number of receiver antennas, the type of diversity combining and the severity of multipath fading. Our performance evaluation confirms that physical layer FEC (i.e., binary convolutional codes) does increase significantly the performance of the 802.11a WLAN devices.

The second contribution of the thesis concerns IEEE 802.11 WLANs modelling (i.e., cross layer modelling in 802.11). An analytical model that accounts for the positions of stations with respect to the *access point* (AP) while evaluating the performance of 802.11 MAC layer (PHY/MAC modelling), has been proposed. In this model, given the position of one station, it computes the saturation throughput of wireless network, while conditioning on the positions of the other concurrent stations. Further, this model provides the total saturation throughput of the medium. The model which has been solved numerically has shown that the saturation throughput per station is strongly dependent not only on the position of stations but also on the positions of the other stations. It can also be used to dimension 802.11 wireless access networks and to study their capacities and performances.

The third contribution of the thesis concerns rate adaptation mechanisms and especially cross layer algorithms between MAC and PHY layers. After some practical experiments with various 802.11 wireless devices, we identified some important characteristics of the 802.11 systems that must be taken into account to design rate selection mechanisms. In particular, we emphasized the contrasts between low latency and high latency systems. An adaptive rate selection algorithm, called *adaptive auto rate fallback* (AARF) for low latency systems that improves upon *auto rate fallback* (ARF) to provide both short-term and long-term adaptation has been proposed. This new algorithm has very low complexity while obtaining a performance similar to more complex rate selection algorithms like RBAR, which requires incompatible changes to the 802.11 MAC and PHY protocol as well. In this field, we also presented a new rate adaptation algorithm designed for high latency systems that has been implemented and evaluated on an AR5212-based device. Experimentation results showed a clear performance improvement over the algorithm implemented in the AR5212 driver. In this part of thesis, we also proposed a closed-loop, dynamic rate selection algorithm that can be implemented in all IEEE 802.11 a/b/g compliant wireless local area networks. This algorithm called *closed-loop adaptive rate allocation* (CLARA) is a culmination

of the best attributes of the transmitter-based ARF and the RBAR control mechanisms with additional practical features to facilitate multipath fading channel sensing and feedback control signalling.

The last contribution of the thesis is on the optimization of real time multimedia transmission over IEEE 802.11 based networks. In particular, we proposed a simple and efficient cross layer mechanism, called *media oriented rate selection algorithm* (MORSA), for dynamically selecting the transmission mode considering both the channel conditions and characteristics of the media. MORSA is an adaptive communication mechanism which uses a simple cross layer approach between application, MAC, and PHY layers. An evaluation of this mechanism for *mobile ad hoc networks* (MONETs) has been provided using simulations with ns-2. The video quality obtained with a fine grain scalable video encoder based on a motion-compensated spatio-temporal wavelet transform has been analyzed as well. The transmission of a sample video flow over an 802.11a wireless channel has been evaluated with MORSA and compared with the traditional approach and significant improvement has been observed in throughput, latency and jitter while keeping a good level of video quality.

MÉCANISMES D'INTERACTIONS INTERCOUCHES POUR AMÉLIORER LA PERFORMANCE DES RÉSEAUX LOCAUX SANS FIL IEEE 802.11

Mohammad Hossein Manshaei
Directeur de thèse: Thierry Turletti et Walid Dabbous
Projet Planète, INRIA Sophia Antipolis, France

RÉSUMÉ EN FRANÇAIS

Le standard IEEE 802.11 est devenu la référence de facto pour les réseaux locaux sans fil par son très large déploiement dans le monde. L'extrême variabilité des caractéristiques du canal de transmission propre au standard 802.11 et l'impossibilité de pouvoir réserver des ressources posent de nombreux problèmes de recherche intéressants. Pouvoir optimiser de manière dynamique la transmission des données en prenant en compte à la fois les besoins des différentes applications et les caractéristiques très variables du médium est un challenge qui demande des interactions nouvelles entre différentes couches de la pile de communication. L'objectif de cette thèse est de montrer comment on peut tirer profit de ces nouvelles interactions pour concevoir de nouveaux algorithmes de transmission adaptatifs et efficaces.

Après une description rapide des protocoles des couches liaison et physiques du standard IEEE 802.11, nous présentons une analyse de performance des modes de transmission spécifiés par les normes IEEE 802.11a et IEEE 802.11b. Bien que le standard IEEE 802.11 a fait l'objet de nombreuses études, très peu d'entre elles ont porté sur les performances des modes de transmission et de l'impact des algorithmes implantés dans la couche physique. Notre analyse de la couche physique se base sur des modèles de canal à bruit blanc additifs Gaussiens et des modèles d'évanouissement de Rayleigh. Nos résultats de simulation montrent que les performances obtenues sont très dépendantes de plusieurs facteurs au nombre desquels figurent le système de modulation, l'algorithme de détection du signal, le nombre d'antennes de réception, le type de recombinaison de la diversité et l'importance de l'évanouissement du signal sur des chemins multiples. Nous avons pu aussi confirmer que la présence de codes correcteurs d'erreurs dans la couche physique du standard IEEE 802.11a améliore considérablement les performances.

La deuxième contribution de cette thèse porte sur la modélisation des réseaux locaux sans fil IEEE 802.11. Nous proposons ainsi un modèle analytique qui prend en compte la position des terminaux radio par rapport au point d'accès pour évaluer les performances de la couche MAC. Ce modèle est capable de calculer le débit de saturation du réseau considéré en fonction de la position des terminaux et du point d'accès. La résolution numérique de ce modèle montre que le débit total de saturation par terminal dépend fortement non seulement de la position du terminal considéré mais aussi de la position des autres terminaux sans fil. Ce modèle peut ainsi être appliqué au problème de dimensionnement des réseaux locaux sans fil 802.11.

Notre troisième contribution concerne l'élaboration de mécanismes de sélection des modes de transmission, en utilisant des interactions entre la couche liaison et la couche physique. Une campagne de mesures nous a permis d'identifier les principales caractéristiques des terminaux 802.11 qu'il est nécessaire de prendre en compte pour concevoir des algorithmes de contrôle des modes de transmission. En particulier, nous avons identifié deux classes de terminaux: ceux dits à faible latence et ceux dits à grande latence et nous avons proposé des algorithmes adaptés à ces deux classes de terminaux. Nous avons élaboré l'algorithme AARF (Adaptive Auto Rate Fallback) qui est basé sur l'algorithme ARF (Auto Rate Fallback). Ce nouvel algorithme de sélection de modes de transmission s'intéresse à la classe des terminaux dits à faible latence et offre des services d'adaptation à court et à long terme. La complexité de cet algorithme est très faible mais il permet d'obtenir des performances très similaires à celles d'algorithmes beaucoup plus complexes tel que RBAR, tout en restant compatible avec le standard 802.11. Un algorithme basé sur les mêmes idées que celles développées pour AARF mais conçu pour

fonctionner avec des systèmes à grande latence à été implanté et expérimenté sur des terminaux à base du processeur AR-5212 de la firme Atheros. Les expérimentations menées avec ce driver modifié confirment le gain en performance obtenu avec les simulations. Enfin, nous avons proposé un algorithme de contrôle dynamique du mode de transmission en boucle fermée CLARA (closed-loop adaptive rate allocation) qui peut être implanté de manière compatible aux standards 802.11a/b/g. Ce nouvel algorithme donne de bonnes performances en présence d'évanouissement du signal sur des chemins multiples et peut être implanté de manière compatible avec le standard 802.11.

Enfin, la dernière contribution de cette thèse porte sur l'optimisation de la transmission de flots multimédias temps réel sur des réseaux 802.11. Plus précisément, nous avons élaboré MORSA (media oriented rate selection algorithm), un mécanisme d'optimisation intercouche peu complexe et efficace qui permet de sélectionner le mode de transmission à utiliser en fonction de l'état du canal de transmission radio et des caractéristiques de l'application, et en particulier de sa tolérance éventuelle aux erreurs de transmission. MORSA est un mécanisme de communication adaptif qui utilise des interactions entre les couches application, liaison et physique. Nous avons évalué les performances de l'algorithme MORSA avec des simulations ns-2 pour des réseaux sans-fil ad hocs. Nous avons analysé la qualité de réception obtenue lorsque l'émetteur utilise un encodeur vidéo en couches à granularité fine. Tout en conservant un bon niveau de qualité, nous avons observé avec MORSA une amélioration conséquente du débit, et une diminution de la latence et de la gigue par rapport à l'approche classique.

ACKNOWLEDGMENTS

First of all, I want to thank my wife Zeinab, for all the supports that she gave me during these years, I have been working on this thesis. I could have never achieved this goal without her supports. I dedicate this thesis with affection to Zeinab, whose courage and patience have been an inspiration to me last years.

I would like to express my deep and sincere gratitude to my supervisor Dr. Thierry Turletti. His understanding and encouraging have provided a good basis for the present thesis. His comments have been of greatest help at all times. I wish to thank Dr. Walid Dabbous, for giving me the opportunity to come to INRIA and work in Planete group. I also acknowledge generous research grants from French ministry of science and INRIA to complete this thesis.

During past years, my interaction with the members and visitors of my research group at INRIA has been extremely beneficial. Many thanks go to Vijay Arya, Ali Boudani, Mohammad Malli, Abdelbasset Trad, Fethi Filali, Lamia Romdhani, Laurent Fazio, Mohammad Abid, Mohammad-Ali Kaafar, Musfiq Rahman, Alexis Gourdon, Rares Serban, Laurentiu Barza, Fatma Louati, Miguel A. Ruiz Sanchez, Prof. Torsten Braun, Dr. Arnaud Legout, Dr. Qiang Ni, and Dr. Hossam Afifi. I am also grateful to Aurelie Richard for all administration works.

I would like to thank all those who contributed to the success of this work as well. Many thanks go to Dr. Chadi Barakat for his valuable ideas and numerous discussions about DAW model presented in Chapter 4. I also thank Gion Reto Cantieni for his collaboration in this field.

Many thanks go to Mathieu Lacage for our many discussions and providing me many advices and tips during our collaborations concerning the physical rate adaptations presented in Chapter 5. I thank him and Thierry Parmentelat for their help in preparing the thesis French summary as well.

I would like to thank Dr. Ceilidh Hoffmann, who visited us for seven months, and with whom I had many discussions concerning the digital communications. The works presented in Chapter 3 and CLARA mechanism are done under her excellent guidance.

Many thanks go to Prof. Christine Guillemot and Dr. Thomas Guionnet from IRISA for providing their video encoder WAVIX to perform the evaluation of multimedia transmission over wireless LANs presented in Chapter 6. Many thanks go to Prof. Marwan Krunz from University of Arizona for profitable discussions during his sabbatical in INRIA and helps concerning MORSA mechanism presented in Chapter 6. I wish also to thank Ramin Khalili for all discussions that we had about the performance of Viterbi decoder.

Many thanks go to Hitoshi Asaeda to give me the opportunity to work as a research assistance in the framework of project about group member authentication for mobile Ad Hoc

networks. The results of this project is not presented in this manuscript.

In particular, I wish to express my appreciation to Dr. Kave Salamatian for his valuable suggestions and guidance during my PhD thesis.

I would also like to thank the members of my PhD committee who took effort in reading and providing me with valuable comments on earlier versions of this thesis: Prof. Michele Zorzi, Prof. Mohamed Naimi, Dr. Isabelle Guerin-Lassous, and Dr. Kave Salamatian. Many thanks go to Prof. Pierre Bernhard for serving as the chairman of jury committee.

During my stay at University of Padova for an internship supported by European Embedded WiSeNts project, I had the opportunity to associate with a number of truly remarkable people. I would like to deeply thank Prof. Michele Zorzi, Dr. Andrea Zanella, and Dr. Leonardo Badia for giving me the opportunity to work as part of their research group during the internship.

Last but not least, the special thanks go to my family: my parents, Nasrin, Saleh, Maedeh, Faezeh, and my wife's family for the unconditional love and support that they have showed over the years.

Mohammad Hossein Manshaei
manshaei@sophia.inria.fr
Sophia Antipolis, France

CONTENTS

List of Figures

LIST OF TABLES

1

INTRODUCTION

The focus of this thesis is on *cross layer interactions* for *adaptive communications* in IEEE 802.11 wireless LAN. We first present a performance analysis of the MAC and physical layer protocols in 802.11 wireless LANs with analytical models. Then, we propose cross layer interaction solutions between the MAC and physical layers to improve the performance of wireless devices. Specifically, we present novel physical rate selection mechanisms for 802.11 standards. The last part of this thesis deals with an adaptive communication protocol for video transmission over wireless LANs, with the help of cross layer interactions between the application, MAC, and physical layers.

1.1 Introduction to Wireless Networking

Wireless networking refers to the technology that enables two or more computers to communicate over radio, employing a network protocol. Mainly because of the recent advances in wireless networking technology, *wireless local area networks* (WLANs) are deployed almost everywhere. The WLANs can be set up in a very simple way to allow mobility and portability of computers and other devices located in almost all areas. This has enabled individuals to easily connect to the Internet from virtually anywhere as well.

Wireless network devices require the use of technologies that deal with radio frequencies and data transmission. The most widely employed standard is 802.11 which has been proposed and elaborated by the working group 11 of the *institute of electrical and electronic engineers* (IEEE) standard committee [1]. Table A.1 in Appendix A shows the active working groups and their tasks within the IEEE standard specification. The original version of this standard named *802.11 legacy* [1] and released in 1997, specifies two physical data rates (1 and 2

Mbps) and *carrier sense multiple access with collision avoidance* (CSMA/CA) for the media access method. Basically, the most popular techniques for over-the-air modulations are defined by 11a [2], 11b [3], and 11g [4] working groups which is why we will focus on them in this thesis. IEEE 802.11b was the first widely accepted wireless networking standard followed by 802.11a and 802.11g. The 802.11b/g standards use the unlicensed 2.4 GHz band while the 802.11a standard uses the 5 GHz band. The other standards in the family are service enhancement, extensions, or corrections to the previous specifications. The latest working group, i.e., 802.11n started in January 2004, is developing 802.11 standard which targets 540 Mbps as real data throughput. This standard will build upon previous 802.11 standards by adding *multiple-input multiple-output* (MIMO)[1] and *orthogonal frequency-division multiplexing* (OFDM).

1.2 Cross Layer Interactions

Typically, network protocols are divided into several layers. Each layer should be designed independently and usually the interactions between layers are defined by some specific interfaces. Of course, by creating many protocols independently of each other, designers have a much simpler task. For instance, the old protocols can be simply replaced by the new ones without modification in the rest of protocol stacks: this brings flexibility. But there are issues that naturally span many layers, especially in *wireless networks*.

For example in wireless networks, the physical link can no longer be considered as a separate entity whose performance is independent of other layers. Suppose that the physical layer selects one data rate based on physical channel conditions. This will then affect packet delay over the radio link. In the network layer, routing decisions will depend on the link delay (i.e., physical data rate or multihops connections). Thus the routing decision will depend on the physical channel conditions. In other words, the communications of various layers are somehow interrelated.

Another example could be the task of power control in the wireless devices. Generally, the higher the transmission power is, the higher the received throughput is. But unlike wired networks, in wireless environments the transmit power of a particular node affects all other nodes in the network. In particular, the transmission power changes the level of interference experienced by other nodes. This could directly change the routing policy as well. Thus the power control is completely coupled with the MAC, physical, and routing layers in wireless networks. Other examples can be cited to show the effects of cross layer interactions in wireless environment. Although, cross layer interactions usually make the design of protocols more complex, they are an opportunity for the system designers to increase efficiency of wireless

[1] MIMO uses multiple transmitter and receiver antennas to allow for increased data throughput through spatial multiplexing and increased range by exploiting the spatial diversity (see Section 3.5.3).

networks. Indeed, new mechanisms can be designed to span multiple layers of the protocol stack. For instance in this thesis, with very simple cross layer interactions between the MAC and physical layers to select the best physical transmission rate, a large increase of performance has been obtained without any significant overhead.

1.2.1 Motivations and Contributions

In order to design cross layer interaction mechanisms in WLANs, we primarily needed to focus on performance evaluation of the 802.11 MAC layer and different WLAN physical layers. Our evaluations have been done over AWGN, Rayleigh fading channel and multipath receiver for 802.11a/b/g standard specifications.

IEEE 802.11 MAC/PHY layer modelling

We propose an analytical model named *distance aware* (DAW), that accounts for the positions of stations with respect to the access point while evaluating the performance of the IEEE 802.11 MAC layer (MAC/PHY modelling) [5]. In this model, given the position of one station, it computes the saturation throughput of the wireless network, while conditioning on the positions of the other concurrent stations. This model is also helpful to dimension 802.11 wireless access networks and to study their capacities and performances.

IEEE 802.11 Rate Selection Mechanisms

We elaborate several rate selection mechanisms for the 802.11 standards (i.e., cross layer interactions between MAC and PHY layer). We discuss the key parameters of the 802.11 systems that must be taken into account to design an efficient rate selection mechanism. We propose three physical rate selection mechanisms named, *adaptive auto rare fallback* (AARF) [6], *closed-loop adaptive rate allocation* (CLARA) [7], and *adaptive multi rate retry* (AMRR) [6, 8]. AARF has very low complexity while obtaining performance similar to very complex rate selection mechanisms like RBAR. CLARA is a mechanism that takes into account the best attributes of the ARF and the RBAR rate selection mechanisms with additional practical features to facilitate multipath fading channel sensing and feedback control signalling. AMRR is a practical approach that has been implemented and evaluated on an AR5212-based device.

1.3 Adaptive Communications

The term *adaptive communications* refers to a communication system that automatically uses feedback information obtained from the system itself. This information which can be obtained from the signals carried by the system as well, is used to modify one or more system

operational parameters dynamically to improve system performance. This will help the system to resist against degradation as well. The modification of the system parameters can be discrete or continuous.

Adaptive communication mechanism is one of the constant feature of high-performance communication systems, because of the extreme variability of wireless channels. Adaptive communications are widely deployed in several wireless systems with dedicated channels, cellular systems, and multi-hop networks. For instance, adaptive communication aware routing protocols for wireless ad hoc networks, which use directional antenna [9] or adaptive modulation techniques for digital communication systems [10].

1.3.1 Motivations and Contributions

Generally, in existing WLAN environments, different protection strategies are used at the various layers of the protocol stack. For example, the *forward error correction* codes can be used in the application layer as well as the physical layer. Thus a cross-layer interaction is desirable in order to provide an optimal overall performance for the transmission of multimedia. Employing such techniques, very large performance improvements over static designs can be obtained.

Multimedia Transmission over IEEE 802.11

In this thesis, we propose a simple and novel cross layer interaction design implemented with minor modifications on the IEEE 802.11 standard, that could improve the network throughput significantly for *mobile ad hoc networks* (MONET). We work towards the optimization of real time multimedia transmission over IEEE 802.11 based networks [11, 12]. In particular, we propose a simple and efficient cross layer algorithm, named *media oriented rate selection algorithm* (MORSA), for dynamically selecting the transmission mode that considers both the channel conditions and characteristics of the media. We evaluate this mechanism for MONETs using simulations with NS-2. We analyze the video quality obtained with a fine grain scalable video encoder based on a motion-compensated spatio-temporal wavelet transform. The transmission of a sample video flow over an 802.11a wireless channel is evaluated with MORSA and compared with the traditional approach in this part of thesis.

1.4 Thesis Organization

The first part of this dissertation covers the design and performance analysis of data transmission in 802.11 wireless LANs. We first present the 802.11 MAC and physical layers in Chapter 2. CSMA/CA, 802.11e MAC layer, and the over-the-air modulations are discussed in

this chapter. Then in Chapter 3, we evaluate the performance of IEEE 802.11 data transmission over AWGN, Rayleigh fading, and multipath wireless channels.

The second part of this dissertation deals with the modelling of IEEE 802.11 standards. In Chapter 4, we first study the existing models for IEEE 802.11 WLANs and then propose and evaluate an analytical model for the MAC and physical layers in the IEEE 802.11 standard. The main contribution of this model is to take into account both the PHY and MAC layer protocols to analyze the performance of existing IEEE 802.11 standard.

The third part of this dissertation deals with issues concerning physical rate adaptation mechanisms. Typically, the rate adaptation mechanisms use the cross layer interactions between MAC and physical layers. In Chapter 5, we first enumerate the key parameters that should be taken into account to propose an efficient physical layer rate adaptation. We then survey related works in the area of physical rate adaptation. We describe the three physical rate adaptation mechanisms designed as a part of this dissertation work, named AARF, CLARA, and AMRR. This chapter also includes a performance analysis section, which is used to validate the mechanisms. For performance analysis, we have used NS-2, MATLAB, and experimental results.

The fourth part of this dissertation deals with adaptive communication over 802.11 WLANs. In Chapter 6 we present a simple and efficient cross-layer mechanism (between application, MAC, and PHY layers) for dynamically selecting the physical transmission mode considering both the channel conditions and characteristics of the media. We also provide a performance evaluation of this mechanism by analyzing the video quality obtained with a fine-grain scalable video encoder based on a motion-compensated spatio-temporal wavelet transform. We summarize and conclude this dissertation in Chapter 7.

Appendix A addresses the IEEE 802.11 working groups and their functionalities. Appendix B presents the available frequency of 802.11a/b wireless LANs. Finally, Appendices C and D detail the complete pseudo codes for the AARF and AMRR rate selection mechanisms.

2

IEEE 802.11 WIRELESS LAN OVERVIEW

2.1 Summary

Nowadays, the IEEE 802.11 WLAN technology offers the largest deployed wireless access to the Internet. This technology specifies both the *medium access control* (MAC) and the *physical layer* (PHY) protocols. The PHY layer aims to use the best modulation scheme given the channel conditions and provides the necessary bandwidth, whereas the MAC layer decides in a distributed manner on how the offered bandwidth is shared among all stations. This standard allows the same MAC layer to operate on top of one of several PHY layers. We present an overview of the MAC and PHY layer protocols in this chapter. Two medium access control protocols named basic and RTS/CTS are discussed. We describe the new MAC protocol to support QoS in WLANs as well, i.e., the EDCA and HCCA mechanisms defined in the IEEE 802.11e standard. We address the functionalities of all building blocks in 802.11a/b PHY layer protocols by presenting the data flow transmission in these standards. The channel identifiers and channel center frequencies in IEEE 802.11a/b standards are discussed in this chapter as well. The physical layer transmission techniques, binary convolutional codes (FEC), data modulations (differentially and classic BPSK and QPSK, QAM, and CCK), and carrier modulation are presented and discussed. We describe the functionalities of these blocks using a simple example for data flow transmission. Finally a short overview of frame format and packet encapsulation in 802.11 is presented.

2.2 Introduction

In recent years, high-speed *wireless local area networks* (WLANs) have become widely popular in various sectors, including health care, manufacturing and academic centers. These sectors benefited from the productivity gains of using hand-held terminals and notebook computers to transmit real-time information within physically distributed environments. Currently, IEEE 802.11 is the de facto standard for WLANs [1]. It specifies both the *medium access control* and the *physical* layers for WLANs. The scope of IEEE 802.11 *working groups* (WGs) is to propose and develop MAC and PHY layer specifications for WLAN to handle mobile as well as portable stations. A portable station is one that is moved from location to location, but that is only used while at a fixed location. Mobile stations actually access the LAN while in motion. In this standard, the MAC layer operates on top of one of several physical layers. Medium access is performed using *carrier sense multiple access with collision avoidance* (CSMA/CA).

The increasing number of wireless users and the demand for high-bandwidth multimedia applications over WLANs led the IEEE working groups to provide powerful physical layers and to extend the MAC layer to provide QoS support (i.e., IEEE 802.11e [13]). Concerning the physical layer, three IEEE 802.11 standards are currently available: a, b, and g. The 802.11b standard is the most widely deployed in todays WLANs [3]. Since the end of 2001, higher data rate products based on the 802.11a standard have appeared in the market [2]. More recently, the IEEE 802.11 working group has approved the 802.11g standard, which extends the data rate of the IEEE 802.11b to 54 Mbps [4]. The 802.11g specification offers transmission over relatively short distances at up to 54 Mbps. The 802.11g PHY layer employs all available modulations specified for 802.11a/b. So the performance evaluation of IEEE 802.11g is straightforward using the evaluation of 802.11a/b. For this reason, we will not focus on this standard specification in this dissertation.

To elaborate and deploy adaptive and cross layer mechanisms over IEEE 802.11 standard devices, a complete knowledge about the functionalities of these MAC and PHY layer protocols is required. Thus, in this chapter we overview the salient features of the IEEE 802.11a/b/e MAC and PHY layers. This chapter is organized as follow. We first present the reference model for the 802.11 standard in Section 2.3. We discuss the CSMA/CA protocol in basic mode and RTS/CTS optional mode in Section 2.4. A short introduction to the IEEE 82.11e MAC protocol which supports QoS in 802.11 is presented in Section 2.4.1. We then review all the PHY layer transmission techniques of the IEEE 802.11 standards which include *direct sequence spread spectrum*, *frequency hopping*, *infrared*, and *orthogonal frequency division multiplexing* in Section 2.5. After presenting a transmission model for the 802.11 PHY layers in Section 2.6, we discuss the functionalities of the PHY layer building blocks in the 802.11a/b standards in Section 2.7 and Section 2.8. Finally, we discuss the data and control frame formats in Section 2.9 followed by

the conclusions in Section 2.10.

2.3 IEEE 802.11 Reference Model

Figure 2.1 shows the reference model of the IEEE 802.11 architecture [1]. All PHY layers consist of two sublayers and two management entities. The *physical medium dependent* (PMD) sublayer defines characteristics of wireless medium and performs data encoding and modulation as well. The *physical layer convergence procedure* (PLCP) sublayer allows the MAC to operate with minimum dependence on the physical characteristics of the wireless medium. The PLCP sublayer also sets up the frame called *PHY protocol data unit* (PPDU) using the information provided by MAC layer. The payload part of the PPDU frame is called *MAC protocol data unit* (MPDU).

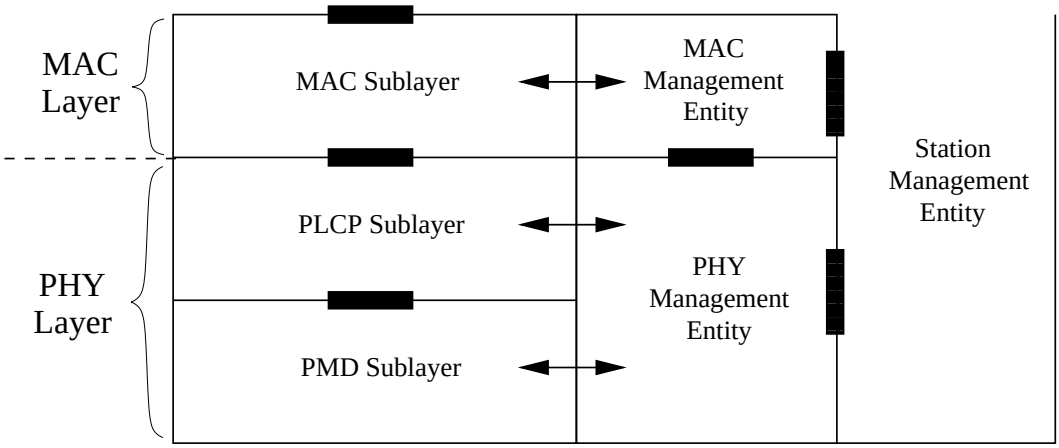


Figure 2.1: Reference model of PHY and MAC layer covered by IEEE 802.11 standards.

The MAC layer communicates with the PHY layer using PLCP via specific primitives through a PHY service access point. Management entities (for each layer) perform the management of the PHY and MAC layer. Generally, the IEEE 802.11 WG addresses the following issues [1]:

- Functions required for an 802.11 compliant device to operate either in a peer-to-peer (ad hoc) fashion or integrated with an existing wired LAN.
- Operations of the 802.11 device within possibly overlapping 802.11 wireless LANs and the mobility of this device between multiple wireless LANs.
- MAC level access control and data delivery services.
- Several physical layer signaling techniques and interfaces.

- Privacy and security of user data being transferred over the wireless media.

In general, the wireless networking can be implemented in two significantly different operating modes: *ad hoc* and *infrastructure* modes. The *infrastructure* mode consists of an *access point* (AP) acting as a hub for the network with each client communicating through it. This mode is generally for larger networks which may include sub-networks consisting of more than one access point. This means that an infrastructure network is more expensive to setup and usually requires more advanced configuration.

Ad-hoc mode essentially eliminates the need for an access point. In this mode, the mobile nodes can be connected dynamically in an arbitrary manner. All nodes of these networks behave as routers and take part in discovery and maintenance of routes to other nodes in the network. In other words, routing from one node to another requires an on-demand routing protocol, like DSR [14], AODV [15], or OLSR [16]. The following sections of this chapter focus on MAC layer protocol, different physical layer transmission techniques, QoS, and frame format in the 802.11 WLAN standard.

2.4 IEEE 802.11 MAC Layer

The *distributed coordination function* (DCF) is the basic medium access mechanism of IEEE 802.11, and uses a *carrier sense multiple access with collision avoidance* (CSMA/CA) algorithm to mediate the access to the shared medium. On the other hand, the *point coordination function* (PCF) is a centralized, polling-based access mechanism which requires the presence of a base station that acts as an access point. If PCF is supported, both PCF and DCF coexist and the time is divided into superframes. A superframe starts with a beacon frame. The AP generates beacon frames at regular beacon frame intervals. Thus, every station knows when the next beacon frame will arrive; this time is called *target beacon transition time* (TBTT) and is announced in every beacon frame.

Each superframe consists of a contention period where DCF is used and a *contention free period* (CFP) where PCF is used. During the CFP, the AP sends poll frames to high priority stations. Whenever a station is polled, it has the right to use the medium. To ensure that no station interrupts this mode of operation, the *interframe space* (IFS) between PCF data frames is shorter than the usual *DCF interframe space* (DIFS). This space is called a *PCF interframe space* (PIFS). To prevent starvation of low priority flows, the contention period must always be long enough for at least the transmission of one maximum frame length. The AP at any point during CFP period can suspend the PCF mode and return to DCF mode by sending a CF-end frame.

The DCF protocol in IEEE 802.11 standard defines how the medium is shared among stations. DCF is based on CSMA/CA [1]. It includes a basic access method and an optional

channel access method with *request-to-send* (RTS) and *clear-to-send* (CTS) exchanged as shown in Figure 2.2 and 2.3 respectively. First, we explain the basic access method.

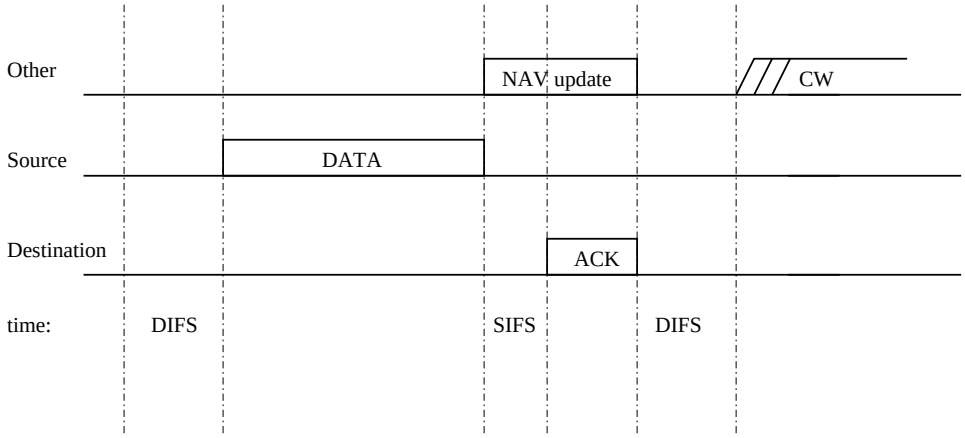


Figure 2.2: Basic access CSMA/CA protocol in DCF.

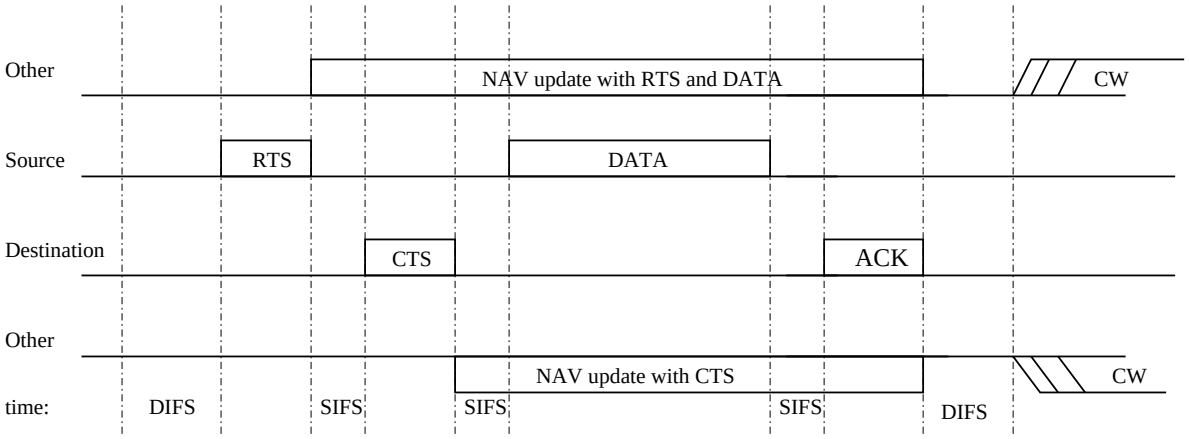


Figure 2.3: RTS/CTS exchange in CSMA/CA protocol.

If the channel is busy for the source STA, a backoff time (measured in slot times[1]) is chosen randomly in the interval $[0, CW)$, where CW is called the *contention window*. This timer is decremented by one as long as the channel is sensed idle for a DIFS, i.e., *distributed inter frame space* time. DIFS is equal to $\mathrm{SIFS} + 2 \times \mathrm{SlotTime}$. It stops when the channel is busy and resumes when the channel is idle again for at least a DIFS period. CW is an integer whose range is determined by the PHY layer characteristics: $CW_{\min}$ and $CW_{\max}$. CW is doubled after

[1]The slot time is the sum of the RX-to-TX turnaround time, MAC processing delay, and CCA detect time [1]. The value of slot time for different PHY layer protocols is shown in Table 2.1.

each unsuccessful transmission, up to the maximum value which is determined by $CW_{\max}+1$.

When the backoff timer reaches zero, the source transmits the data packet. The ACK is transmitted by the receiver immediately after a period of time called SIFS, i.e., *short inter frame space* time which is less than DIFS. When a data packet is transmitted, all other stations hearing this transmission adjust their *network allocation vector* (NAV), which is used for virtual *carrier sense* (CS) at the MAC layer. The NAV maintains a prediction of future traffic on the medium based on the duration information that is announced in Data frames (or RTS/CTS frames as will be explained in the following) prior to the actual exchange of data. In addition, whenever a node detects an erroneous frame, the node defers its transmission by a fixed duration indicated by EIFS, i.e., *extended inter frame space* time. This time is equal to the $\mathrm{SIFS}+\mathrm{ACK}_{\mathrm{time}}+\mathrm{DIFS}$ time.

If the optional access method is used, an RTS frame should be transmitted by the source and the destination should accept the data transmission by sending a CTS frame prior to the transmission of the actual data packet. Note that STAs in the sender's range that hear the RTS packet should update their NAVs and defer their transmissions for the duration specified by the RTS. Nodes that overhear the CTS packet update their NAVs and refrain from transmitting. This way, the transmission of the data packet and its corresponding ACK can proceed without interference from other nodes (hidden nodes problem). Table 2.1 shows the important time interval between frames in different standard specification called *inter frame space* (IFS) [2, 3, 4]. It should be considered that the IEEE 802.11g uses one of the IFS set based on its operating mode.

Table 2.1: Inter frame space and *CW* time for different PHY layers.

Parameters	*802.11a*	*802.11b (FH)*	*802.11b (DS)*	*802.11b (IR)*	*802.11b (High Rate)*
Slot Time (μs)	9	50	20	8	20
SIFS (μs)	16	28	10	10	10
DIFS (μs)	34	128	50	26	50
EIFS (μs)	92.6	396	364	205 or 193	268 or 364
$CW_{\min}(\mathrm{SlotTime})$	15	15	31	63	31
$CW_{\max}(\mathrm{SlotTime})$	1023	1023	1023	1023	1023

2.4.1 IEEE 802.11e MAC Protocol to Support QoS in WLANs

In Chapter 6, we propose a media oriented rate selection algorithm which is based on differentiated services. So, in this section, we briefly review the IEEE 802.11e standard which

aims to provide QoS in WLANs. The IEEE 802.11e draft [13] proposes many features to support QoS in WLANs. This standard includes an additional coordination function called *hybrid coordination function* (HCF) that is only usable in QoS network configurations. HCF shall be implemented in all *QoS-enhanced stations* (QSTAs). It combines functions from the DCF and PCF with some enhanced, QoS-specific mechanisms and frame subtypes. Usually, there is a centralized controller for other stations within the BSS which is called the *hybrid coordinator* (HC) and typically resides in the AP. HCF uses both a contention-based channel access method, called the *enhanced distributed channel access* (EDCA) and a controlled channel access, referred to as *HCF controlled channel access* (HCCA) mechanism, for contention free transfer.

Figure 2.4 shows EDCA in a QSTA. The EDCA mechanism provides differentiated and distributed access to the channel for QSTAs using 8 different *user priorities* (UPs). In EDCA, the probability of winning the contention can be differentiated among traffic types whereas the channel access unit can be defined based on the channel access time. The EDCA mechanism defines four *access categories* (ACs) that provide support for the delivery of traffic tagged with UPs at the QSTAs. User priorities are assigned by the application layer and are mapped to access categories based on a simple mapping table.

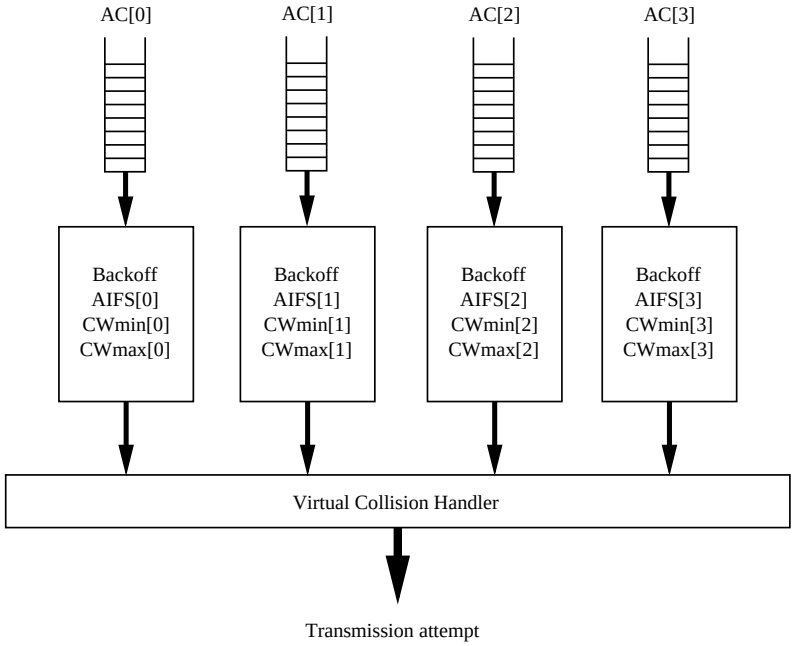


Figure 2.4: EDCA at IEEE 802.11e.

Table 2.2 shows that one or more UPs can be mapped to the same AC queues. Based on the IEEE 802.11e standard specification, this mapping [2] is done to reduce complexity. The EDCA

[2]In the first 802.11e draft there were 8 ACs. To simplify the implementation it has been reduced to 4 ACs [13].

still uses the same CDMA/CA to access the channel, but the channel access parameters are defined per AC. In other words, EDCA defines *arbitration interframe space* (AIFS) instead of a constance DCF time. EDCA allocates different CW sizes ($CW_{\min}$, $CW_{\max}$) for different ACs as well.

Table 2.2: Mapping between *user priority* and *access category*.

User priority	802.11D Designation	802.11e AC	Service type
2	Not defined	0	Best Effort
1	Background (BK)	0	Best Effort
0	Best Effort (BE)	0	Best Effort
3	Excellent Effort (EE)	1	Video Probe
4	Controlled Load (CL)	2	Video
5	VI (Video $<$ 100ms latency and jitter)	2	Video
6	VO (Video $<$ 10ms latency and jitter)	3	Voice
7	Network Control (NC)	3	Voice

The HCCA mechanism uses a HC which is a QoS-aware centralized coordinator. This controller is responsible for giving the *transmission opportunity time* (TXOP) to the user. In other words, once a station wins the contention, it will be assigned a TXOP period. During its TXOP the user can send a series of frames. Further information about these mechanisms is out of scope of this dissertation. A complete survey of QoS enhancements for 802.11 WLANs is available in the following references [17, 18, 19].

2.5 IEEE 802.11 Physical Layer Characteristics

As explained in Section 2.2, three different PHY layers are available for the IEEE 802.11 WLAN as shown in Table 2.3 [2][3][4]. IEEE 802.11b radios transmit at 2.4 GHz and send data up to 11 Mbps using *direct sequence spread spectrum* (DSSS), *infrared* (IR), and *frequency hopping* (FH) [3]; whereas IEEE 802.11a radios transmit at 5 GHz and send data up to 54 Mbps using *orthogonal frequency division multiplexing* (OFDM) [2]. The IEEE 802.11g standard [4], extends the data rate of the IEEE 802.11b to 54 Mbps in an upgraded PHY layer named *extended rate* PHY layer (ERP).

Different PHY transmission modes are defined with different modulation schemes, and coding rates. The performance of the modulation schemes can be measured by their robustness against path loss, interferences and fading that causes variations in the received SNR. Such variations also cause variations in the BER, since the higher the SNR, the easier it is to demod-

ulate and decode the received bits. We present a detailed evaluation of different transmission modes in Chapter 3.

Table 2.3: Characteristics of the various physical layers in the IEEE 802.11 standard.

Characteristic	*802.11a*	*802.11b*	*802.11g*
Frequency	5 GHz	2.4 GHz	2.4 GHz
Rate (Mbps)	6, 9, 12, 18, 24, 36, 48, 54	1, 2, 5.5, 11	1, 2, 5.5, 6, 9, 11, 12, 18, 22, 24, 33, 36, 48, 54
Modulation	BPSK, QPSK, 16 QAM, 64 QAM	DBPSK, DQPSK, CCK	BPSK, DBPSK, QPSK, DQPSK, CCK
	(OFDM)	(DSSS, IR, and FH)	16 QAM, 64 QAM (OFDM and DSSS)
FEC Rate	1/2, 2/3, 3/4	NA	1/2, 2/3, 3/4
Basic Rate	6 Mbps	1 or 2 Mbps	1,2, or 6 Mbps

In each physical layer, there is a basic transmission mode (usually used to send ACK, RTS, CTS and PLCP header[3]) which has the maximum coverage range among all transmission modes. This maximum range is obtained using BPSK or DBPSK modulations which have the minimum probability of bit error for a given SNR compared to other modulation schemes. The basic rates have the minimum data rate as well. The basic transmission rates for different standards are shown in Table 2.3. For instance, the basic rate is 1 Mbps (with DBPSK modulation and CRC 16 bits) for 802.11b and 6 Mbps (with BPSK and FEC rate equal to 1/2) for 802.11a.

As shown in Figure 2.5, each packet may be sent with two different rates [1]: its PLCP header is sent at the basic rate while the rest of the packet might be sent at a higher rate. The higher rate, used to transmit the physical-layer payload, is stored in the PLCP header. The receiver can verify that the PLCP header is correct (using CRC or Viterbi decoding with parity), and uses the transmission mode specified in the PLCP header to decode the MAC header and payload.

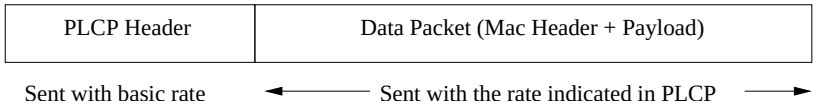


Figure 2.5: Data rates for packet transmission.

Table 2.3 also shows that the IEEE 802.11 standard defines four PHY layer transmission techniques to send data over wireless channel: *direct sequence spread spectrum* (DSSS), *fre-*

[3]Note that the AP can define a set of data transfer rates, called basic rate set, which all the stations in a BSS need to be capable of using to receive and transmit frames to/from the wireless medium. These rates can be used to send control frames but the PLCP header should always be sent with the basic rates specified in Table 2.3.

quency hopping spread spectrum (FH), *infrared* (IR), and *orthogonal frequency division multiplexing* (OFDM).

In *code division multiple access* (CDMA) systems, all users transmit in the same frequency band simultaneously. Communication systems following this concept are called *spread spectrum* (SS) systems. In this transmission technique, the frequency spectrum of a data-signal is spread using a code uncorrelated with that signal. As a result the bandwidth occupancy is much higher than required. The codes used for spreading have low cross-correlation values and are unique to every user. A receiver which has knowledge about the code of the intended transmitter, is capable of selecting the desired signal.

The SS techniques were first used in the military field, because of the difficulty to jam or detect spread spectrum signals. However, nowadays, spread spectrum systems are gaining popularity also in commercial applications. There exist different techniques to spread a signal like *direct sequence* (DS), *frequency hopping* (FH), *time hopping* (TH), and *multi carrier* CDMA (MC-CDMA). It is also possible to combine them in a single system.

DS is the best known SS Technique. In this technique, the data signal is multiplied by a *pseudo random noise* (PN) code. A PN code is a sequence of -1 and 1 (polar) or 0 and 1 (non-polar) with an specific period named chip period. The following $11-\mathrm{chip}$ Barker sequence code shall be used as the PN code sequence in the IEEE 802.11 standard: $+1, -1, +1, +1, -1, +1, +1, +1, -1, -1, -1$ [1]. A PN code has noise-like properties. This results in low cross-correlation values among the codes and the difficulty to jam or detect a data message.

In 802.11b DSSS, each information bit is combined via an XOR function with a PN sequence as shown in Figure 2.6. The result is a high speed digital stream which is then modulated. As shown in Figure 2.6, the effect of the PN code sequence is to spread the transmitted bandwidth of the resulting signal by a ratio of $11:1$ (i.e., *spread spectrum*).

Frequency hopping is another spread spectrum techniques. This technique enables the co-existence of multiple networks in the same area. The 802.11b uses FH to keep from interfering with others using the same spectrum [1]. This FH consists of 79 non-overlapping frequency channels with 1 MHz channel spacing. The transmitter frequency changes in a pseudo random fashion and each AP and each mobile STA run on different frequency hopping patterns. Typically, IEEE 802.11 does 1 MHz jumps one every tenth of second (i.e., 0.1 Sec). Since the maximum bandwidth for FH in 802.11 is 1 MHz, the transmission data can be carried out at only 1 or 2 Mbps.

The infrared PHY utilizes infrared light to transmit binary data either at 1 Mbps (basic access rate) or 2 Mbps (enhanced access rate) using a specific modulation technique for each [1]. For 1 Mbps, the infrared PHY uses a *16-pulse position modulation* (PPM). The concept of PPM is to vary the position of a pulse to represent different binary symbols. Infrared transmission at 2 Mbps utilizes the *4-PPM modulation* technique. As the IR PHY layer requires free sight between

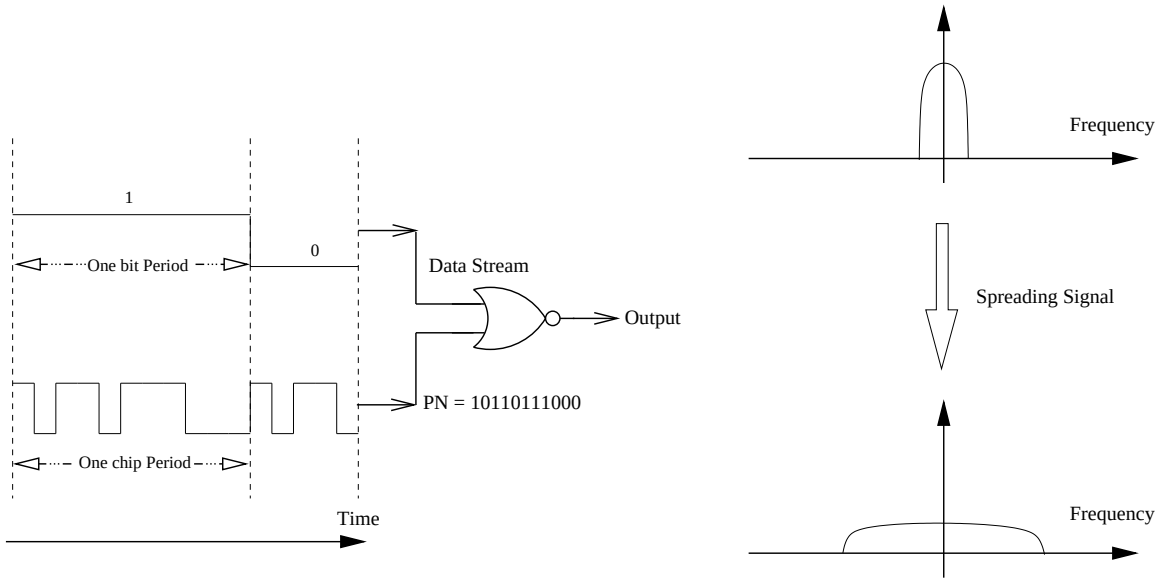


Figure 2.6: Using PN code in 802.11b to spread the signal.

sender and receiver, it is not expected to be used in large scale networks.

OFDM is a multi-carrier transmission technique, which divides the available spectrum into many carriers, each one being modulated by a low rate data stream. IEEE 802.11a/g OFDM [2, 4] is very similar to the modulation used in the *asymmetrical digital subscriber loop* (ADSL) standard called *discrete multi tone* (DMT), where the system sends several sub-carriers in parallel using the *inverse fast fourier transform* (IFFT), and receives these subcarriers using the *fast fourier transform* (FFT). The main advantage of OFDM is that it uses the available radio frequency efficiently since the sub-channels can be packed close together. The main drawback of the OFDM modulation is that due to its high frequency band, e.g., 5 GHz in IEEE 802.11a, the transmission range is very small (e.g., about 100 meters).

2.6 Physical Layer Transmission Model

Figure 2.7 shows a block diagram of the IEEE 802.11a/b PHY layer signal processing. In the following sections, we describe the functionalities of each block diagram with simple examples in 802.11a/b. All 802.11 standard senders use a scrambler before sending the data. This data whitener (scrambler and descrambler) uses a length-127 frame-synchronous scrambler followed by a 32/33 bias-suppression encoding to randomize the data and to minimize the data DC bias and maximum run lengths. The frame synchronous scrambler uses the generator polynomial $S(x)$ as follows:

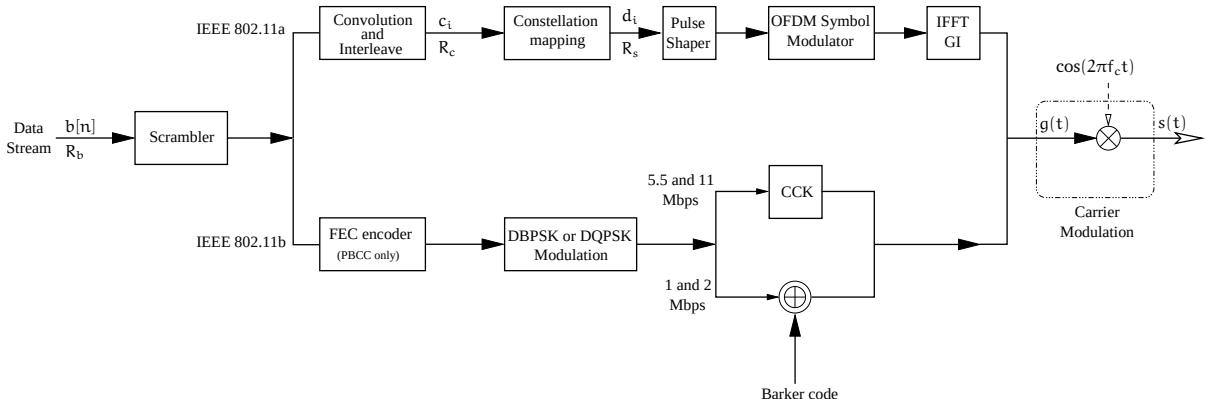


Figure 2.7: Sender block diagram for IEEE 802.11a/b.

$$S(x) = x^7 + x^4 + 1 \tag{2.1}$$

and is illustrated in Figure 2.8. The scramblers which are used for PLCP and transmit data at sender and receiver (to descramble) are identical. The data whitening starts with the first bit of the PSDU, which follows the last bit of the PLCP Header. In the two following sections, we address the functionalities of the PHY layer building blocks for 802.11a and 802.11b

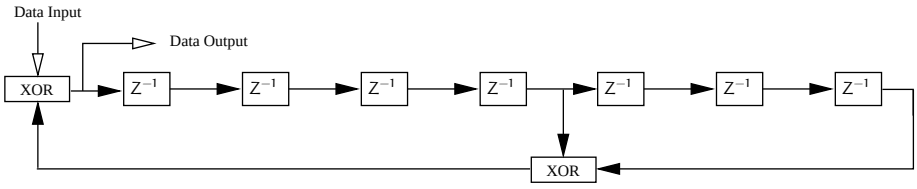


Figure 2.8: Frame synchronous scrambler/descrambler.

2.7 IEEE 802.11a Transmission Procedure

Table 2.4 shows all the available transmission modes in IEEE 802.11a WLANs along with their characteristics. Each transmission mode has a specific modulation, FEC rate, and data rate. To evaluate the performance of each transmission mode we will first describe the structure of the sender in 802.11a with a simple example.

In 802.11a transmissions, there are 5 important blocks: *convolution and interleave*, *constellation and pulse shaper*, *symbol modulator*, and *IFFT* to implement binary convolutional encoder, data modulator, and OFDM modulator. Let's consider that the sender wants to send a vector

Table 2.4: 802.11a transmission modes and their characteristics.

Mode	*Modulation*	*Data Rate* R_b *(Mbps)*	*FEC Rate (r)*	*Coded Rate* R_c *(Mbps)*	*Symbol Rate* R_s *(Msps)*	*Coded bits per symbol*	*Coded bits per OFDM symbol*	*Data bits per OFDM symbol*
1	BPSK	6	1/2	12	12	1	48	24
2	BPSK	9	3/4	12	12	1	48	36
3	QPSK	12	1/2	24	12	2	96	48
4	QPSK	18	3/4	24	12	2	96	72
5	16-QAM	24	1/2	48	12	4	192	96
6	16-QAM	36	3/4	48	12	4	192	144
7	64-QAM	48	2/3	72	12	6	288	192
8	64-QAM	54	3/4	72	12	6	288	216

of raw information bits $\mathbf{B} = [b_0, b_1, b_2, b_3, b_4, \cdots]$ which is generated at rate R_b *bits per second* (bps). Note that each b_i is equal to -1 or $+1$. First, the data should be scrambled as explained in the previous section. The following functions should be performed in an IEEE 802.11a mobile STA.

2.7.1 Binary Convolutional Encoder

The second step in data transmission in the 802.11a standard is a *binary convolutional coder* or *forward error correction* (FEC) encoder. Table 2.4 shows that there are three different FEC rates available in 802.11a. We define the code rate of this coder as $r = R_b/R_c$, where Rb is the raw information bit rate and R_c is the coded bit rate in bit per second[4]. Typically, the data string should be coded with a convolutional encoder of coding rate 1/2, 2/3, or 3/4, corresponding to the desired data rate. The convolutional encoder uses an industry-standard polynomial generator, $g_0 = (133)_8$ and $g_1 = (171)_8$, of rate R equal to 1/2, as shown in Figure 2.9. The bit denoted as *Output Data A* shall be output from the encoder before the bit denoted as *Output Data B*. Other rates (2/3 and 3/4) are derived from this encoder by employing bit-stealing and bit-insertion procedures, i.e., *puncturing*. The puncturing patterns are illustrated in [2]. Each encoded data bits shall be interleaved by a block interleaver as well. From now on, we consider that the vector $\mathbf{C} = [c_0, c_1, c_2, c_3, c_4, \cdots]$ represents the coded bits with the rate equal to R_c bits per second.

[4]Generally, in communication books, the notation is $r = k/n$ where k and n are raw bit rate and coded bit rate respectively.

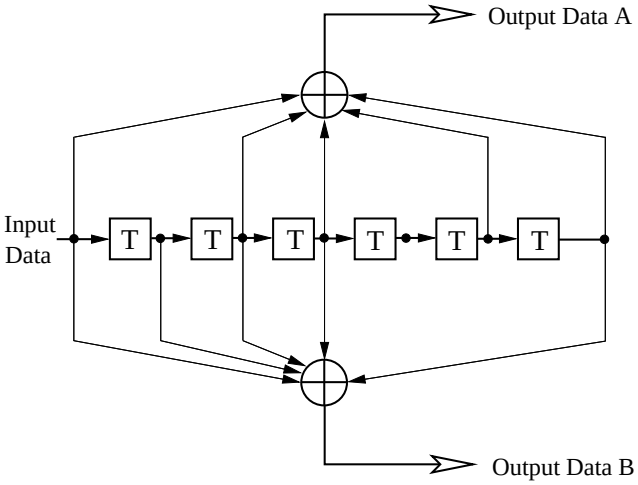


Figure 2.9: Convolutional encoder (k=7).

2.7.2 Constellation Mapping

After binary convolutional encoding, the coded bits should be mapped into constellation points. In other words, the coded bits **C** should be mapped into the vector of complex symbols $\mathbf{D} = [d_0, d_1, d_2, d_3, d_4, \cdots]$, where each symbol has the normalized Q-part and I-part as $d_m = x_m + jy_m$. Figure 2.10 shows an example of this mapping for the 16-QAM constellation. Note that all conversions have been performed according to Gray-coded mapping [2]. The output values, d_m, have also been normalized by K_{MOD},

$$d_m = (I + jQ)/K_{MOD}. \tag{2.2}$$

The normalization factor depends on the modulation, as shown in Table 2.5. The purpose of the normalization factor is to achieve the same average power for all the mappings. If we consider that the raw data bit rate is R_b, then the coded bit rate will be $R_c = R_b/r$, where r stands for the FEC rate. Let's consider p as the number of coded bits per symbol, i.e., d_m. Then the symbol transmission rate is $R_s = R_c/p = R_b/(r \times p)$ *symbol per second* (sps). Table 2.4 shows the value of R_b, R_c, and R_s for all transmission modes. Note that the number of transmitted symbols per second is identical for all transmission modes and is equal to 12 Msps.

2.7.3 OFDM Symbol Modulator

In this step, symbols are converted to waveforms for sending over the air. Symbol modulation converts each complex d_m symbol to the real waveform or time samples. Typically for a single carrier or one wideband signal modulation the *intermediate frequency* (IF) signal for each

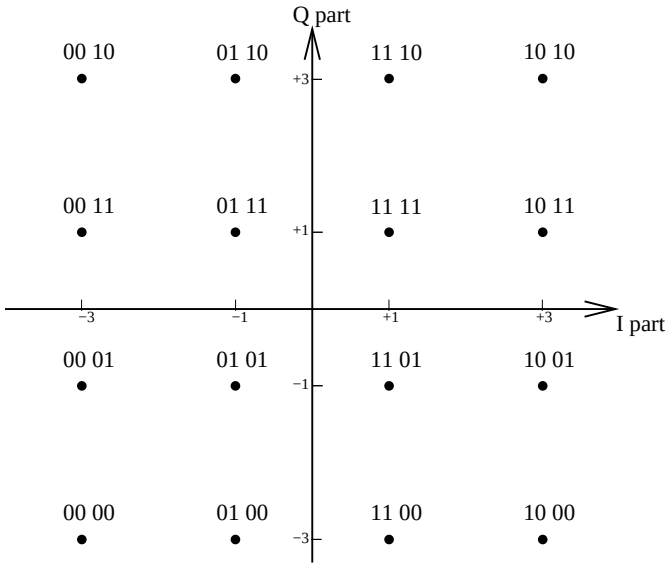


Figure 2.10: Constellation points for bit to symbol mapping for 16-QAM in 802.11a standard.

Table 2.5: Normalized factors for 802.11a modulations.

Modulation	K_{MOD}
BPSK	1
QPSK	$1/\sqrt{2}$
16-QAM	$1/\sqrt{10}$
64-QAM	$1/\sqrt{42}$

symbol is obtained by:

$$g_m(t) = x_m \cos(2\pi f_0 t) + y_m \sin(2\pi f_0 t) \qquad \text{for} \quad 0 \leq t < T \tag{2.3}$$

where f_0 (IF) is a frequency to which a carrier frequency is shifted as an intermediate step in transmission or reception and T is the symbol duration. Let's suppose that symbols are sent with rate equal to 12 Msps by a single carrier. In such a case, T is equal to 83.33 nsec (i.e., $1/R_s$). Note that in the indoor wireless environment, the maximum propagation delay between two signal paths could be several number of nanoseconds which causes *inter-symbol interference* (ISI). ISI occurs when the symbol length of the system is shorter than the multipath delay spread of the channel. In such cases, the individual symbols are smeared into each other, which typically requires an equalizer to compensate at the receiver. One well known solution for the ISI effect is the OFDM modulation. OFDM can largely avoid this ISI effect

by transforming its operating frequency band into a bank of narrower parallel channels and sending data symbols over these narrow band channels in parallel. With these narrow channels instead of one wideband channel, the data symbols in each channel can then be several times longer (for the same data rate), and so they become much longer than the delay spread of the channel. As a result, ISI is prevented, and there is little penalty for the use of the OFDM approach relative to single-carrier narrowband approaches.

Typically in 802.11a, instead of sending 12 Msps using one carrier signal, the sender uses 48 narrowband signals in parallel and each signal carries $12/48 = 0.25$ Msps. Thus, the duration of each narrowband signal is 4 μsec. On the other hand, the total RF transmission bandwidth per channel is 20 MHz based on 802.11a standard specification [2]. If we use 64 sub-carriers for OFDM modulation, each carrier spacing is $20/64 = 312.5$ kHz[5]. To maintain mutual orthogonality among subcarrier modulated signals, the duration of each symbol in every subcarrier must be $1/312.5\ msec = 3.2\ \mu sec$. To reduce the possible ISI, the 802.11a standard proposed to insert a *guard interval* (GI) of $0.8\ \mu sec$ for each symbol in every subcarrier. Finally the total symbol duration in each subcarrier will be $4\ \mu sec$. Recall that the number of subcarriers is 64 due to implementation issues and only 48 subcarriers are used for data transmission. The 802.11a standard uses 4 subcarriers for pilot and the remaining subcarriers are the guard bands. Table 2.4 shows the number of coded bits and data bits in each OFDM symbol for all 802.11a transmission modes.

2.7.4 Carrier Modulator

Before sending the signal, the carrier modulator converts the IF waveform to a *radio frequency* (RF) waveform. This is done by multiplying $g_m(t)$ to RF carrier signal, i.e., $g_m(t)\cos(2\pi f_c t)$. With bandpass filtering, the final RF signal is:

$$s_m(t) = x_m \cos(2\pi(f_0 + f_c)t) + y_m \sin(2\pi(f_0 + f_c)t) \qquad \text{for } 0 \leq t < T, \tag{2.4}$$

where the f_c (RF) is selected such that $(f_0 + f_c)$ is the designed RF center carrier frequency. The list of channel identifiers, channel center frequencies, and regulatory domains of IEEE 802.11a are presented in Appendix B.

[5]Note that the samples of the multicarrier signals can be obtained using the IFFT of the data symbols. At the receiver, the FFT can be used to obtain the data symbols as well. Since most of FFT and IFFT implementations use a power of 2 input/output, 802.11a proposed to use 64 subcarriers.

2.8 IEEE 802.11b Transmission Procedure

The first standard specification of 802.11 WLAN defined a DSSS system that provides a wireless LAN with both 1 and 2 Mbps data payload communication capability. The DSSS system uses baseband modulations of *differential binary phase shift keying* (DBPSK) and *differential quadrature phase shift keying* (DQPSK) to provide the 1 Mbps and 2 Mbps data rates, respectively [1]. In 1999, the higher-speed physical layer extension of WLAN proposed to use CCK modulation to provide higher speeds in the 2.4 GHz Band. These high rates are based on the CCK modulation scheme for 5.5 Mbps and 11 Mbps. An optional PBCC mode is also provided to potentially enhanced performance.

Table 2.6 shows all the available transmission modes in IEEE 802.11b WLANs. Again we will address the procedure of data transmission for different modulations (i.e., DBPSK, DQPSK, and CCK) with a simple example.

Table 2.6: Transmission modes in 802.11b.

Mode	*Modulation*	*Data Rate (Mbps)*	*FEC Rate*
1	DBPSK	1	NA
2	DQPSK	2	NA
3	CCK/PBCC	5.5	NA / 1/2
4	CCK/PBCC	11	NA / 1/2

2.8.1 DBPSK and DQPSK Modulations

According to the standard specification, the transmitted signal for 1 and 2 Mbps data rates is differentially encoded and modulated by BPSK and QPSK for 1 and 2 Mbps respectively. Tables 2.7 and 2.8 show the Differentially BPSK and QPSK encoder respectively [1]. The receiver can detect the signal coherently or differentially. In the latter case, it is not necessary to lock and track the carrier phase precisely. If the signal is coherently detected, we denote the modulations as *differentially encoded*, i.e., DE-BPSK and DE-QPSK. In the second case, we denote the modulations as DBPSK and DQPSK. Both cases could be implemented at the receiver.

2.8.2 CCK Modulation

The high speed extension of the IEEE 802.11 standard specifies *Complementary Code Keying* (CCK) as the modulation scheme for 5.5 and 11Mbps data rates in the 2.4 GHz band [3]. The

Table 2.7: 1 Mbps DE-BPSK encoding table.

Bit input	*Phase change (+jω)*
0	0
1	π

Table 2.8: 2 Mbps DE-QPSK encoding table.

Dibit input	*Phase change (+jω)*
00	0
01	$\pi/2$
11	π
10	$3\pi/2(-\pi/2)$

length 8 complementary codes which are used in 802.11b, can be written as a function of four phase elements ϕ_1, ϕ_2, ϕ_3, and ϕ_4 by:

$$\begin{aligned} C(\phi_1,\phi_2,\phi_3,\phi_4) &= [e^{j(\phi_1+\phi_2+\phi_3+\phi_4)}, e^{j(\phi_1+\phi_3+\phi_4)}, e^{j(\phi_1+\phi_2+\phi_4)}, -e^{j(\phi_1+\phi_4)}, \\ &\quad e^{j(\phi_1+\phi_2+\phi_3)}, e^{j(\phi_1+\phi_3)}, -e^{j(\phi_1+\phi_2)}, e^{j(\phi_1)}] \end{aligned} \tag{2.5}$$

For example, to generate the $2^8 = 256$ codewords needed to transmit data at 11 Mbps from this expression, the four phase parameters are each allowed to take one of the four values 0, $\pi/2$, π, $3\pi/2$. This is similar to allowing each phase to be drawn from a QPSK constellation. In order to understand the mathematical representation of CCK Modulation, it is useful to show how a code is generated in CCK for 11 Mbps data rate. A signal in CCK Modulation starts out as an eight-bit binary word $\mathbf{D} = d_7d_6d_5d_4d_3d_2d_1d_0$. The 8 bits are used to encode the phase parameters (i.e., the ϕ_1 to ϕ_4). The encoding is based on the differential QPSK modulation. The first dibit (d_0, d_1) encodes ϕ_1 based on the DQPSK specified in Table 2.8. Then, the data dibits (d_2, d_3), (d_4, d_5), and (d_6, d_7) encode ϕ_2, ϕ_3, and ϕ_4 respectively, based on QPSK as specified in Table 2.9. Note that this table is binary (not Grey) coded. For example, for a data stream given as 01100011, we get from Table 2.8, 2.9, and 2.10: $d_1d_0 = 11$, $\phi_1 = \pi$, $d_3d_2 = 00$, $\phi_2 = 0$, $d_5d_4 = 10$, $\phi_3 = 3\pi/2$, $d_7d_6 = 01$, $\phi_4 = \pi/2$. Finally, using the Equation (2.5), we can find out the codes which should be sent from all possible codes.

Table 2.9: QPSK encoding table for CCK 11 Mbps.

Dibit input	*Phase change (+jω)*
00	0
01	$\pi/2$
10	π
11	$3\pi/2(-\pi/2)$

Table 2.10: Phase parameter encoding scheme.

Dibits	*Phase Parameter*
(d_1, d_0)	ϕ_1
(d_3, d_2)	ϕ_2
(d_5, d_4)	ϕ_3
(d_7, d_6)	ϕ_4

2.9 IEEE 802.11 Frame Format

Practically, an IP based packet encapsulation in IEEE 802.11 is performed as described in Figure 2.11. This figure shows all headers from the application layer to the PHY layer. Note that in the UDP and MAC levels, checksum fields are done over the whole packet. There is also a checksum for the IP header. In the IEEE 802.11 MAC layer, each MPDU packet consists of the following basic components: a MAC header, optional IP/UDP/RTP/NAL headers, a variable length information frame body, and a *frame check sequence* (FCS). All the fields except the frame body, which is 28 octets in total, contribute to the MAC overhead for a data/fragment frame. The format of the MAC header with the exception of FCS is shown in Figure 2.12[6]. Figure 2.12 also shows the modified frame format for IEEE 802.11e which contains a new field named *QoS Control field*. The QoS Control field is a 16-bit field that identifies the TC to which the frame belongs and various other QoS-related information about the frame that varies by frame type and subtype. More information about the functionalities of these control fields can be found in [13]. The frame format of RTS, CTS, and ACK packets are shown in Figure 2.13 as well.

Figure 2.14 illustrates the PLCP preamble and header format in IEEE 802.11a. The PLCP preamble field, with the duration of $t\mathrm{PLCPPreamble}$, is composed of 10 repetitions of a short training sequence ($0.8\mu s$) and 2 repetitions of a long training sequence ($4\mu s$). The PLCP pream-

[6]There is another address field named *Address4* in MAC header which is assigned for *wireless distribution system* (WDS) frames being distributed from one AP to another AP. This field is omitted when it is not applicable (N/A).

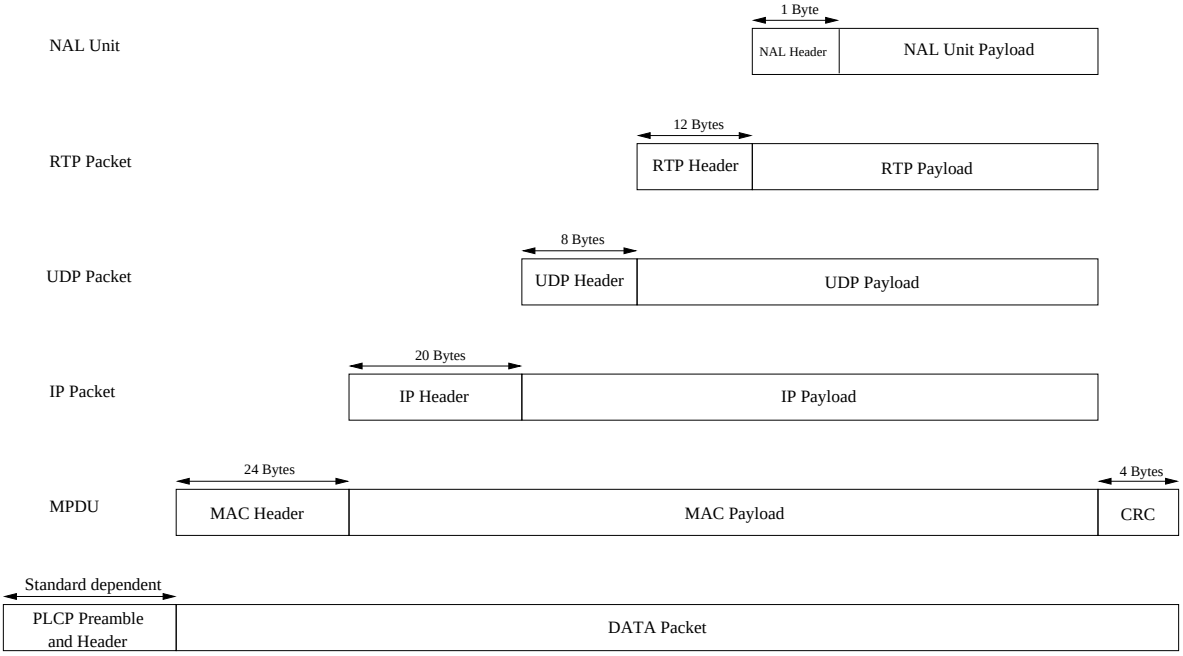


Figure 2.11: Packet encapsulation in 802.11.

ble field is used for synchronization. The PLCP header except the SERVICE field, with the duration of $tPLCPHeader$, constitutes a separate OFDM symbol, which is transmitted with BPSK modulation and the rate-1/2 convolutional coding.

There are two different PLCP frame formats in IEEE 802.11b: Long and Short PLCP as shown in Figure 2.15. The long PLCP including the High Rate PLCP preamble and the High Rate PLCP header. The PLCP preamble contains the two following fields: *synchronization* (SYNC) and *start frame delimiter* (SFD). The PLCP header contains the four following fields: SIGNAL, SERVICE, LENGTH, and CCITT CRC-16 (CRC). Each of these fields is described in detail in the standard. The PLCP header and preamble must be sent using the basic mode corresponding

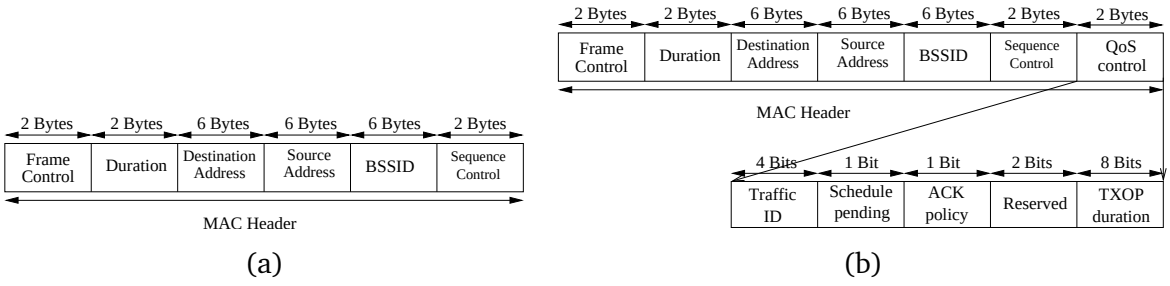
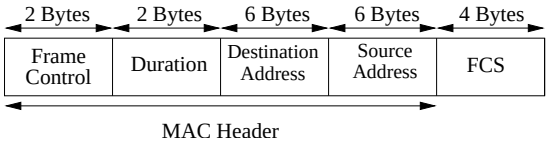
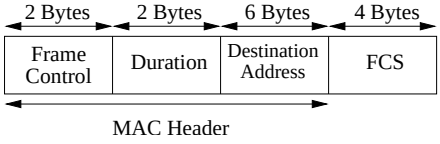


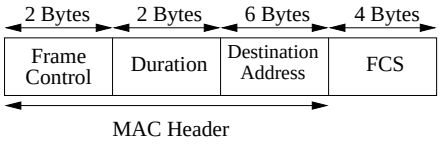
Figure 2.12: MAC header format in (a) IEEE 802.11 and (b) IEEE 802.11e standards.



(a) RTS frame format



(b) CTS frame format



(c) ACK frame format

Figure 2.13: Control frames in IEEE 802.11 standard.

to 1 Mbps and DBPSK modulation in 802.11b. It is worth mentioning that the short frame format is not compatible with the PPDUs used in the classic DSSS PHY layer. The short PLCP header uses the 2 Mbps with DQPSK modulation and a transmitter using the short PLCP can only interoperate with the receivers which are capable of receiving this short PLCP format. The short PLCP preamble and header may be used to minimize overhead and thus maximize the network data throughput.

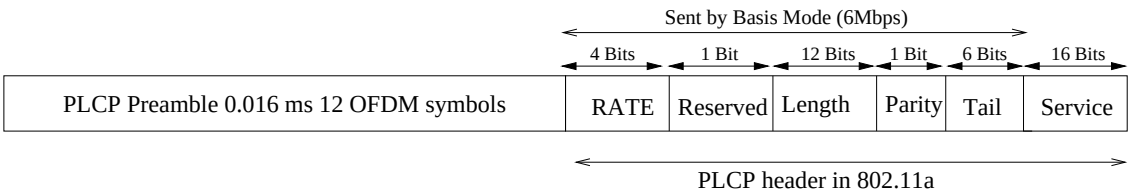
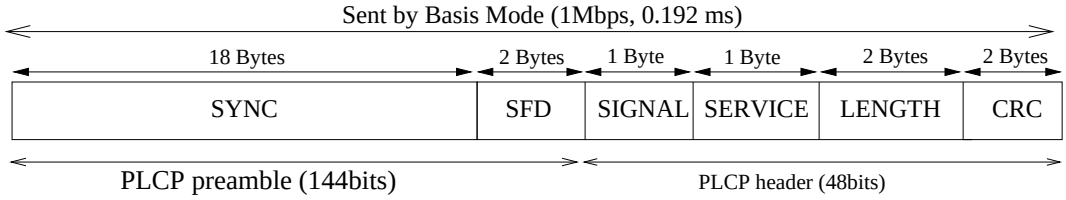
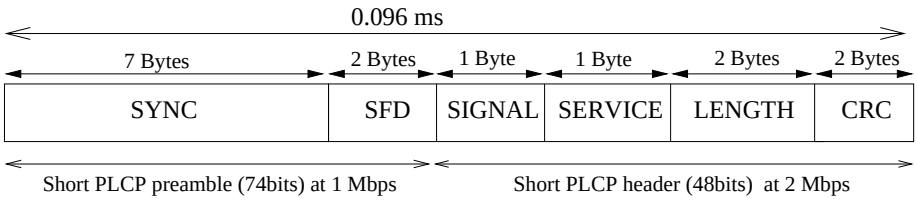


Figure 2.14: PLCP format in 802.11a.



(a) Long PLCP



(b) Short PLCP

Figure 2.15: (a) Long and (b) short PLCP header format in 802.11b.

2.10 Conclusions

In this chapter, we reviewed the basic components of the IEEE 802.11 standard specification. We presented the reference model of IEEE 802.11. We explained the CSMA/CA protocol which is employed for medium access control in the IEEE 802.11 standard. We also reviewed the optional RTS/CTS protocol which is used to solve the hidden node problem in the IEEE 802.11 WLANs. Important factors such as the inter frame space time and the contention windows are derived for all IEEE 802.11 standards. The most important features of the 802.11e standard, i.e., HCCA and EDCA, have been summarized. Then we explained the PHY layer transmission techniques, i.e, DSSS, IR, FH, and OFDM, and we described the transmission model for 802.11a/b devices. The functionalities of all PHY layer building blocks at 802.11a/b were discussed in this chapter with simple examples. We explained how a data stream should be treated before transmitting over the air.

The binary convolutional code techniques and the data and carrier modulation techniques used in the 802.11 standards are explained in this section. In order to evaluate the performance of the wireless devices, we need to know the above knowledge from MAC and PHY layer. It is necessary to know the functionalities of these building blocks to design new adaptive and cross layer protocols in WLANs. In summary, we showed in this chapter, the variety of factors and parameters that should be considered to propose and evaluate adaptive and cross layer mechanisms in IEEE 802.11 WLANs.

3

Performance Analysis of IEEE 802.11 Data Transmission

3.1 Summary

The IEEE 802.11a/b/g WLAN standards define several transmission modes. While a lot of performance studies have been done for the MAC layer in 802.11 standards, very few analysis of 802.11a/b/g transmission modes considering the physical layer parameters as well as wireless channel characteristics are available. In this chapter, we present a complete performance evaluation of all available data transmission modes in 802.11a/b. Note that the performance evaluation of IEEE 802.11g is straightforward using the evaluation of 802.11a/b. We first discuss the structure of 802.11a/b receivers. Then three more frequently used models for wireless channels (i.e., AWGN, Rayleigh, and Rice) are presented in this chapter. The performance of transmission modes are evaluated by calculating the probability of bit error and symbol error versus the signal to noise ratio at the receiver. We consider the data modulation, Viterbi decoding, and data rate to calculate the probability of bit/symbol error rates. We perform this evaluation with AWGN and Rayleigh fading model for the wireless channel. We also consider the case where the receiver uses several replicas of the received signal to decode the transmitted signal, i.e., a multipath receiver. The influence of wireless channel models and the number of antennas (i.e., receiver structure) on the performance of transmission modes are shown by the simulation results. Based on simulation results we also show that some transmission modes are not efficient in 802.11a. Our performance evaluation confirms that the physical layer FEC increases significantly the coverage range of the 802.11a WLAN devices.

3.2 Introduction

Unlike wired networks that can provide large bandwidth, the bandwidth of WLANs is rather limited because they rely on an inexpensive, but error prone, physical medium (i.e., air). Hence, it is important to evaluate the performance of wireless devices by considering the transmission characteristics, wireless channel parameters, and device structure. In this chapter, we investigate all the parameters cited above and present a complete evaluation of all transmission modes for 802.11a/b devices. It is worth mentioning that, in this chapter, we do not focus on the 802.11g standard since it uses the same transmission modes as 802.11a and 802.11b (see Section 2.2).

Typically, the performance of data transmission over wireless channels is well captured by observing their bit error rate which is a function of the signal to noise ratio at the receiver. Several models have been already proposed and investigated to calculate the signal to noise ratio in wireless channels. These models are a function of the distance between the sender and the receiver, the path loss exponent, and the channel gain. The channel gain which is a time-variation parameter is modelled by probability distribution functions. There are several candidates for these probability distributions [20]. In this chapter, we describe the three most important and commonly used distributions for this probability distribution, i.e., AWGN, Rayleigh, and Rician models. We discuss the multilink receiver structure in this chapter as well. In these kind of receivers, the signal is detected and decoded by employing several replicas of the received signal.

As we discussed in Chapter 2, each PHY layer transmission mode consists of a specific set of modulation, binary convolutional coding, and data rate. In addition, according to the IEEE 802.11 standard specification, each device should use a wireless transmission technique (over-the-air modulation) among OFDM, DSSS, FH, and IR. In our evaluation we consider all above parameters. For binary convolutional coding we suppose that the receiver uses a Viterbi decoder which is already recommended by the standard specification. This chapter is organized as follow.

We first present the structure of IEEE 802.11a/b receivers in Section 3.3. An introduction about wireless channel modelling and their important parameters is presented in Section 3.4. In the same section, we discuss the additive white Gaussian and fading channels. Then we present a complete performance evaluation of 802.11a and 802.11b transmission modes in Section 3.5 and Section 3.6 respectively. This analysis is done over AWGN channel and Rayleigh fading channels, considering multipath receiver structure. We also evaluate the performance of Viterbi decoders in 802.11a. Finally, we conclude the chapter in Section 3.7.

3.3 IEEE 802.11 Receiver Structure

To evaluate the performance of IEEE 802.11 WLAN mobile STAs, it is important to know the structure of the WLAN STAs for the IEEE 802.11a/b standard. Figure 3.1 shows the receiver structure of an 802.11a mobile STA. As explained in Section 2.9, the 802.11a receiver STA uses the first ten short symbols for frame detection. It uses two OFDM frames after the ten short symbols for frequency offset estimation. These two frames contain two identical FFT symbols back-to-back. The channel estimation and compensation module also uses the same two OFDM symbols for the frequency offset estimation. Once a frame start is detected, frequency offset is estimated, and signal samples are compensated. The FFT is employed in the receiver to obtain the data symbols. Using FFT, there is no need to employ several oscillators, filters, and other physical layer detection mechanism for each subcarrier. It is worth mentioning that the popularity of OFDM is due to the use of IFFT/FFT which have efficient implementations. Finally Viterbi decoding should be performed on the quantized channel symbols to recover the original binary data.

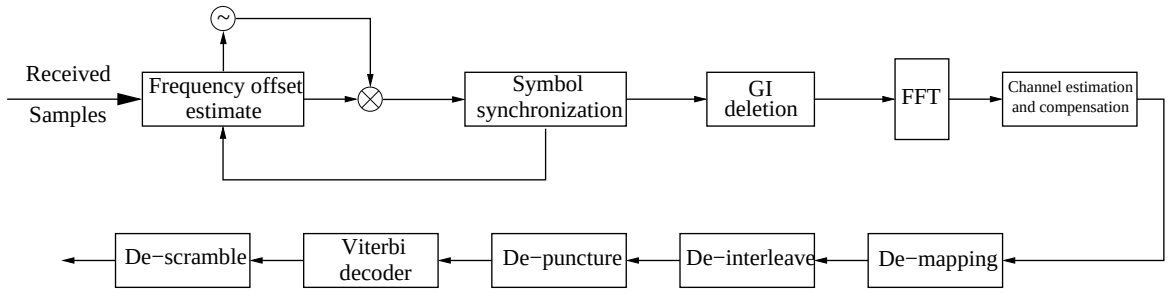


Figure 3.1: Block diagram of the 802.11a receiver PHY layer.

Figure 3.2 shows the structure of receiving 802.11b mobile STAs. Note that the structure of receiver for 1 and 2 Mbps has the despreading building block which uses the same Barker codes described in Section 2.5. This structure depends on coherent or differentially detection of signal as well.

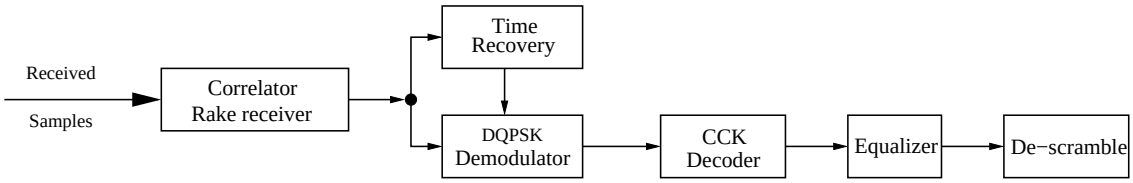


Figure 3.2: Block diagram of the 802.11b receiver PHY layer.

3.4 Wireless channel modelling

In practice, transmitted radio waves are reflected by surrounding objects. When striking the surface of these objects, radio waves are partially reflected, partially absorbed and partially transmitted through the objects. The signal at the receiver is a superposition of the line-of-sight and several non-line-of-sight signals. These signals combined with noise are decoded at the receiver. The coherent addition (cancellation) of these signals leads to a high (low or zero) received signal power at the receiver. For the purpose of designing and optimizing receiver structures for digital communication systems, it is mandatory to construct mathematical models that represent the typical characteristics of these channels. In order to model such wireless channels, we can suppose that for each point in the 3-dimensional space, the wireless channel is a linear time-varying filter with following impulse response [20]:

$$h(t,\tau) = \sum_{k=0}^{N(\tau)-1} a_k(t)\delta[\tau - \tau_k(t)]e^{j\theta_k(t)} \tag{3.1}$$

where t and τ are the observation time and application time of the impulse, respectively, $N(\tau)$ is the number of multipath components, $a_k(t)$, $\tau_k(t)$, $\theta_k(t)$ are the random time-varying amplitude[1], arrival-time, and phase sequence, respectively, and δ is the delta function. The output of the wireless channel for a given transmitted signal $s(t)$ can be calculated by:

$$y(t) = \int_{-\infty}^{\infty} s(t)h(t-\tau)d\tau + n(t) \tag{3.2}$$

where $n(t)$ is the background noise. The wireless channel can be completely characterized by the above parameters as shown in Figure 3.3. In this model, the power of the received signal is a function of the distance between the sender and the receiver, the path loss exponent (which is usually 3 for indoor environment), and the average channel gain considering the reflected signals.

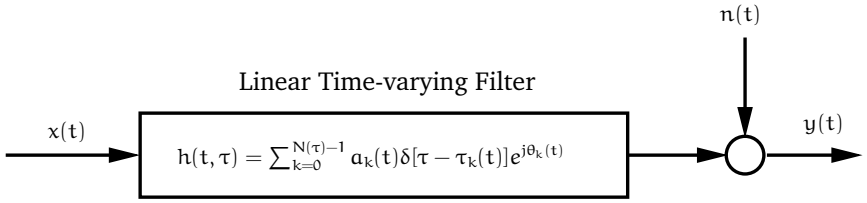


Figure 3.3: Mathematic model for wireless channel.

[1]Since path loss attenuation depends on the distance between sender and receiver.

3.4.1 Additive White Gaussian Noise Channel: AWGN

An AWGN channel adds white Gaussian noise to the signal that passes through it. It is the basic model for a digital communication channel and therefore used as a standard channel model. In this model, the transmitted signal gets disturbed by *only* a simple additive white gaussian noise process. Thus the received signal can be obtained by:

$$y(t) = s(t) + n(t), \tag{3.3}$$

where $n(t)$ denotes a sample function of the additive white Gaussian noise process with power spectral density of $\Phi_{nn}(f) = \frac{1}{2}N_o$ W/Hz [21]. The white noise is a noise which has a frequency spectrum that is continuous and uniform over a specified frequency band.

Comparing to Equation (3.1), in AWGN channel model we suppose that there is no random scattering and diffraction transmitted signals at the receiver. Unfortunately, most wireless channel links cannot be modelled as AWGN channels. In the next section we address the wireless fading channel which is more realistic.

3.4.2 Fading Channel

Multipath fading is due to the constructive and destructive combination of randomly delayed, reflected, scattered, and diffracted signal components. This type of fading is relatively fast and is therefore responsible for the short-term signal variations [22]. Depending on the nature of the radio propagation environment, there are different models describing the statistical of the multipath fading channel, like *Rayleigh*, *Nakagami-q*, *Nakagami-n* (Rice), and *Nakagami-m* model. The Rayleigh distribution is frequently used to model multipath fading with no direct *line of sight* (LOS) path. In this case, the channel fading amplitude α is distributed according to

$$p_\alpha(\alpha) = \frac{2\alpha}{\Omega}\exp(-\frac{\alpha^2}{\Omega}), \tag{3.4}$$

where α is a RV with mean-square value $\Omega = \overline{\alpha^2}$. In this model, the instantaneous SNR per symbol of the channel γ is distributed according to an exponential distribution given by

$$p_\gamma(\gamma) = \frac{1}{\overline{\gamma}}\exp(-\frac{\gamma}{\overline{\gamma}}) \qquad \gamma \geqslant 0. \tag{3.5}$$

Similarly, the Rice distribution which is often used to model propagation paths consisting of one strong direct LOS component can be characterized by:

$$p_\alpha(\alpha) = \frac{2(1+n^2)e^{-n^2}\alpha}{\Omega}\exp\left(-\frac{(1+n^2)\alpha^2}{\Omega}\right) I_0\left(2n\alpha\sqrt{\frac{1+n^2}{\Omega}}\right), \qquad \alpha \geqslant 0 \tag{3.6}$$

where n is the Nakagami-n fading parameter, which ranges from 0 to ∞. $I_0(.)$ is the zero-order modified Basel function of the first kind. The SNR of the channel, γ, is distributed (i.e., p_γ) according to a noncentral chi-square distribution given by,

$$p_\gamma(\gamma) = \frac{(1+n^2)e^{-n^2}}{\overline{\gamma}} \exp\left(-\frac{(1+n^2)\gamma}{\overline{\gamma}}\right) I_0\left(2n\sqrt{\frac{(1+n^2)\gamma}{\overline{\gamma}}}\right) \qquad \gamma \geqslant 0. \tag{3.7}$$

Similar distributions can be found in [22] for other fading models. In this thesis, we will use the Rayleigh fading model which is the most frequently used model for multipath wireless channels.

Generally, the performance of communication systems over such wireless channels is captured and analyzed using the probability of *bit error rate* (BER) as a function of *signal to noise and interference ratio* (SNR). The BER decreases for high SNR at the receiver. However, the actual performance of the wireless system depends on several implementation issues and wireless channel characteristics. In the next sections, we present a complete evaluation of the different transmission modes in 802.11a/b.

3.5 IEEE 802.11a Performance Evaluation

As detailed in Section 2.7, there are eight transmission modes in 802.11a which use different sets of FEC and modulation codes. In this section, we evaluate the performance of these transmission modes over AWGN, and Rayleigh fading channels and multipath receiver.

3.5.1 Performance Evaluation in AWGN Channel

In 802.11a devices, BPSK is used for the first two transmission modes and the rest use different versions of QAM modulation. It can be shown that in AWGN the probability of bit error (or symbol error rate as there is one bit in each symbol) is obtained by [21]:

$$P_b = Q\left(\sqrt{2 \cdot \frac{\mathcal{E}_s}{N_o}}\right) \tag{3.8}$$

where $\frac{\mathcal{E}_s}{N_0}$ is the average signal to noise ratio per symbol (or bit) and Q-function is defined as: $Q(x) = \frac{1}{\sqrt{2\pi}} \cdot \int_x^\infty e^{-t^2/2} dt$. Note that if $\frac{\mathcal{E}_b}{N_o}$ (γ_b) is the average signal to noise ratio per bit, γ_b can be derived from the SNR using the following equation:

$$SNR = \frac{Signal\ Power}{Noise\ Power} = \frac{R_b \cdot \mathcal{E}_b}{BW \cdot N_o} = \frac{R_s \cdot \mathcal{E}_s}{BW \cdot N_o}, \tag{3.9}$$

where R_b and R_s is the maximum bit rate and symbol rate of the modulation scheme respectively, $\mathcal{E}_b$ and $\mathcal{E}_s$ is the energy of bit and symbol respectively, and BW is the bandwidth of

the signal. Regarding the noise in AWGN channels, we assume that the noise over the wireless channel is white Gaussian with spectral density equal to $N_o/2$ (Other models for N_o can be considered as well.). Typically, we suppose that N_o is the power of the thermal noise which can obtained by,

$$N_o = N_f \cdot N_t = N_f \cdot kTW, \tag{3.10}$$

where N_f denotes the circuit noise value, k the Boltzmann constant, T the temperature in Kelvin and W is the frequency bandwidth.

For the $M-$ary QAM modulation, the probability of symbol error can be calculated by [21],

$$P_s = P_M = 1 - (1 - P_{\sqrt{M}})^2, \tag{3.11}$$

where $P_{\sqrt{M}}$ is the symbol error probability for the $\sqrt{M}-$ary PAM and can be obtained by,

$$\begin{aligned} P_{\sqrt{M}} &= 2\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3}{M-1} \cdot \frac{\mathcal{E}_s}{N_o}}\right) \\ &= 2\left(1 - \frac{1}{\sqrt{M}}\right) Q\left(\sqrt{\frac{3k}{M-1} \cdot \frac{\mathcal{E}_b}{N_o}}\right). \end{aligned} \tag{3.12}$$

Note that for the $M-$ary QAM modulation there are $k = \log_2 M$ bit per symbol, thus $\frac{\mathcal{E}_s}{N_o}$ is equal to $k \cdot \frac{\mathcal{E}_b}{N_o}$. To facilitate our calculation lets define $a = 2(1 - \frac{1}{\sqrt{M}})$ and $b = \sqrt{\frac{3k}{M-1}}$, thus the probability of symbol error rate can be calculated as:

$$\begin{aligned} P_s &= 1 - \left(1 - a\, Q\left(b\sqrt{\frac{\mathcal{E}_b}{N_o}}\right)\right)^2 \\ &= 2a\, Q\Big(b\sqrt{\frac{\mathcal{E}_b}{N_o}}\Big) - a^2\, Q^2\Big(b\sqrt{\frac{\mathcal{E}_b}{N_o}}\Big). \end{aligned} \tag{3.13}$$

With a Gray coding, the bit error probabilities for the QPSK (4-QAM) and BPSK are identical and is equal to probability of symbol error for BPSK. For other QAM modulations (i.e., 16-QAM and 64-QAM) used in 802.11a, we use a recursive algorithm to calculate the probability of bit error presented in [23]. Table 3.1 shows the probability of bit error and symbol error for all modulations available in 802.11a with an AWGN channel model.

3.5.2 Performance Evaluation in Rayleigh Fading Channel

As explained in Section 3.4.2, in the case of communication over a slow-fading channel, the instantaneous signal to noise ratio per bit, γ_b, is a time-invariant random variable with a PDF, $p_{\gamma_b}(\gamma_b)$, defined by the type of fading. In this section, we focus on the Rayleigh fading model.

Table 3.1: Probability of symbol error and bit error in AWGN for the modulations available in 802.11a.

Modulation	*Symbol error*	*Bit error*
BPSK	$Q\left(\sqrt{2\cdot\frac{\mathcal{E}_s}{N_o}}\right)$	$Q\left(\sqrt{2\cdot\frac{\mathcal{E}_b}{N_o}}\right)$
4-QAM	$2Q\left(\sqrt{\frac{\mathcal{E}_s}{N_o}}\right)-Q^2\left(\sqrt{\frac{\mathcal{E}_s}{N_o}}\right)$	$Q\left(\sqrt{2\cdot\frac{\mathcal{E}_b}{N_o}}\right)$
16-QAM	$3Q\left(\sqrt{\frac{\mathcal{E}_s}{5N_o}}\right)-\frac{9}{4}Q^2\left(\sqrt{\frac{\mathcal{E}_s}{5N_o}}\right)$	$\frac{3}{4}Q\left(\sqrt{\frac{4\mathcal{E}_b}{5N_o}}\right)+\frac{1}{2}Q\left(3\sqrt{\frac{4\mathcal{E}_b}{5N_o}}\right)$
64-QAM	$\frac{7}{2}Q\left(\sqrt{\frac{\mathcal{E}_s}{21N_o}}\right)-\frac{49}{16}Q^2\left(\sqrt{\frac{\mathcal{E}_s}{21N_o}}\right)$	$\frac{7}{12}Q\left(\sqrt{\frac{2\mathcal{E}_b}{7N_o}}\right)+\frac{1}{2}Q\left(3\sqrt{\frac{2\mathcal{E}_b}{7N_o}}\right)$

To evaluate the performance of the above modulations, one must evaluate an integral whose integrand consists of the following product of the Q-function and fading PDF [22]:

$$I=\int_0^\infty Q(C\sqrt{\gamma_b})p_{\gamma_b}(\gamma_b)\,d\gamma_b, \tag{3.14}$$

where C is a constant that depends on the modulation and detection combination (see Table 3.1). We use the desired form of the Gaussian Q-function integral i.e.,

$$Q(x)=\frac{1}{\pi}\int_0^{\pi/2} exp(-\frac{x^2}{2\sin^2\theta})d\theta, \tag{3.15}$$

The result is,

$$I=\int_0^\infty \frac{1}{\pi}\int_0^{\pi/2} exp(-\frac{C^2\gamma_b}{2\sin^2\theta})\,d\theta\,p_{\gamma_b}(\gamma_b)\,d\gamma_b=\frac{1}{\pi}\int_0^{\pi/2}\Big[\int_0^\infty exp(-\frac{C^2\gamma_b}{2\sin^2\theta})\,p_{\gamma_b}(\gamma_b)\,d\gamma_b\Big]\,d\theta \tag{3.16}$$

Since the inner integral is in the form of a laplace transform with respect to the variable γ_b Equation (3.16) can be rewritten as:

$$I=\frac{1}{\pi}\int_0^{\pi/2} M_{\gamma_b}\Big(-\frac{C^2}{2\sin^2\theta}\Big)\,d\theta, \tag{3.17}$$

where M_{γ_b} is the *moment generating function* (MGF) of γ_b,

$$M_{\gamma_b}(s)=\int_0^\infty e^{s\gamma_b}\,p_{\gamma_b}(\gamma_b)\,d\gamma_b \tag{3.18}$$

In the Rayleigh channel, the PDF of the instantaneous signal to noise ratio per bit is given by Equation (3.5):

$$p_{\gamma_b}(\gamma_b) = \frac{1}{\bar{\gamma_b}} \, exp(-\frac{\gamma_b}{\bar{\gamma_b}}), \qquad \gamma_b \geq 0 \tag{3.19}$$

where $\bar{\gamma_b}$ is the average signal to noise ratio per bit. The Laplace transform of the Rayleigh PDF can be presented in the closed form by:

$$M_{\gamma_b}(-s) = \frac{1}{1 + s\bar{\gamma_b}}, \qquad s > 0 \tag{3.20}$$

Finally using the Equation (3.8), the probability of the bit (symbol) error in Rayleigh fading channel can be calculated as:

$$P^{Ray}_{s,BPSK} = \frac{1}{\pi} \int_0^{\pi/2} \frac{d\theta}{1 + \frac{\bar{\gamma_b}}{\sin^2 \theta}} = \left(1 - \sqrt{\frac{\bar{\gamma_b}}{1 + \bar{\gamma_b}}}\right) = P^{Ray}_{b,BPSK}. \tag{3.21}$$

Similar calculations can be done for other modulations. For example, using Equation (3.13) the probability of symbol error for QAM modulations will be:

$$P^{Ray}_{s,M-QAM} = a \left(1 - \sqrt{\frac{b^2\bar{\gamma_b}/2}{1 + b^2\bar{\gamma_b}/2}}\right) - \frac{a^2}{4} \left(1 - \sqrt{\frac{b^2\bar{\gamma_b}/2}{1 + b^2\bar{\gamma_b}/2}} \times \frac{4}{\pi} \tan^{-1} \left(\sqrt{\frac{1 + b^2\bar{\gamma_b}/2}{b^2\bar{\gamma_b}/2}}\right)\right) \tag{3.22}$$

Table 3.2 shows the probability of symbol and bit error in Rayleigh fading channel for all available modulations in 802.11a.

Table 3.2: Probability of symbol error and bit error in Rayleigh fading channel for 802.11a modulations.

Modulation	*Symbol error*	*Bit error*
BPSK	$\frac{1}{2}\left(1 - \sqrt{\frac{\bar{\gamma_b}}{1+\bar{\gamma_b}}}\right)$	$\frac{1}{2}\left(1 - \sqrt{\frac{\bar{\gamma_b}}{1+\bar{\gamma_b}}}\right)$
4-QAM	$\left(1 - \sqrt{\frac{\bar{\gamma_b}}{1+\bar{\gamma_b}}}\right) - \frac{1}{4}\left(1 - \sqrt{\frac{\bar{\gamma_b}}{1+\bar{\gamma_b}}} \times \frac{4}{\pi} \tan^{-1}(\sqrt{\frac{1+\bar{\gamma_b}}{\bar{\gamma_b}}})\right)$	$\frac{1}{2}\left(1 - \sqrt{\frac{\bar{\gamma_b}}{1+\bar{\gamma_b}}}\right)$
16-QAM	$\frac{3}{2}\left(1 - \sqrt{\frac{2\bar{\gamma_b}/5}{1+2\bar{\gamma_b}/5}}\right) - \frac{9}{16}\left(1 - \sqrt{\frac{2\bar{\gamma_b}/5}{1+2\bar{\gamma_b}/5}} \times \frac{4}{\pi} \tan^{-1}(\sqrt{\frac{1+2\bar{\gamma_b}/5}{2\bar{\gamma_b}/5}})\right)$	$\frac{5}{8} - \frac{3}{8}\sqrt{\frac{2\bar{\gamma_b}}{5+2\bar{\gamma_b}}} - \frac{1}{4}\sqrt{\frac{18\bar{\gamma_b}}{5+18\bar{\gamma_b}}}$
64-QAM	$\frac{7}{4}\left(1 - \sqrt{\frac{\bar{\gamma_b}/7}{1+\bar{\gamma_b}/7}}\right) - \frac{49}{64}\left(1 - \sqrt{\frac{\bar{\gamma_b}/7}{1+\bar{\gamma_b}/7}} \times \frac{4}{\pi} \tan^{-1}(\sqrt{\frac{1+\bar{\gamma_b}/7}{\bar{\gamma_b}/7}})\right)$	$\frac{13}{24} - \frac{7}{24}\sqrt{\frac{\bar{\gamma_b}}{7+\bar{\gamma_b}}} - \frac{1}{4}\sqrt{\frac{9\bar{\gamma_b}}{7+9\bar{\gamma_b}}}$

3.5.3 Performance of Multichannel Receivers

As explained in Section 3.4, signal multipath occurs when the transmitted signal arrives at the receiver via multiple propagation paths and with different delays. If several replicas of the same signal over independent channels are supplied to the receiver, the probability that all the signal components fade at the same time is decreased [21]. In this section, we explain the structure of a multichannel receiver and evaluate its performance on Rayleigh fading channels.

Figure 3.4 shows the mathematic model of a multipath channel, where the transmitted signal is received over L independent slowly varying flat fading channels, as shown in [22].

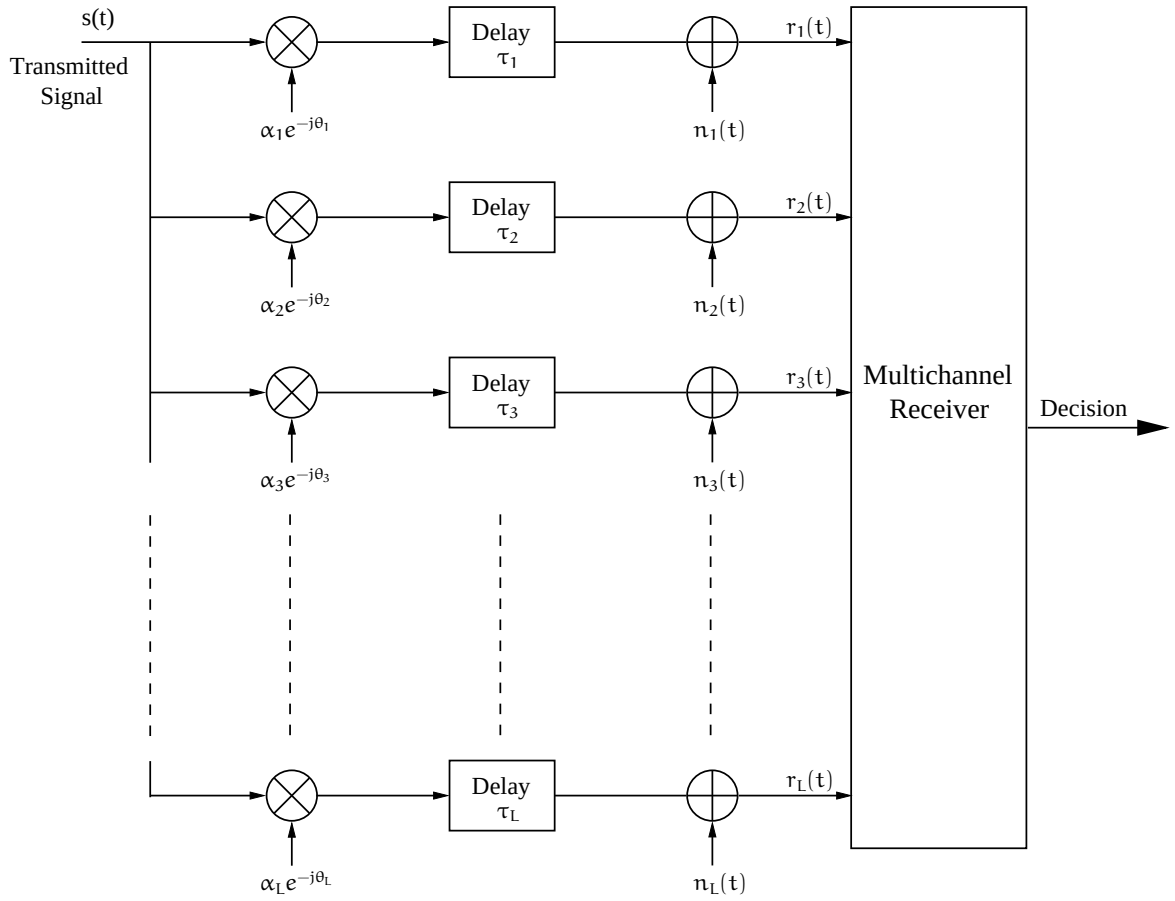


Figure 3.4: Multipath channel model.

We suppose that $\{\alpha_\ell\}_{\ell=1}^{L}$, $\{\theta_\ell\}_{\ell=1}^{L}$, and $\{\tau_\ell\}_{\ell=1}^{L}$ are the random channel amplitudes, phases, and delay respectively, where ℓ is the channel index. We also assume that the sets $\{\alpha_\ell\}_{\ell=1}^{L}$, $\{\theta_\ell\}_{\ell=1}^{L}$, and $\{\tau_\ell\}_{\ell=1}^{L}$ are mutually independent which means that the fading process is Rayleigh distributed with L resolvable iid paths. With this assumption, the fading-averaged $\bar{\gamma_b}$ can be computed with:

$$\overline{\gamma}_b = \mathbb{E}\Big[\sum_{\ell=1}^{L} \alpha_\ell^2 \gamma_\ell\Big] \tag{3.23}$$

where $\mathbb{E}[.]$ is the expectation operator, α_ℓ's are iid Rayleigh-distributed path gains and γ_ℓ is the received bit energy to noise density ratio for the ℓ^{th} path. It can be shown that with maximum-ratio diversity combining, the fading-averaged BER for BSPK and QPSK, can be obtained by [22, 21]:

$$P_b = \Big[\frac{1-\mu_c}{2}\Big]^L \sum_{l=0}^{L-1} \binom{L-1+l}{l} \Big[\frac{1+\mu_c}{2}\Big]^l, \tag{3.24}$$

where $\mu_c = \sqrt{\frac{\overline{\gamma}_b}{1+\overline{\gamma}_b}}$. Similar expressions can be derived for other modulations. For instance, considering the Equation (3.13), the performance of M-QAM modulation (i.e., symbol error) over L *i.i.d.* Rayleigh fading channel can be calculated by:

$$\begin{aligned} P_s &= 2a\Big[\frac{1-\mu_c}{2}\Big]^L \sum_{l=0}^{L-1} \binom{L-1+l}{l} \Big[\frac{1+\mu_c}{2}\Big]^l - a^2 \times \Big\{\frac{1}{4} - \frac{\mu_c}{\pi}\Big[(\frac{\pi}{2} - \tan^{-1}\mu_c) \sum_{l=0}^{L-1} \frac{\binom{2l}{l}}{[4(1+\frac{b^2\overline{\gamma}_b}{2})]^l} \\ &\quad - \sin(\tan^{-1}\mu_c) \sum_{l=1}^{L-1} \sum_{i=1}^{l} \frac{T_{il}}{(1+\frac{b^2\overline{\gamma}_b}{2})^l} [\cos(\tan^{-1}\mu_c)]\Big]^{2(l-i)+1}\Big\} \end{aligned} \tag{3.25}$$

where $T_{il} = \frac{\binom{2l}{l}}{\binom{2(l-i)}{l-i}[4^i[2(l-i)+1]]}$ and $\mu_c = \sqrt{\frac{b^2\overline{\gamma}_b}{2+b^2\overline{\gamma}_b}}$. A similar expression can be written for the probability of bit error for M-QAM modulations.

3.5.4 Performance of Viterbi decoder

In 802.11a/g, we assume that the receiver uses FEC Viterbi decoding. The upper bound probability of error provided in [24] is used under the assumption of binary convolutional coding and hard-decision Viterbi decoding. Specifically, for a packet of length L this probability is:

$$P_e(L) \leq 1 - (1 - P_u)^{8L}, \tag{3.26}$$

where the union bound P_u of the first-event error probability is given by

$$P_u = \sum_{d=d_{free}}^{\infty} a_d \times P_d \tag{3.27}$$

d_{free} is the free distance of the convolutional code and a_d is the total number of error events of weight d. Table 3.3 shows the values of a_d coefficients and d_{free} provided in [25] for available FEC coder in 802.11a/g standard.

Table 3.3: Distance spectrum codes for available FEC encoders in IEEE 802.11a, Generator (133,171), K=7, and n=2.

FEC rate	d_f	$(a_{d_f}, a_{d_f+1}, a_{d_f+2}, \cdots)$
1/2	10	(11, 0, 38, 0, 193, 0, 1331, 0, 7275, 0, 40406, 0, 234969, 0, 1337714, 0, 7594819, 0, 433775588, 0, $\cdots$)
2/3	6	(1, 16, 48, 158, 642, 2435, 9174, 34701, 131533, 499312, $\cdots$)
3/4	5	(8, 31, 160, 892, 4512, 23297, 120976, 624304, 3229885, 16721329, $\cdots$)

P_d is the probability that an incorrect path at distance d from the correct path is chosen by the Viterbi decoder. When hard decision decoding is applied, P_d is given by:

$$P_d = \begin{cases} \sum_{k=(d+1)/2}^{d} \binom{d}{k} \rho^k (1-\rho)^{d-k} & d \text{ is odd} \\ \frac{1}{2}\binom{d}{d/2} \rho^{d/2}(1-\rho)^{d/2} + \sum_{k=d/2+1}^{d} \binom{d}{k} \rho^k (1-\rho)^{d-k} & d \text{ is even} \end{cases} \tag{3.28}$$

where ρ is the bit error probability for the physical modulation. The probability of bit error can be calculated as explained above considering the wireless channel and receiver structure.

3.5.5 Simulation Results and Discussion

We have evaluated the performance of all available modulations in IEEE 802.11a. Figure 3.5 shows the probability of bit error for BPSK, QPSK, 16-QAM, and 64-QAM modulations. We obtained these plots using the equations from Table 3.1, 3.2 and Equation (3.24). In this simulation study, the probability of bit error is calculated for Rayleigh fading channel with one path ($L = 1$), two paths ($L = 2$) and for an AWGN channel (i.e., $L = \infty$ paths). The results confirm that the BPSK modulation (i.e., basic mode) has the minimum BER comparing to other modulations for all channel models. Another important observation from the simulation results is that for a target BER, which is usually 10^{-5} for most of the applications, the SNR varies based on receiver structure dramatically. For example in 64-QAM using two paths at the receiver the required energy per bit is decreased by 40% compared to the Rayleigh fading channel with one path.

Now we consider the available data rates in 802.11a and use Equation (3.9) to evaluate the performance of the above modulations in 802.11a. In this manner, we plot the BER based on SNR instead of $\frac{\mathcal{E}_b}{N_o}$. The results are shown in Figure 3.6 for AWGN channel and Rayleigh fading channel with one path. Note that in this simulation we do not use FEC coding neither Viterbi decoding. The results show that the probability of bit error for a specific SNR increases when the data transmission rate increases.

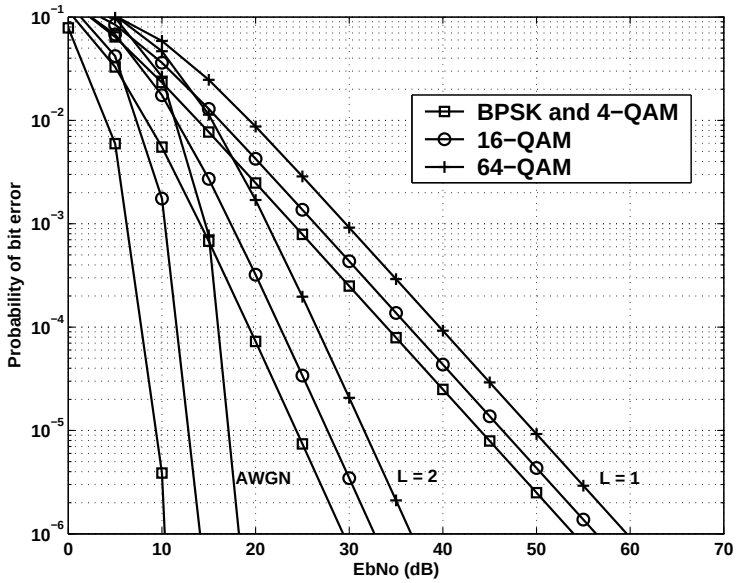


Figure 3.5: Probability of bit error for various modulations.

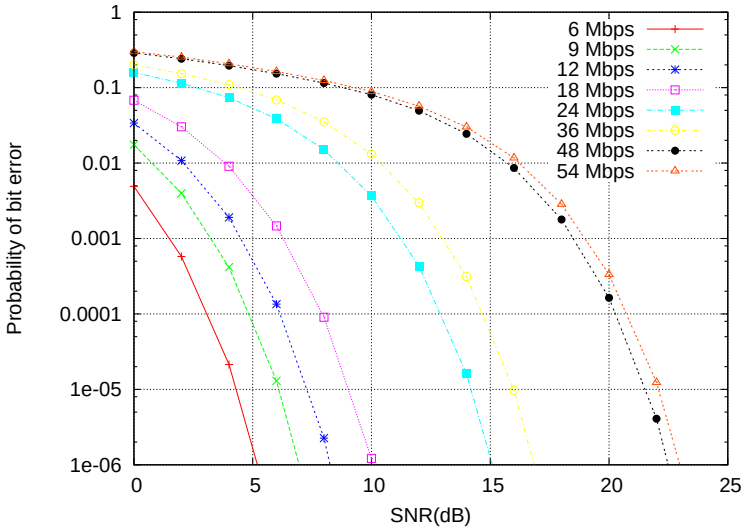
Finally, we have obtained the upper bound for the probability of packet error rate for the different 802.11a transmission modes in an AWGN channel at the output of the Viterbi decoder using Equation (3.26). The results are shown in Figure 3.7 for a packet length equal to 500 and 1500 bytes. One interesting observation is that this probability for mode 3 (12 Mbps) which uses QPSK and FEC=1/2 is always less than that of mode 2 (9 Mbps) which uses BPSK and FEC=3/4. This shows that the combination of BPSK, 9 Mbps data rate , and FEC=3/4 is useless in the IEEE 802.11a standard.

3.6 IEEE 802.11b Performance Evaluation

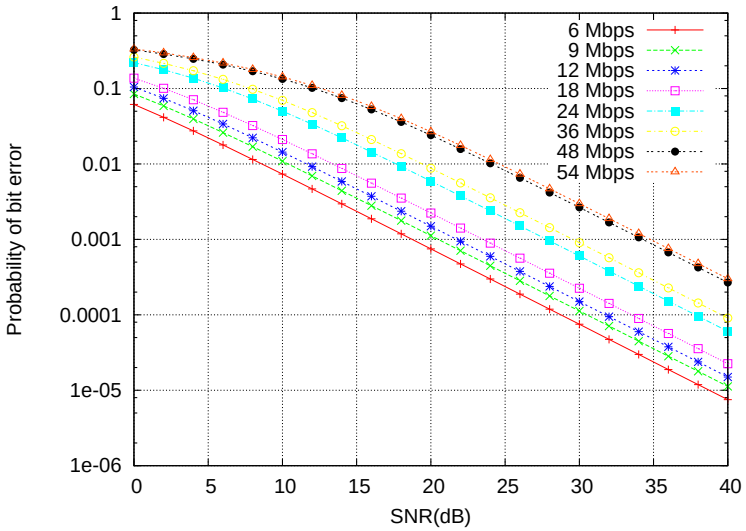
As explained in Section 2.8, four transmission modes are defined in 802.11b. In this section, we evaluate the performance of these transmission modes.

3.6.1 1 and 2 Mbps Transmission Modes

In this section, we focus on the BER calculation of the 1 and 2 Mbps transmission modes in IEEE 802.11b WLANs. As explained in Section 2.8, for 1 and 2 Mbps transmission modes the modulations could be DBPSK and DQPSK or DE-BPSK and DE-QPSK according to receiver structure. It can be shown that in AWGN with Gray mapping, the probability of bit error for DE-BPSK and DE-QPSK are [26, 21]:

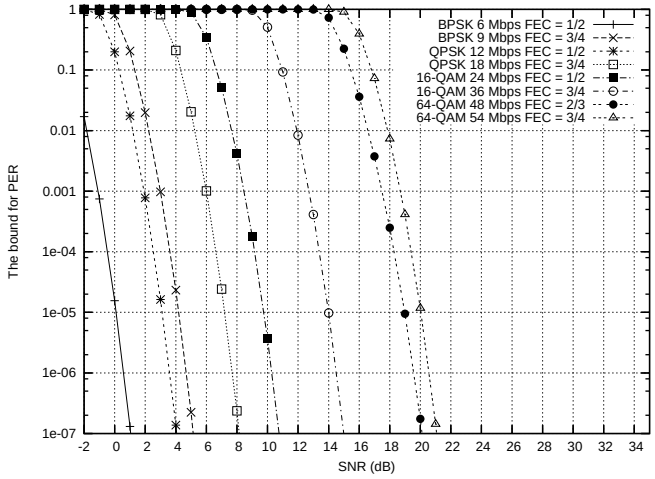


(a) AWGN

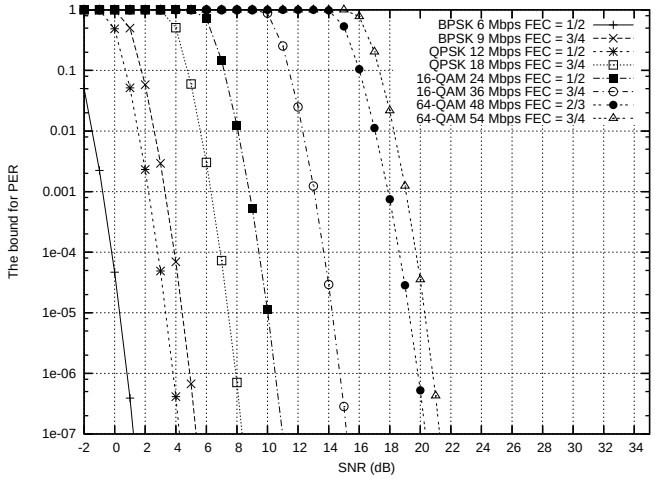


(b) Rayleigh Fading (L=1)

Figure 3.6: Probability of bit error at 802.11a without FEC and Viterbi decoder.



(a) L=500 Bytes



(b) L=1500 Bytes

Figure 3.7: The upper bound for the PER in 802.11a.

$$\begin{aligned} P_b^{DE-BPSK} &= P_b^{DE-QPSK} \\ &= 2Q\Big(\sqrt{2\frac{\mathcal{E}_b}{N_o}}\Big) - 2Q^2\Big(\sqrt{2\frac{\mathcal{E}_b}{N_o}}\Big) \end{aligned} \tag{3.29}$$

Similar to the calculations performed in Section 3.5.2 and 3.5.3, if we assume that the fading process is Rayleigh distributed with L resolvable i.i.d. paths, it can be shown that with maximum-ratio diversity combining, the fading-averaged BER for DE-BSPK and DE-QPSK can be obtained by [22]:

$$\begin{aligned} P_b &= 2\Big[\frac{1-\mu_c}{2}\Big]^L \sum_{l=0}^{L-1} \binom{L-1+l}{l} \Big[\frac{1+\mu_c}{2}\Big]^l - 2 \times \Big\{\frac{1}{4} - \frac{\mu_c}{\pi}\Big[(\frac{\pi}{2} - \tan^{-1}\mu_c) \sum_{l=0}^{L-1} \frac{\binom{2l}{l}}{[4(1+\overline{\gamma}_b)]^l} \\ &\quad - \sin(\tan^{-1}\mu_c) \sum_{l=1}^{L-1} \sum_{i=1}^{l} \frac{T_{il}}{(1+\overline{\gamma}_b)^l} [\cos(\tan^{-1}\mu_c)]\Big]^{2(l-i)+1}\Big\} \end{aligned} \tag{3.30}$$

where $T_{il} = \frac{\binom{2l}{l}}{\binom{2(l-i)}{l-i}\{4^i[2(l-i)+1]\}}$ and $\mu_c = \sqrt{\frac{\overline{\gamma}_b}{1+\overline{\gamma}_b}}$. For the non-coherent detection receiver, i.e. DBPSK and DQPSK, the probability of bit error for AWGN is:

$$P_b^{DBPSK} = \frac{1}{2} exp(-\frac{\gamma_b}{2}) \tag{3.31}$$

and,

$$P_b^{DQPSK} = \frac{1}{2}[1 - Q_1(b,a) + Q_1(a,b)], \tag{3.32}$$

where $a = \sqrt{2\gamma_b(1-\sqrt{2}/2)}$ and $b = \sqrt{2\gamma_b(1+\sqrt{2}/2)}$. For the receiver with maximum-ratio diversity combining the probability of bit error for DBPSK and DQPSK can be obtained by [21]:

$$P_b = \frac{1}{2}\Big[1 - \frac{\mu_c}{\sqrt{2-\mu_c^2}} \sum_{l=0}^{L-1} \binom{2l}{l} \Big(\frac{1-\mu_c^2}{4-2\mu_c^2}\Big)^l\Big], \tag{3.33}$$

where $\mu_c = \frac{\overline{\gamma}_b}{1+\overline{\gamma}_b}$ for DBPSK and $\mu_c = \frac{2\overline{\gamma}_b}{1+2\overline{\gamma}_b}$ for DQPSK.

3.6.2 5.5 and 11 Mbps Transmission Modes

For the high data rate transmission modes in 802.11b, i.e. 5.5 and 11 Mbps, the CCK and binary convolutional coding have been adopted as explained in Section 2.8.2. The latter one is an optional transmission mode in 802.11b. In this section, we focus on the performance evaluation of 5.5 and 11 Mbps transmission modes with the CCK modulation. The performance of

CCK modulation (i.e., BER) directly depends on the equalizer used (see Figure 3.2). Basically there are three common equalizers: *linear equalizer* (LE), *decision feedback equalizer* (DFE), and *maximum likelihood sequence estimator* (MLSE). Note that it is too complex to calculate the closed form equations for BER calculations for CCK modulation as we have done for other modulations. In this dissertation we *only* address how we can derive the performance evaluation of CCK modulation as it is presented in [27] for the MLSE equalizer. The simulation results for MLSE and DFE detections are available in [27]. We can also find the empirical curves provided by Intersil for its chipset called HFA3861B which shows the BER versus SNR for these two modes in [28].

To evaluate the performance of CCK modulations in 802.11b we first calculate the *pairwise error probability* (PEP). PEP or $P_e(\alpha, \beta)$ is the probability that a code word c^α is transmitted and that the maximum likelihood receiver decides in favor of c^β where $\alpha \neq \beta$. In [29], the authors show that $P_e(\alpha, \beta)$ can be calculated by:

$$P_e(\alpha, \beta) = \int_{x=-\infty}^{0} p_{\Delta(\alpha,\beta)}(x)dx \tag{3.34}$$

where $\Delta(\alpha, \beta)$ is a random variable which depends on the mean and the autocorrelation matrix of h (i.e., the *channel impulse response* (CIR) between transmit and receive antenna) with pdf $p_{\Delta(\alpha,\beta)}$. We then calculate the union bound on the average bit error rate by,

$$P_b \leq \frac{1}{|C|} \sum_{\alpha=1}^{|C|} \sum_{\substack{\beta=1 \\ \beta\neq\alpha}}^{|C|} \frac{n(\alpha, \beta)}{8} \times P_e(\alpha, \beta), \tag{3.35}$$

where $n(\alpha, \beta)$ denotes the number of bit errors if c^α is transmitted and c^β is detected.

3.6.3 Simulation Results and Discussion

Figure 3.8 and 3.9 show the probability of bit error for all the possible modulations in IEEE 802.11b at 1 and 2 Mbps. These probabilities are calculated for differentially encoded and coherently (DE-BPSK and DE-QPSK) or differentially (DBPSK and DQPSK) detected. The probability of bit error is plotted based on $\frac{\mathcal{E}_b}{N_o}$ and SNR in Figures 3.8 and 3.9 respectively. Again note that DE-QPSK has the same probability of bit error than DE-BPSK since it can be considered (using in-phase and quadrature demodulators) as two parallels, independently detected DE-BSPK. For differential detection, the phase difference is detected over two consecutive symbols. Figure 3.9 shows that differentially encoded modulations obtain always better performance compared to differentially detected modulations. Another interesting observation is again the influence of receiver structure over the performance of the wireless device: the values of bit error are highly dependent on the type of demodulator/detector, the number of receiving antennas, the type of diversity combining and the severity of multipath fading.

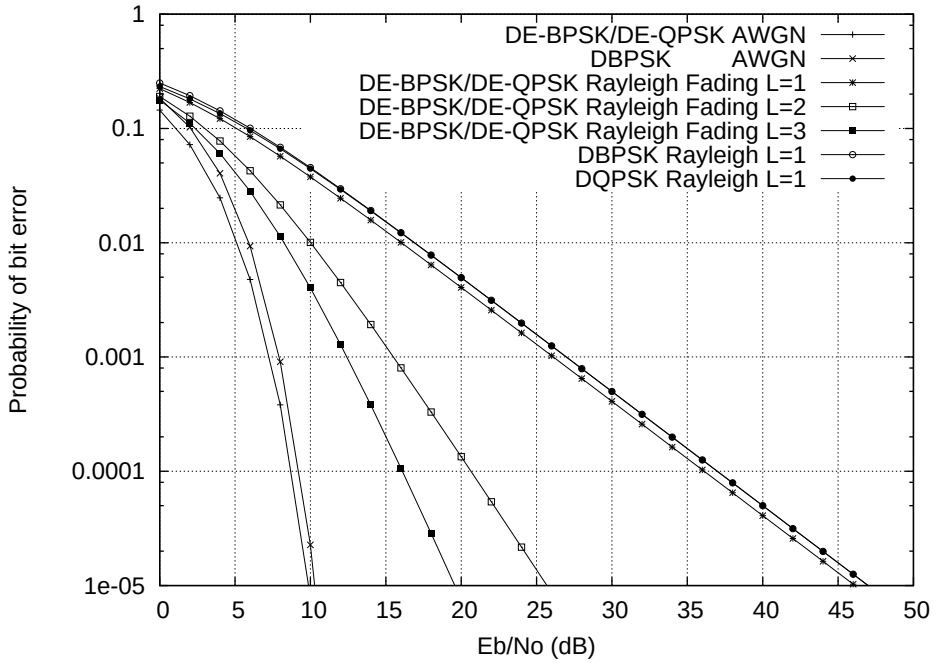


Figure 3.8: Probability of bit error for DE-BPSK, DE-QPSK, DBPSK, and DQPSK modulations.

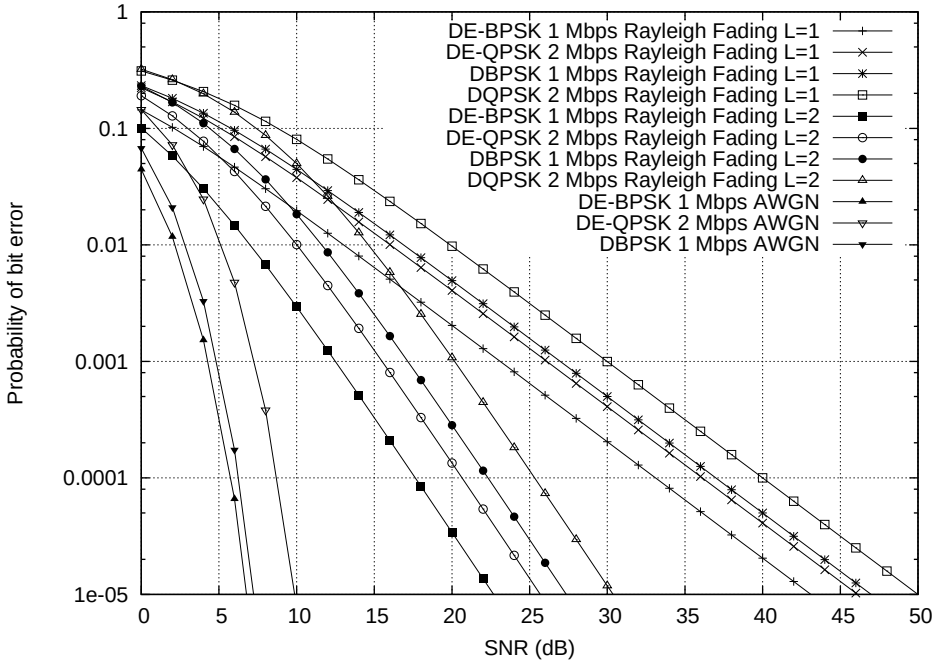


Figure 3.9: Performance evaluation of 1 and 2 Mbps transmission modes in 802.11b.

3.7 Conclusions

In this chapter, we presented a complete performance evaluation of data transmission modes for IEEE 802.11 a/b. We investigated the performance of transmission modes by calculating the probability of bit error over AWGN and Rayleigh fading channels. We discussed the receiver structure for 802.11 a/b mobile STAs and we evaluated the performance of the transmission modes for multilink receiver STAs as well. We also evaluated the performance of FEC (binary convolutional coding) in the PHY layer considering Viterbi decoding at the receiver. Then, we discussed the performance of the CCK modulation in 802.11b. The simulation results showed that the performance of transmission modes are highly dependent upon several parameters. Among them, the wireless channel model, the type of signal detection (coherent or differential detection), the type of encoders and decoders, the number of receiver antennas, data modulations and data rates have important influence over the performance of data transmission modes. These parameters will be taken into account in the following sections, where we propose new cross layer mechanisms for adaptive communications.

4

IEEE 802.11 MAC and Physical Layer Modelling

4.1 Summary

In this chapter, we present an analytical model that accounts for the positions of stations with respect to the access point (i.e., PHY layer) while evaluating the performance of 802.11 MAC layer. Our work is based on the Bianchi's model where the performance of the 802.11 MAC layer is computed using a discrete time Markov chain, but where all stations are implicitly assumed to be located at the same distance of the access point. In our model, given the position of one station, we compute its saturation throughput while conditioning on the position of the other concurrent stations. Furthermore, our model provides the total saturation throughput of the medium. We solve the model numerically and we show that the saturation throughput per station is strongly dependent not only on the position of the station but also on the positions of the other stations. Results confirm that a station achieves a higher throughput when it is closer to the AP but bring out that there is a distance threshold above which the throughput decrease is fast and significant. When a station is far from the AP compared to the other stations, it will end up by contending for the bandwidth not used by the other stations. We believe that our model is a good tool to dimension 802.11 wireless access networks and to study their capacities and their performances.

4.2 Introduction

Recently, different analytical models and simulation studies have been elaborated to evaluate the 802.11 MAC layer performance. These studies mainly aim at computing the saturation throughput of the MAC layer and focus on its improvement. One of the most promising models has been the so-called Bianchi model [30]. It provides closed-form expressions for the saturation throughput and for the probability of a packet transmission failure due to collision.

The modelling of the 802.11 MAC layer is an important issue for the evolution of this technology. One of the major shortcomings in most of the existing models is that the PHY layer conditions are not considered. The existing models for 802.11 assume that all STAs have the same physical conditions at the receiving STA (same power, same coding, same modulation, ...), so when two or more STAs emit a packet in the same slot time, all their packets are lost, which may not be the case in reality when for instance one STA is close to the receiving STA and the other STAs far from it [31]. This behavior, called the *capture effect*, can be analyzed by considering the spatial positions of the STAs. In [32] the spatial positions of STAs are considered for the purpose of computing the capacity of wireless networks, but only an ideal model for the MAC layer issued from the information theory is used.

Our main contribution in this chapter is considering both the PHY and MAC layer protocols to analyze the performance of the existing IEEE 802.11 standard. Our work reuses the model for 802.11 MAC layer from [30, 33] and extends it to consider interference from other STAs. Finally, we are able to compute, for a given topology, the throughput of any wireless STA using the 802.11 MAC protocol with a specific PHY layer protocol.

Without losing the generality of the approach, we only consider in this thesis traffic flows sent from the mobile STAs in direction of the AP. The case of bidirectional traffic is a straightforward extension; we omit it to ease the exposition of our contribution. Further, we assume that all STAs use the *distributed coordination function* (DCF) (see Section 2.4) of 802.11 and that they always have packets to send (i.e., saturated sources). We present an evaluation of our approach for 802.11b with data rates equal to 1 and 2 Mbps. The results indicate that the model leads to a very accurate analysis. Note that the model can simply employ other transmission modes considering their probability of bit error and packet error rate as explained in Section 3.5 and Section 3.6.

The chapter is structured as follows. We first discuss some related works on MAC and PHY layer modelling in IEEE 802.11 in Section 4.3. In Section 4.4, we present our model and derive the characterizing equations for it. The numerical and simulation results obtained are presented in Section 4.6. Section 4.7 concludes this chapter with some pointers for future work.

4.3 Related Works

There have been various attempts to model and analyze the saturation throughput and delay of the IEEE 802.11 DCF protocol since the IEEE 802.11 standards have been proposed. We divide these models into two main categories: MAC layer models and PHY layer models.

4.3.1 IEEE 802.11 MAC layer models

As explained in the introduction, there are different analytical models and simulation studies that analyze the performance of the 802.11 MAC layer. As an example, Foh and Zuckerman present the analysis of the mean packet delay at different throughput for IEEE 802.11 MAC in [34]. Kim and Hou [35] analyze the protocol capacity of IEEE 802.11 MAC with the assumption that the number of active stations having packets ready for transmission is large. In [36] and [37] Chatzimisios et al. have suggested some extensions to the model proposed in [30] to evaluate the packet delay, the packet drop probability and the packet drop time. Since in our model we have used the Bianchi's model [30] and its extension proposed in [33] by Wu et al., we detail these models in this section.

The Bianchi's model uses a simple and elegant discrete-time Markov chain to analyze the case of saturated STAs, i.e. STAs that always have packets to send. Wu et al. [33] proposed a scheme named DCF+ to enhance the performance of reliable transport protocol over WLAN and analyzed it with an extension of Bianchi's model by considering finite packet retry limits as defined in the IEEE 802.11 standard.

The retransmission limit is defined in the IEEE 802.11 MAC standard specification with the help of the two following counters: *short retry count* (SRC) and *long retry count* (LRC). These counters are incremented and reset independently. SRC is incremented every time an RTS fails whereas LRC is incremented when data transmission fails. Both SRC and LRC are reset to zero after a successful data transmission. Data frames are discarded when LRC (SRC) reaches *dot11LongRetryLimit* (*dot11ShortRetryLimit*). The default values for *dot11LongRetryLimit* and *dot11ShortRetryLimit* are 4 and 7 respectively.

Considering this limitation, the Markov chain proposed by Bianchi is modified in [33] as shown in Figure 4.1. Unlike paper [30], in the Wu's model, m is the maximum backoff stage (retransmission count) which is different for the data and RTS frames. In Wu's model, m' represents the maximum contention window, i.e. $2^{m'}(CW_{\min}+1) = (CW_{\max}+1)$. In fact, the key difference between Bianchi's model [30] and Wu's model [33], concerns the Markov chain models. They are different because Wu's model considers the effects of frame retransmission limit.

In Bianchi's model, the time is divided into slots of variable duration based on what happens during a slot: no transmission, correct transmission, or collision. The model computes among

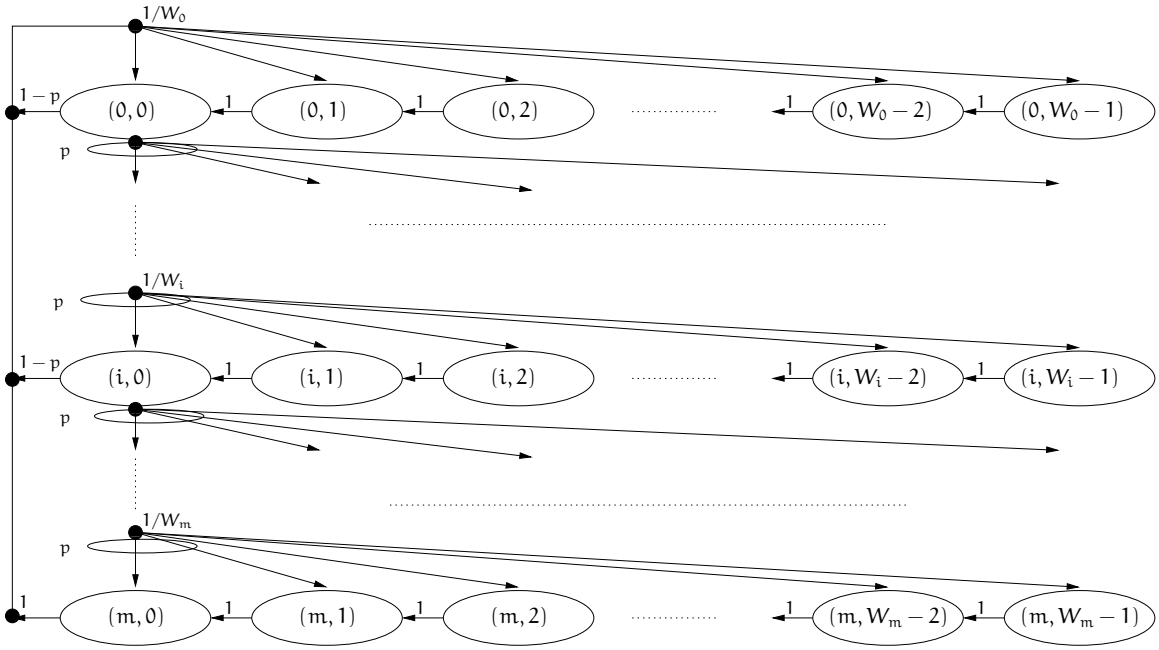


Figure 4.1: Markov chain model of backoff window size in CSMA/CA.

others, the probability that a station transmits in a slot τ, the probability that a transmitted packet collides with other transmissions p, and the saturation throughput of a station $Z(p,\tau)$, which is a function of τ and p as well as other physical parameters. The packet loss probability is computed as:

$$p = 1 - (1-\tau)^{n-1} \tag{4.1}$$

where n is the total number of STAs. The model also provides the expression of τ as a function of the packet loss probability p using the Markov chain that describes the system, see Figure 4.1. Let **B** be the function relating p and τ in this model, then:

$$\tau = \mathbf{B}(p) = \frac{1-p^{m+1}}{1-p} b_{0,0} \tag{4.2}$$

and $b_{0,0}$ be the stationary probability to find the Markov chain in state $(0,0)$. $b_{0,0}$ can be obtained by solving the Markov chain as shown in Equation (4.3).

$$b_{0,0} = \begin{cases} \frac{2(1-2p)(1-p)}{W(1-(2p)^{m+1})(1-p)+(1-2p)(1-p^{m+1})} & m \le m' \\ \frac{2(1-2p)(1-p)}{W(1-(2p)^{m'+1})(1-p)+(1-2p)(1-p^{m+1})+W2^{m'}p^{(m'+1)}(1-2p)(1-p^{m-m'})} & m > m' \end{cases} \tag{4.3}$$

Equations (4.1) and (4.2) are solved for the values of p and τ. Once these probabilities are obtained, this model computes the saturation throughput $Z(p, \tau)$ of a station. The expression of the throughput is given in Section 4.5, where we adapt this model to our context.

None of the above models have considered the channel characteristics of the PHY layer. Actually, there are a few studies that consider the PHY layer characteristics in 802.11 WLANs. We overview them in what follows.

4.3.2 IEEE 802.11 PHY Layer Models

In [38], the impact of an error-prone channel over all performance measures is analytically analyzed for a traffic saturated IEEE 802.11 WLAN. A modified Markov chain is used to compute the transmission probability per station. The Markov chain is for the backoff window size that considers the frame-error rates and the maximal allowable number of retransmission attempts. The transition probability from one stage to another is denoted by p. It is also the probability of an unsuccessful (re)transmission attempt perceived by a test station as its frame is being transmitted on the channel. In this model [38], the authors supposed that the unsuccessful (re)transmission attempt can happen due to *collision* of this station with at least one of the $n-1$ remaining stations, occurring with probability p_1:

$$p_1 = 1 - (1 - \tau)^{n-1} \tag{4.4}$$

and/or an *error frame*, occurring with probability P_f (due to the channel fading and/or noise). The authors supposed that both events are independent and the probability p can be expressed as:

$$p = 1 - (1 - p_1)(1 - P_f) \tag{4.5}$$

Other calculations are similar to Bianchi's Markov model, see Equation (4.2). Similarly to Bianchi's model [30], they express the normalized saturation throughput of IEEE 802.11 DCF within a single WLAN cell in an error-prone channel [38].

In [39], an improved analytical model that calculates IEEE 802.11 DCF performance taking into account both packet retry limits and transmission errors for the IEEE 802.11a protocol is proposed. The authors have validated their model using the OPNET™ simulator. Their analysis is very similar to the model presented in [38]. The main difference with Bianchi's model is how they calculate p. In this model, p stands for the probability that a transmitted packet encounters a collision or is received in error. The probability p is expressed as:

$$p = 1 - (1 - \tau)^{n-1}(1 - BER)^{l+H} \tag{4.6}$$

Where BER is the bit error rate, l is the packet size in bits, H is the packet header size in bits and τ is the probability that a station transmits a packet in a randomly chosen slot time given by Equation (4.2). Again, this leads to a non-linear system that can be solved using numerical methods to obtain τ and p. They have also calculated the saturated throughput and delay with the method presented in [38].

Finally, [40] proposes an analytical model to compute the throughput for the single and multi-user cases with a non ideal channel. This model is validated with system simulations in realistic deployment scenarios. Some changes in Bianchi's Markov chain model are proposed as well. In addition the authors have assumed that the probability P_j, (the failure probability viewed by the station when a packet is transmitted) is due to either a collision on the channel (with probability P_{cj}) or without collision but error transmission (with probability PER). Hence:

$$P_j = P_{cj} + (1 - P_{cj})PER. \tag{4.7}$$

None of these models have considered specific physical aspects such as the modulation, FEC, the PLCP format in IEEE 802.11 or the channel characteristics (especially, the distance between the source and the destination). Our approach tries to provide more precise results considering all these characteristics.

4.4 Our approach: Distance Aware Model

Our model considers the interference from the other STAs and the background noise to compute the packet loss probability p, we call it the *distance aware* (DAW) model. The expression for the transmission probability τ remains the same as that in Equation (4.2). The computation of *packet error rate* (p) is done under the assumptions presented in Section 2.5 and Section 3.6.1.

Let us consider an STA k that transmits a packet to the AP and compute the probability that this packet is lost (i.e., cannot be decoded correctly). We suppose that this STA is located at distance d_k from the AP. We denote its packet loss probability by $p_k(d_k)$. We suppose that all stations use the long PPDU format in 802.11b. Our model can be employed for all other transmission modes and for all standards provided that the packet error rate is calculated (see Chapter 3). Here we evaluate our model in 802.11b where STAs use transmission rate equal to 1 and 2 Mbps. Considering the data packet format shown in Figure 2.5, the probability of packet error is:

$$P_k = 1 - (1 - P_b^{PLCP})^{L_{PLCP}} \cdot (1 - P_b^{Payload})^{L_{Payload}}, \tag{4.8}$$

where L_{PLCP} and $L_{Payload}$ are PLCP and Payload length respectively, and P_b^x is the bit error probability for part x of the packet (PLCP or Payload). If we assume that the noise over the

wireless channel is white Gaussian with spectral density equal to $N_0/2$, P_b^x can be computed using Equation (3.29) considering transmission mode and data rate. This expression for P_k assumes that the bit error process is iid during the reception of the packet and that the data is not protected by any channel coding scheme. As we need the packet loss probability averaged over all values of bit errors, we focus on the computation of the expected value $p_k(d_k) = \mathbb{E}[P_k]$.

To do so, we need to decompose the SNR into identically distributed elements for which a pdf can be defined. Once such a pdf is found, we can obtain $p_k(d_k)$ by substituting this pdf in Equation (4.8) and taking the expectation. In order to decompose the SNR, we first introduce the Bernoulli random variables Y_i ($i = 1, \ldots, n$), being equal to 1 when STA i transmits a packet in a slot time, and equal to 0 otherwise. In the next step we look for the power of signal transmitted by STA i to the AP. We denote such power with X_i and we define it as;

$$X_i = Y_i \cdot L(D_i), \tag{4.9}$$

where $L(D_i)$ expresses the power with which the signal of STA i arrives at the AP after being attenuated over distance D_i and calculated using a simple path loss model;

$$L(D_i) = \frac{P_0}{D_i^{\alpha}}. \tag{4.10}$$

In this expression, P_0 denotes the STA transmission power and α, the path loss exponent, determines the loss rate. We use $\alpha = 3$ which is commonly used to model loss in an urban environment [41]. Note that this model for the power only considers the attenuation caused by the distance between the emitting terminal and the AP, and ignores other factors such as mobility, shadowing, multi-path fading, etc. Nevertheless, we can use other channel models in Chapter 3 to have a more precise evaluation of MAC/PHY layer. Using the power of each STA at the AP, we can compute the interfering power a packet transmitted by STA k faces. We denote this power by I_k and write it as;

$$I_k = \sum_{\substack{i=1 \\ i \neq k}}^{n} Y_i \cdot L(D_i). \tag{4.11}$$

This allows us to write the following expression for the SNR at the AP of a packet/signal coming from STA k at the given distance d_k:

$$SNR_k = \frac{L(d_k)}{N_0 + I_k} = \frac{L(d_k)}{N_0 + \sum_{i=1(i \neq k)}^{n} Y_i \cdot L(D_i)}. \tag{4.12}$$

In this expression, N_0 stands for the background noise (see Equation (3.10)). We can see that to compute P_k using Equation (4.8), the only random variable is I_k. Hence, having the pdf

of I_k, which we denote by $f_{I_k}(x)$, we can compute $p_k(d_k) = \mathbb{E}[P_k]$. Assuming independence of Y_i, as in the Bianchi's model, $f_{I_k}(x)$ can be expressed as an $n-1$ convolution:

$$f_{I_k}(x) = f_{X_1} \otimes \cdots \otimes f_{X_{k-1}} \otimes f_{X_{k-2}} \otimes \cdots \otimes f_{X_n}(x). \tag{4.13}$$

In the analysis above, we kept the distance from STA i to the AP random denoted by D_i, except for STA k for which we are computing p_k. Then, we compute p_k for two cases. First, we compute it when the positions of stations are known (the D_i are deterministic): the only randomness in this case lies in the dynamics of the MAC layer. Second, we compute p_k for a more general case where nodes are uniformly distributed in the plane.

4.4.1 Fixed Topologies

Suppose that we are given the distance vector $\overline{D} = \{d_1, \ldots, d_n\}$, where d_i stands for the distance of STA i to the access point. Since all distances are fixed, we omit in this section the index of distance from loss and transmission probabilities. For an STA k, we aim at finding the pdf of I_k. I_k gives the interfering power produced by all the other STAs at the AP. To compute I_k, we need $f_X(x)$, the pdf of the power at the AP of an individual STA. For STA i, $f_{X_i}(x)$ can be written as:

$$f_{X_i}(x) = \delta_x(0)(1-\tau_i) + \delta_x(L(d_i))\tau_i, \tag{4.14}$$

where $\delta_x(x_0)$ is a Dirac pulse at $x = x_0$ and τ_i denotes the transmission probability of STA i. $f_{I_k}(x)$ can be computed using Equation (4.13), and the values τ_i in $f_{I_k}(x)$ are left unknown. Using Equation (4.8) and taking the expectation value, we get the packet loss probability of STA k:

$$p_k = \mathbb{E}[1 - (1 - P_b^{PLCP})^{L_{PLCP}} \cdot (1 - P_b^{Payload})^{L_{Payload}}] \tag{4.15}$$

For example, when the packet is sent with 1 Mbps, the PLCP and the payload are sent with the same modulation and data rate, so the equation can be simplified as[1]:

$$P_k = 1 - \int_{x=0}^{\infty} \left(1 - Q\left(\sqrt{\frac{2L(d_k)W}{(N_0 + x)R}}\right)\right)^{L_{PLCP}+L_{Payload}} f_{I_k}(x)dx. \tag{4.16}$$

[1]Without loss of generality and to simplify the expression of the PER in this section, we use the expressions for probability of bit error rates for BPSK and QPSK, although 802.11 standards use differentially encoded versions DBPSK and DQPSK for 1 and 2 Mbps respectively, as explained in Section 3.6.1. It is worth mentioning that for coherent detection for high SNR values, classical BER expressions are applicable to both classical and differential modulation schemes.

This expression of p_k is a function of the transmission probabilities of the other STAs via the pdf functions f_{X_i}. From Bianchi's model, the transmission probability of an STA is related to its collision probability via the function in Equation (4.2) (by substituting p by p_k and τ by τ_k). Thus, using Equations (4.2) and (4.15), we set up a non linear system of equations, which can be solved numerically for all p_k and τ_k. Having the p_k and τ_k, the throughput of any STA k can be computed. This computation is shown in Section 4.5.

4.4.2 Random Topologies

We consider now the case where the STAs are uniformly distributed in a disk of radius r around the AP. Thus, the pdf of D (the distance to the AP of an STA) has the following form:

$$f_D(d) = \{1_{1 \leq d \leq r}\} \frac{2d}{r^2}. \tag{4.17}$$

Consider an STA k located at distance d_k from the AP, and let us focus on computing its average performance over all possible positions of the concurrent STAs. As in the fixed topology case, we have to find the pdf of I_k, i.e., the interference caused by the other STAs at the AP.

However, the computation of $f_{X_i}(x)$ (the pdf of signal power at the AP of a random STA i) becomes more complex. We first write the cumulative distribution function of X_i (for $x \geq 0$),

$$F_{X_i}(x) = (1 - \mathbb{E}[\tau_i(D_i)]) + 1_{\{P_0/r^\alpha \leq x \leq P_0\}} \mathbb{E}[\tau_i(D_i)] \left(1 - \frac{(L^{-1}(x))^2}{r^2}\right). \tag{4.18}$$

In this equation, $\mathbb{E}[\tau_i(D_i)]$ is the transmission probability of an STA i averaged over all its possible locations. By differentiation and using the expression of $L(D_i)$, we find the pdf of X_i,

$$f_{X_i}(x) = \delta_x(1 - \mathbb{E}[\tau_i(D_i)]) + 1_{\{P_0/r^\alpha \leq x \leq P_0\}} \frac{2}{\alpha r^2 x} \left(\frac{P_0}{x}\right)^{2/\alpha} \mathbb{E}[\tau_i(D_i)]. \tag{4.19}$$

Assume that the transmission probability of a random STA i is only dependent on its own position and independent of that of the others. Only the number of the other STAs is supposed to influence the transmission probability of STA i. This is the case when the number of STAs is large. Under this assumption, the variables X_i are independent of each other. We can therefore compute the pdf of I_k using Equation (4.13) and (4.19).

Note that f_{I_k} is a function of one unknown $\mathbb{E}[\tau_i(D_i)]$. The packet collision probability can be obtained by plugging f_{I_k} in Equation (4.15). Then, we substitute the expression of $p_k(d_k)$ in Equation (4.2) to get $\tau_k(d_k)$, the probability with which STA k transmits a packet in a slot time averaged over all possible positions of the other STAs. Finally, the throughput of STA k averaged over all locations of the other STAs can be computed in a similar way to the fixed case as will be discussed in Section 4.5.

Now we explain how to find the expression of $\mathbb{E}[\tau_i(D_i)]$. To solve for this expectation, we write an implicit equation with $\mathbb{E}[\tau_i(D_i)]$ as a variable, then we solve this equation numerically. Equations (4.19) and (4.15) give us the expression of $p_k(d_k)$ for STA k as a function of $\mathbb{E}[\tau_i(D_i)]$. Denote by $p_k(d_k) = G(d_k, \mathbb{E}[\tau_i(D_i)])$ this expression. Using 4.15, we can write;

$$\tau_k(d_k) = B(p_k(d_k)) = B(G(d_k, \mathbb{E}[\tau_i(D_i)])) \tag{4.20}$$

We get our implicit equation in $\mathbb{E}[\tau_i(D_i)]$ by summing over all the values of d_k, as shown in Equation (4.21).

$$\mathbb{E}[\tau_i(D_i)] = \mathbb{E}[B(p_k(d_k))] = \int_0^r \frac{2\pi\rho d\rho}{\pi r^2} B(G(\rho, \mathbb{E}[\tau_i(D_i)])). \tag{4.21}$$

Once we obtain $\mathbb{E}[\tau_i(D_i)]$, all transmission probabilities, collision probabilities and throughput can be obtained using our above analysis. In summary, for an STA located at distance d_k:

- The packet collision probability $p_k(d_k)$ can be obtained by substituting Equation (4.20) in Equation (4.15), where the value of $\mathbb{E}[\tau_i(D_i)]$ is computed numerically with the implicit Equation (4.21).
- The packet transmission probability $\tau_k(d_k)$ is computed by substituting p by $p_k(d_k)$ in Equation (4.2).
- Given $p_k(d_k)$ and $\tau_k(d_k)$, the throughput of STA k can be obtained in a similar way to the Bianchi model. The throughput of a random STA can be computed as well.

4.5 Throughput Calculation

We now derive the throughput of a single STA k at a given distance d_k. In the case of a fixed topology, (Section 4.4.1) this throughput depends on the position of all other STAs and their transmission probabilities τ_i, whereas in the case of random topologies (Section 4.4.2), the throughput depends on the other STAs average location and their average transmission probability $\mathbb{E}[\tau_i(D_i)]$.

Consider first the case of fixed topology. The throughput of an STA k is given by the function $Z(p_k, \tau_k)$, which has the following form:

$$Z(p_k, \tau_k) = \frac{\tau_k(1-p_k)L}{(1-P_{tr})\sigma + P_{tr}P_sT_s + P_{tr}(1-P_s)T_c}. \tag{4.22}$$

In the numerator of the throughput expression, we put the average number of useful bits transmitted in a slot time whereas the denominator corresponds to the average duration of a slot. σ is the physical slot time of the 802.11 MAC layer. T_s and T_c are respectively the duration

of a slot time (following the slot definition in Bianchi's model) when a packet is successfully transmitted and the duration of a slot time when two or more packets collide. L is the payload size. We consider MAC, IP and UDP headers in our calculation for packet length (The sum of these headers is denoted by H, see Section 2.9.). In addition, as explained in Section 2.9, the PHY layer adds to each transmission a constant PLCP preamble and header of total duration t_{PLCP}.

Similar to [33], the slot time duration T_s and T_c for basic access mode (T_s^{bas} and T_c^{bas}) and RTS/CTS access mode (T_s^{rts} and T_c^{rts}), considering ACK timeout, become:

$$\begin{cases} T_s^{bas} = 2t_{PLCP} + DIFS + \frac{H+L}{R_d} + SIFS + \frac{ACK}{R_b} + 2\delta \\ \\ T_c^{bas} = 2t_{PLCP} + DIFS + \frac{H+L}{R_d} + SIFS + \frac{ACK}{R_b} \end{cases} \tag{4.23}$$

$$\begin{cases} T_s^{rts} = 4t_{PLCP} + DIFS + 4SIFS + RTS + CTS + \frac{H+L}{R_d} + \frac{ACK}{R_b} + 2\delta \\ \\ T_c^{rts} = 2t_{PLCP} + DIFS + RTS + SIFS + CTS \end{cases} \tag{4.24}$$

R_b is the data rate for the basic transmission mode (i.e., 1 Mbps for 802.11b) and R_d is the data rate for payload (which is 1 or 2 Mbps in our simulations). In the following, we focus on the basic mode. The throughput calculation for RTS/CTS is straightforward considering Equation (4.3) for $m \geq m'$. δ stands for the propagation delay.

We come now to the definition of P_{tr} and P_s in the denominator of Equation (4.22). With P_{tr}, we denote the probability that at least one of the n STAs is transmitting, which can be formulated as follows:

$$P_{tr} = 1 - \prod_{i=1}^{n}(1 - \tau_i) \tag{4.25}$$

Further, P_s, the probability that such a transmission is successful, is equal to:

$$P_s = \frac{\sum_{i=1}^{n} \tau_i(1 - p_i)}{P_{tr}} \tag{4.26}$$

In the case of random positions, only the expressions of P_{tr} and P_s change. These expressions are shown in Equation (4.27) and Equation (4.28) for an STA k located at distance d_k to the AP.

$$P_{tr} = 1 - (1 - \tau_k(d_k))(1 - \mathbb{E}[\tau_i(D_i)])^{n-1} \tag{4.27}$$

$$P_s = \frac{\tau_k(d_k) \cdot (1 - p_k(d_k)) + (n-1)\mathbb{E}[\tau_i(D_i)](1 - \mathbb{E}[p_i(D_i)])}{P_{tr}} \tag{4.28}$$

Table 4.1 summarizes the parameters used to calculate the throughput in this chapter.

Table 4.1: Simulation parameters used for model evaluation in IEEE 802.11b.

Name	*Value*
CW_{min}	31
m'	5
SLOT time	20 μs
DIFS	50 μs
SIFS	10 μs
PLCPlong	192 μs
ACK	112 bits
RTS	160 bits
CTS	112 bits
L	8000 bits
H	592 bits
P_0	20 mW
α	3
W	2 MHz
N_f	7 dB
k	1.38×10^{-23} Joules/Kelvin
T	290 Kelvin

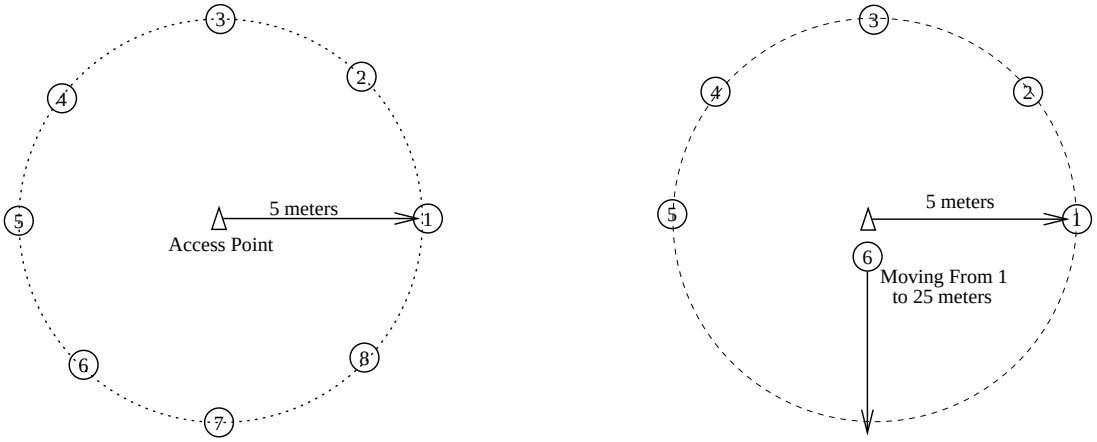
4.6 Model Verification and Simulation Results

We implemented the *DAW* analytical model in MATLAB and compared the results with NS-2 simulations. Our NS-2 simulations[2] are based on the package described in [42]. In this package we consider the effect of wireless physical layer while simulating mobile networks. Physical layer parameters like path loss, interference and noise computation have been added in this NS-2 simulation package. The NS-2 simulation results presented later are averaged over 10 runs with different random seeds.

We first consider a fixed topology, so the D_i values are deterministically set. For this scenario, we used 2 network configurations as shown in Figure 4.2. The first configuration consists of one AP and several STAs which send CBR packets at saturation rate to the AP using UDP connections. All STAs are located 5 meters away from the AP. In this configuration, we calculate the total throughput while varying the number of STAs. The second configuration consists of one AP and 6 STAs: 5 STAs are placed 5 meters away from the AP and the 6^{th} STA is moving

[2]More information about our simulation package can be found in Chapter 5 and 6.

away from the AP from 1 meter to 25 meters. Every meter, it stops moving for 20 minutes and transmits CBR data over a UDP connection. The fixed STAs send continuously the same traffic as the mobile STA to the AP. We calculate the throughput of both moving and fixed STAs for each position of the mobile node. Further, all simulations are done with two transmission modes (i.e., 1 Mbps and 2 Mbps in IEEE 802.11b).

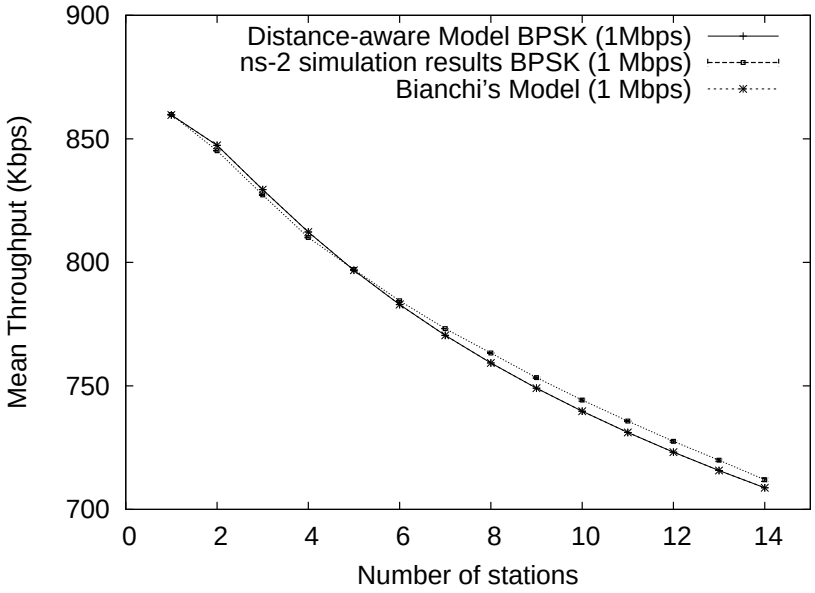


(a) Fixed (b) Moving

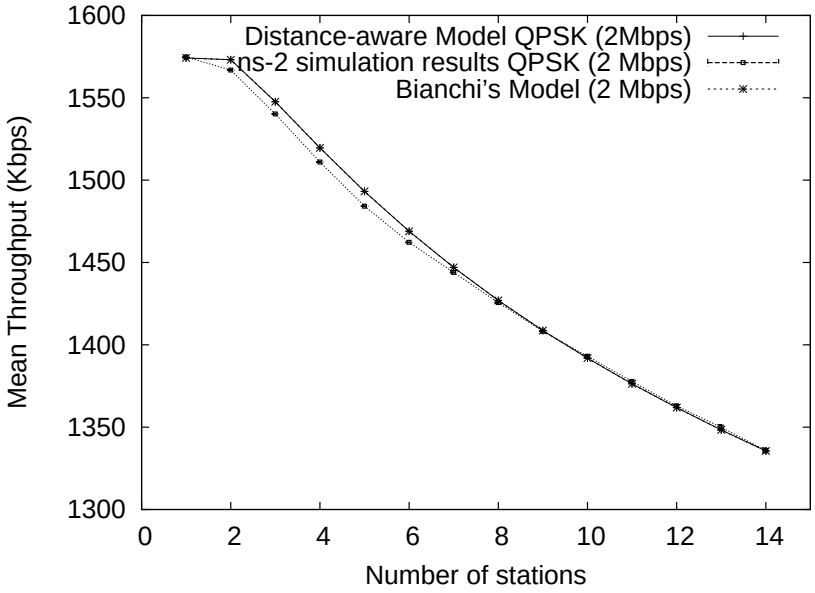
Figure 4.2: The fixed and moving network topologies.

For the first configuration, we compare the throughput obtained with the *DAW* model to the ones computed from Bianchi's model and NS-2 simulations. We expect to obtain very similar results since the probability that a packet is erroneous, and hence dropped, is very close to one when parallel transmissions occur (all STAs are positioned at the same distance from the AP). The corresponding total throughput of our distance-aware model and Bianchi's model along with NS-2 simulations are shown in Figure 4.3. We can observe a very close match between our model and NS-2 simulation results.

We now use the *DAW* model to investigate the throughput of the STA that moves through a fixed topology of wireless STAs using the second configuration explained above. Figure 4.4 shows the throughput of a fixed STA (at 5 meters from the AP) and the throughput of the moving STA. We also plot in the same figure the throughput obtained by an STA if Bianchi's model was used. The results are very interesting. When the moving STA is close to the AP (less that 5 meters), its throughput is greater than that of the fixed stations. When the moving station and fixed stations are close to each other (i.e., around 5 meters) they all have the same throughput and it is equal to the one given by Bianchi's model (which supposes that two colliding packets are automatically lost). Finally, when the moving STA is far from the AP (more than 5 meters), its receiving power level at the AP starts to become lower than that of



(a) 1 Mbps



(b) 2 Mbps

Figure 4.3: Simulation and Model comparison for fixed scenarios for (a) 1 Mbps and (b) 2 Mbps.

the close and fixed STAs and so its packets are lost when they collide with the ones from fixed STAs. The fixed STAs then get a higher throughput than the moving STA and the difference is approximately equal to the bandwidth not used by the close and fixed STAs. The Bianchi's model is no longer good in such a moving case as shown in Figure 4.4. The result is also confirmed with NS-2 simulations as shown in the Figures.

To better illustrate the above results which are obtained for throughput of fixed and moving STA, we evaluate and calculate two conditional probabilities, shown in Figure 4.5:

1. the probability that the moving STA loses its packet when it contends for the medium with one or more fixed STAs ($P_{pkt-loss-mov}$).

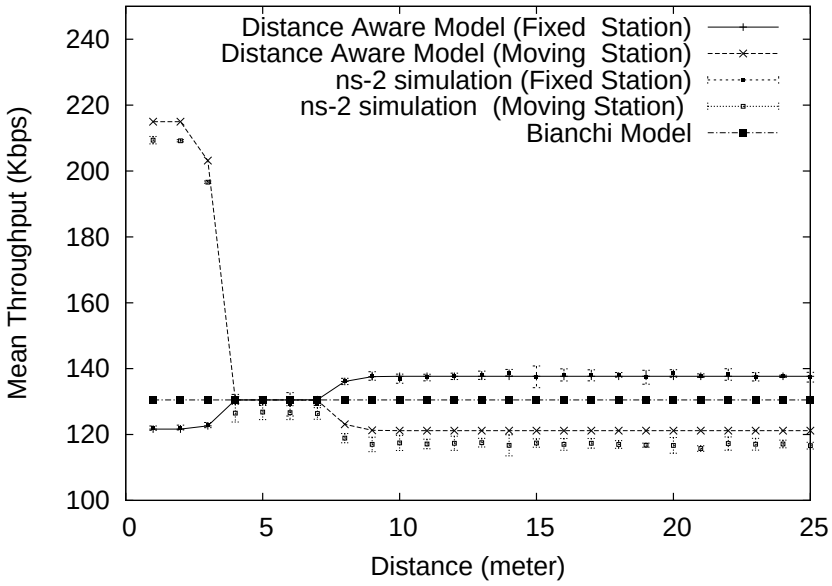
2. the probability that one fixed STA loses its packet when it contends for the medium with the moving STA ($P_{pkt-loss-fix}$).

These two conditional probabilities illustrate the behavior of the *DAW* model in the second configuration where one node moves. As an example, we calculate the second probability ($P_{pkt-loss-fix}$) for the case where data is transmitted with 1 Mbps. Let τ_{fix} (τ_{mv}) be the transmission probability of a fixed (moving) STA and d_{fix} (d_{mv}) be its distance to the AP. Equation (4.29) shows this probability ($P_{pkt-loss-fix}$), where $bino(i,5,\tau_{fix})$ denotes the pdf of a binomial random variable with parameters 5 and τ_{fix} that is shown in Equation (4.30). The sum in Equation (4.29) accounts for the different possible values of the number of fixed stations which are transmitting at the same time as the moving station. Other probability calculations ($P_{pkt-loss-mov}$ and for different transmission rates) are straightforward.

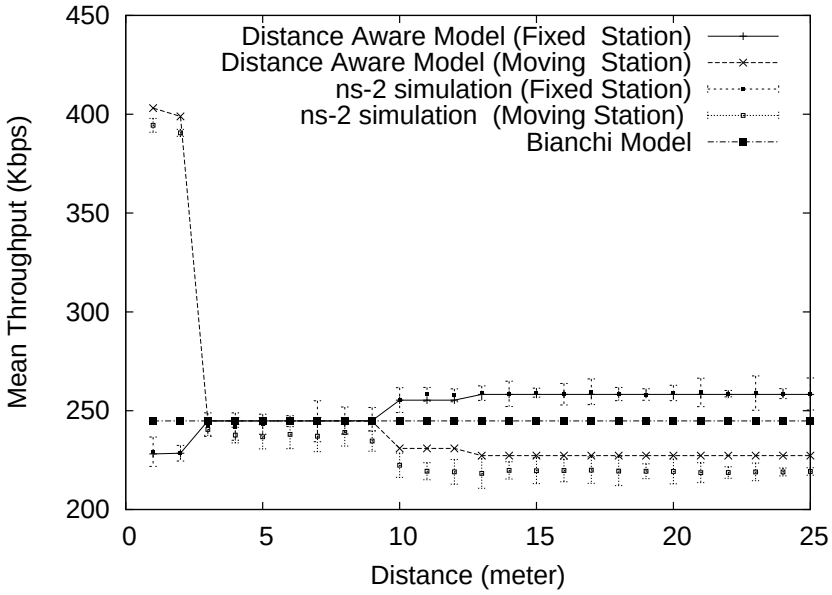
$$P_{pkt-loss-fix} = 1 - \frac{\sum_{i=1}^{5} bino(i,5,\tau_{fix}) \left(1 - Q\left(\sqrt{\frac{2L(d_{mv})W}{(N_0+i\cdot L(d_{fix}))R}}\right)\right)^{L_{PLCP}+L_{Payload}}}{1-(1-\tau_{fix})^5} \tag{4.29}$$

$$bino(i,5,\tau_{fix}) = \sum_{x=0}^{5} \binom{5}{i} \tau_{fix}^{i} (1-\tau_{fix})^{5-i} \delta_x(i). \tag{4.30}$$

As shown in Figure 4.5, both probabilities ($P_{pkt-loss-mov}$ and $P_{pkt-loss-fix}$) are equal to 1 when all STAs have equal distance from AP (same as assumed by the Bianchi's model). $P_{pkt-loss-mov}$ remains equal to 1 when the moving STA is far from the AP since its power level at the AP is low comparing to near and fixed stations. On the other hand, $P_{pkt-loss-fix}$ drops to 0 at around 5 meters, which means that closed and fixed STAs always win when their packets collide with the packets from the moving STA at more than 5 meters from the AP. On the other hand $P_{pkt-loss-mov}$ drops to 0, when the moving station is near the AP (less than 5 meters) and so wins when its packets collide with packets from fixed stations.

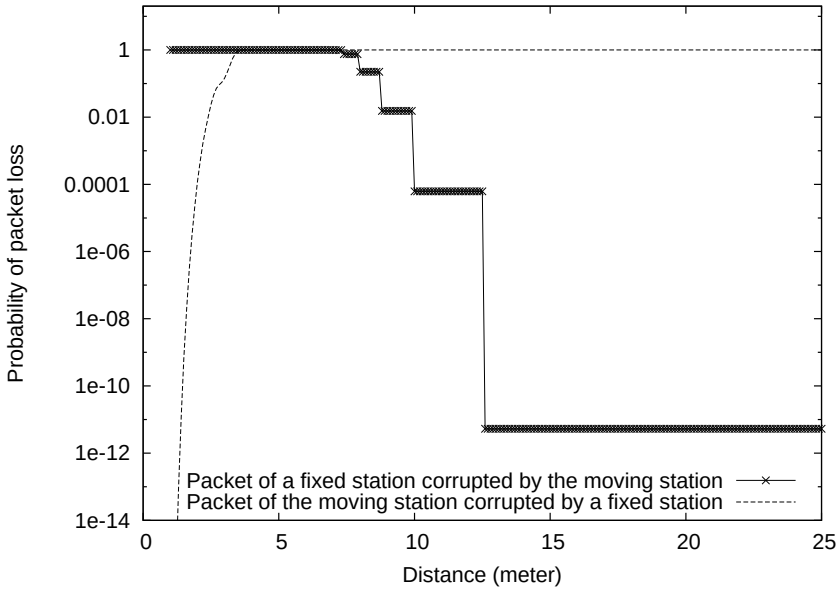


(a) 1 Mbps

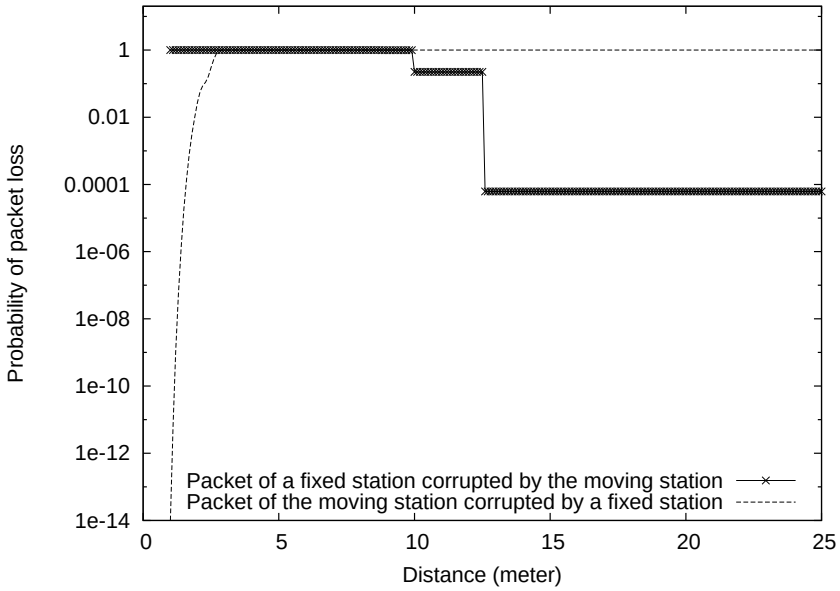


(b) 2 Mbps

Figure 4.4: Simulation and Model comparison for moving scenarios, (a) 1 Mbps and (b) 2 Mbps.



(a) 1 Mbps



(b) 2 Mbps

Figure 4.5: Packet corruption probability versus distance of the moving station, (a) 1 Mbps and (b) 2 Mbps.

We now consider the random topology case, where STAs are uniformly distributed in a disk of radius 10 meters centered at the AP. We select one STA and move it from 1 to 10 meters, and we compute its throughput averaged over all the possible locations of the other 9 STAs using the method explained in Section 4.4.2. We also compute the average throughput of any other fixed STA. To validate these results, we have run 250 numerical simulations (with the *DAW* model) for 250 realizations of the fixed topology using the above scenario. We then use the fixed topology method to find the throughput per realization, and we average it over all the realizations.

The results are shown in Figure 4.6. When the moving STA is close to the AP, it gets a higher throughput than the average throughput of the others, since with a high probability the other STAs are far from the AP. However, this throughput decreases when the STA moves farther from the AP until it drops below the average throughput of the others. The results for 250 realizations, shown in Figure 4.6, validate our analysis in the random case (Section 4.4.2) as well.

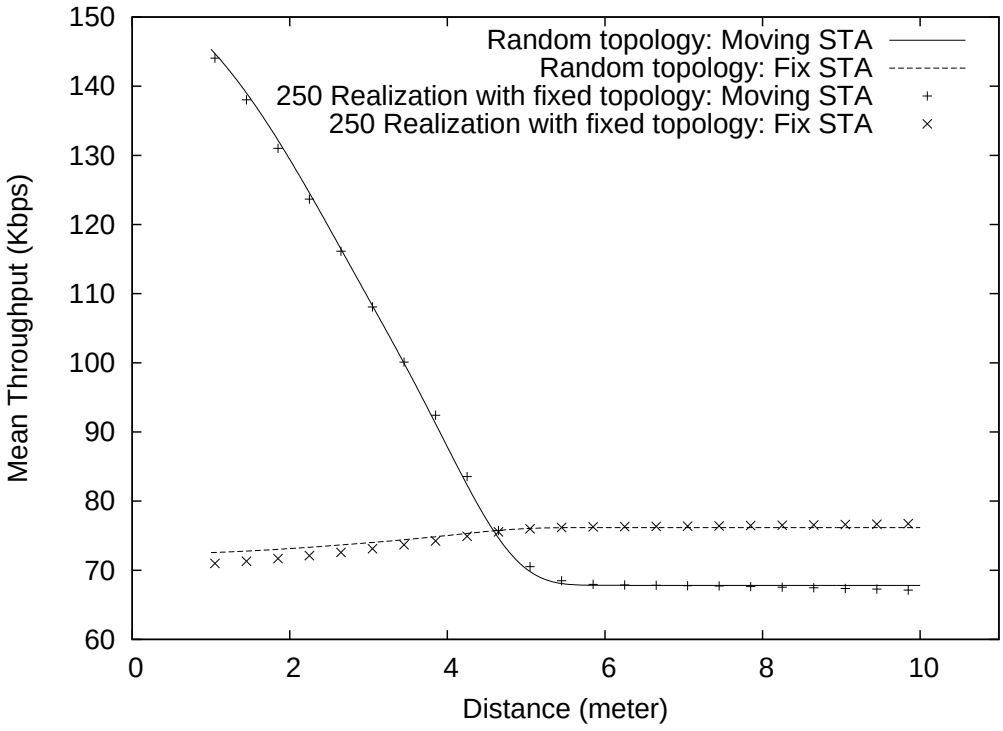


Figure 4.6: Throughput calculation for random topology, BPSK(1 Mbps).

4.7 Conclusions

In this chapter we presented an analytical model that accounts for the positions of stations when evaluating the performance of the 802.11 MAC layer. Our model achieves more realistic results comparing to the models which do not consider PHY channel conditions (in particular, in the scenarios where the stations move). We also validated our model with NS-2 simulation. This model has different extensions on which we are working on. In one extension we consider an active AP which send packets for several users. In such case we will try to find the optimal AP placement for a given topology using our DAW model. Further extensions could consider fading channel models as well. An ad-hoc mode evaluation to numerically approach the wireless capacity found in [32] could be one of its extensions.

5

IEEE 802.11 PHYSICAL RATE ADAPTATION

5.1 Summary

To achieve high performance under varying conditions, IEEE 802.11a/b/g WLAN devices need to adapt their transmission rate dynamically. A lot of research and experimental work has been conducted in this field, since the rate adaptation mechanism is a critical component of WLAN performance. This chapter first presents the most important parameters that should be taken into account to design an efficient rate adaptation mechanism. Specifically, we emphasize on wireless channel characteristics, hardware configurations, contrast between low and high latency systems, power control, frequency channel allocation, and fairness issue in IEEE 802.11 WLANs. We then present a detailed survey on existing rate adaptation mechanisms along with their advantages and drawbacks. We then describe several propositions to improve the performance of rate adaptation mechanisms. We present two new mechanisms named AARF and CLARA. CLARA is a culmination of the best attributes of the ARF and RBAR mechanisms with additional practical features to facilitate multipath fading channel sensing and feedback control signalling. AARF is proposed for low latency systems and has low complexity while obtaining a performance similar to RBAR in stationary and non-fading wireless channels. Finally, we present a new rate adaptation algorithm designed for high latency systems that has been implemented and evaluated on an AR5212-based device. Experimentation results show a clear performance improvement over the algorithm previously implemented in the AR5212 Madwifi driver.

5.2 Introduction

As explained in Chapter 2, the IEEE 802.11a/b/g WLAN working groups provide a multi-rate physical layer capability [1]. This is obtained by employing different sets of modulation and channel coding. The standard defines minimum requirements for physical and MAC layer functionalities. However, the exact transceiver architecture and the rate adaptation mechanisms are left open to WLAN device manufacturers.

Recent studies on rate adaptation mechanisms at the MAC layer have focused on dynamic rate switching and frame length selection by either sender inference of wireless channel [43, 28], receiver's feedback from wireless channel condition [44, 45, 46, 47, 48], power control [49], or by increasing the time fairness between mobile STAs [46, 50]. This chapter presents a detailed survey of rate adaptation mechanisms and the most important parameters that should be taken into account to propose an efficient rate selection algorithm.

Basically, according to the wireless channel conditions, transmission can take place at a number of rates. High data rates are possible when the channel condition is good. Practically, the channel condition can vary significantly for mobile or stationary wireless nodes. There are many reasons for the highly volatile nature of the wireless medium used by the IEEE 802.11 standard: fading, attenuation, interference from other radiation sources, interference from other 802.11 WLANs or Bluetooth, etc. These quality variations can be categorized to *transient short* and *durable long* term modifications.

For instance, if someone walks around, closes a door, or moves big objects, this will modify the channel characteristics for a very short time. Its throughput capacity might drop sharply but not for long. If one decides to move to another office, thus approaching the access point, the attenuation will decrease and this will have a longer lasting effect on the energy of the radio signal that will decrease the bit error rate. This, as explained above, will allow higher application-level throughput since the packet error rate is lower. There are some scenarios in that signal to noise ratio and distance between source and destination have little predictive value for loss rate [51]. In such conditions, the loss rates are probably due to multi-path fading rather than attenuation or interference transition. The performance of wireless card devices depends on their structures as well.

This chapter addresses the wireless channel conditions and the wireless card interface structures which play an important role in rate adaptation mechanisms. In addition, the implementation issues in mobile STAs are addressed here. Because we are not aware of any paper that discusses the issues surrounding real implementations of 802.11 rate adaptation algorithms, one of our contributions in this chapter is the identification of two classes of 802.11 devices, namely *low latency* and *high latency* systems. *Time fairness*, *power control*, and *frequency allocations* are the parameters that have been considered in most of the proposed rate adaptation

mechanisms. This chapter discusses these issues in detail as well.

Finally, we present our solutions which include three mechanisms called *Adaptive Auto Rate Fallback* (AARF), *Adaptive Multi Rate Retry* (AMRR), and *Closed-Loop Adaptive Rate Allocation* (CLARA). AARF and AMRR are two simple novel algorithms each designed for one of the two classes of devices identified. Their performances are close to the optimum represented by the impractical RBAR [44] in the case of infrastructure networks and AWGN channel (non-fading channel). We present experimental results showing that our new algorithms can be readily implemented on existing hardware and offer considerable performance improvements over current solutions. The third proposed mechanism, CLARA, is a culmination of the best attributes of both the ARF and RBAR mechanisms while removing their respective weaknesses. The implementation of CLARA can be carried out with very little difficulty. Since CLARA is designed to work transparently within the confines of the standards, it can be implemented among 802.11 compliant devices. Furthermore, due to its transparency, devices using our algorithm can co-exist with other 802.11 compliant devices within the same basic service set.

The chapter is organized as follows. In Section 5.3, we present the key parameters that should be taken into account for rate adaptation mechanisms, i.e., *wireless channel characteristics*, *low and high communication latency systems*, *time and throughput fairness*, *power control*, and *frequency allocation*. Then in Section 5.4, we present a survey of the existing mechanisms that try to address the task of rate adaptation followed by their shortcomings.

We describe in Section 5.5 our solutions for rate adaptation mechanisms. We explain the AARF algorithm designed for low communication latency systems that is based on per-packet short-term adaptation but introduces a specific long-term adaptation mechanism to improve the application level throughput. We also explain the CLARA rate adaptation mechanism which facilitates multipath fading channel sensing and feedback control signalling. In Section 5.6, we evaluate the performance of the two proposed mechanisms using MATLAB and NS-2 simulations. In Section 5.7, we present our practical solution AMRR, based on the same ideas developed for AARF that has been designed, implemented, and evaluated on a high communication latency system based on the Atheros AR5212 chipset. The experimental measurements obtained confirm the simulation results and offer convincing evidence that the AMRR algorithm achieves higher performance than the default implemented algorithm. Finally, we conclude this chapter in Section 5.8.

5.3 Key Parameters for Rate Adaptation Mechanism

Several key aspects need to be considered to propose a rate adaptation mechanism. The aim of this section is to identify and evaluate them.

5.3.1 Wireless Channel Characteristics

In chapter 3, we presented a detailed evaluation of wireless devices considering wireless channel characteristics. The performance of communication systems over wireless channels is captured and analyzed, using the bit error rate as a function of signal to noise and interference ratio and the actual performance of the wireless system depends on several implementation issues and wireless channel characteristics. Let us explain this issue with a simple example which we present the BER calculation for 1 and 2 Mbps transmission rate for IEEE 802.11b WLANs.

According to the standard specification, the transmit signal is differentially encoded and BSPK/QPSK modulated for these transmission modes. The receiver can detect the signal coherently (DE-BPSK and DE-QPSK) or differentially (DBPSK and DQPSK), see Section 3.6 for a more detailed evaluation. The probability of bit error for these modulations in an AWGN channel and a Rayleigh channel can be calculated by Equation (3.29) and Equation (3.30) respectively.

Figure 5.1 shows the results of probability of bit error for these transmission rates using one or two paths (antennas) at the receiver side. We have also plotted the BER for AWGN channel ($L = \infty$). As we discussed in Chapter 3, the results show that for a target BER (e.g., 10^{-5}), it is difficult to select the suitable SNR for the transmission rates, without taking into account the number of receiver antennas, the type of diversity combining, the receiver structure, and the severity of multipath fading. This selection will be more difficult when we consider other PHY layer implementation issues like demodulator/detector, binary convolutional encoders, and interleavers in IEEE 802.11a/g presented in Chapter 3.

The above example shows that an efficient rate adaptation algorithm should take into account the receiver structure (e.g., number of antennas at the receiver) and the wireless channel conditions. For example, in public areas like airports the wireless channel suffers from a deep fading caused by moving objects while in a small and calm place (e.g., work offices) the channel suffers from a very short time modifications. In summary, the rate adaptation mechanisms should behave in different manners for different devices and wireless conditions. To the best of our knowledge, the existing rate adaptation mechanisms have not taken into account all the above issues.

5.3.2 Low and High Communication Latency Systems

Typically, all the 802.11 systems contain at least the following two subsystems, the *802.11 PHY layer or radio* and the *802.11 MAC layer*. The 802.11 radio integrates the modulation, demodulation, encoding, decoding, ADC, DAC, and filtering functions. These functions are always entirely implemented in hardware. Usually, the MAC layer is implemented by a combination of dedicated hardware and dedicated software. The exact split between these two domains is

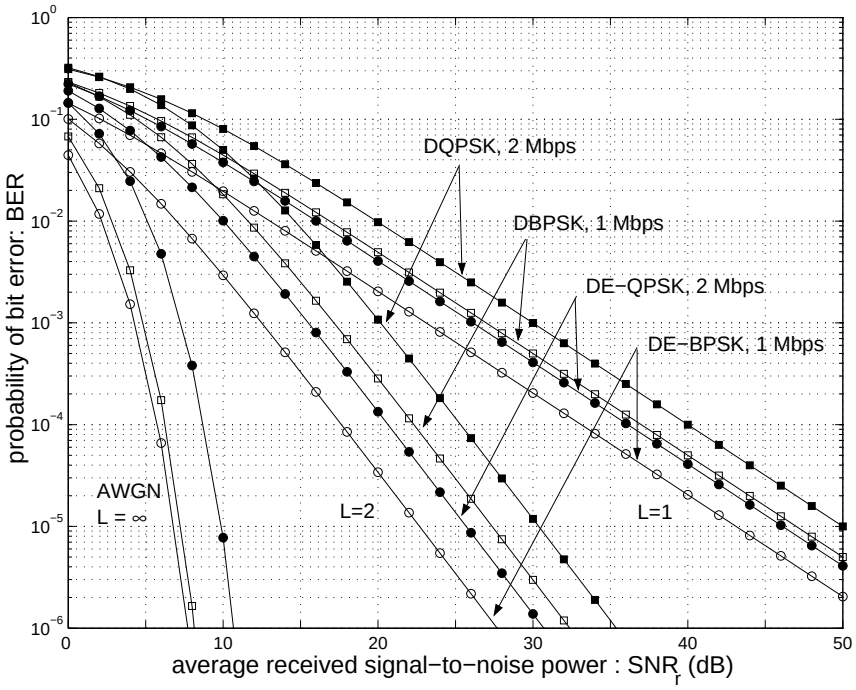


Figure 5.1: BER as a function of SNR for 1 Mbps and 2 Mbps transmission rates in 802.11b.

entirely device specific.

The rate adaptation mechanisms we are interested in are part of the MAC layer. Their function is to choose the rate to be used for each packet that is sent to the PHY layer. The exact architecture of the MAC layer, i.e., how much of the MAC layer is implemented in hardware, varies a lot from one device to another. Therefore, it is very hard to design a device-independent rate adaptation mechanism. However, it is clear that the *communication latency* between the PHY layer and the block that implements the rate adaptation mechanism within the MAC layer is one of the most important parameters to take into account when designing and implementing the mechanism.

Low latency systems allow us to implement per-packet adaptation. This means that for each packet to be sent, feedback information on the transmission status of the previous packet is required before sending. As such, we calculate below the minimum time interval between two successive packet transmissions during a fragment burst ($T_{fragment}$), when a packet transmission fails ($T_{failure}$) and when a packet transmission succeeds ($T_{success}$). In the case of transmission failure because the required ACK has not been received, the sending device starts a backoff procedure after a DIFS at the end of $ACKTimeout$. The transmission of other packets by this device does not start until the end of the backoff procedure which cannot happen until $T_{failure}$ after the end of the transmission. $T_{failure}$ can be expressed as:

$$T_{failure} = ACKTimeout + DIFS + aCW_{rnd} \times aSlotTime, \tag{5.1}$$

where aCW_{rnd} is a uniformly distributed variable between zero and aCW where $aCW_{min} < aCW < aCW_{max}$. In the case of successful transmission, the sending device either starts a SIFS timer if it wants to keep on bursting the following MAC fragments ($T_{fragment} = SIFS$) or it starts a backoff procedure after a DIFS if it wants to transmit another packet that is not a MAC fragment,

$$T_{success} = DIFS + aCW_{rnd} \times aSlotTime. \tag{5.2}$$

Because we are interested in the worst case scenario (that is, the minimum time interval between two transmissions), we assume that $aCW_{rnd} = 0$ which gives;

$$\begin{aligned} T_{fragment} &= SIFS, \\ T_{success} &= DIFS, \\ T_{failure} &= ACKTimeout + DIFS, \end{aligned} \tag{5.3}$$

and since $SIFS < DIFS$,

$$T_{fragment} < T_{success} < T_{failure}. \tag{5.4}$$

The values of these parameters are shown in Table 5.1. If we do not allow the user to use a different rate for each fragment of a burst, the minimum latency requirement for all the values presented in Table 5.1 column $T_{success}$ is 28µs. And if the user can change the rate for each fragment, (for example, by using the new information from each fragment's ACK) the minimum latency requirement will be 10µs.

Table 5.1: Communication latency constraints in the 802.11 standards

Standard	$T_{fragment}$	$T_{success}$
802.11 DSSS	10 µs	50 µs
802.11a	16 µs	34 µs
802.11b	10 µs	50 µs
802.11g	10 µs	28 µs

In summary, any multi-standard system where the two-way communication latency between the PHY layer (where the transmission status is known) and the rate adaptation mechanism (where the information on the transmission status is acted upon) is higher than 28µs

(10μs) cannot implement per-packet (per-fragment) rate adaptation. The WaveLAN 802.11b Chipset [52] which includes an embedded processor and a dedicated communication bus with the baseband controller, can be categorized as a low latency system. More examples for low and high latency wireless systems can be found in [53]. As a result, several rate adaptation mechanisms can not be implemented in many existing devices, considering the latency requirements of the device. Regarding to this issue, we will present an implementation technique considered the high latency system requirements in this chapter.

5.3.3 Time and Throughput Fairness

While multiple data rates have increased considerably, from 1 Mbps to 54 Mbps, the MAC layer remains practically the same despite all the solutions proposed. In fact, the existing DCF mechanism in the 802.11 standard which employed CSMA/CA, provides an equal opportunity to access the channel in the long term. If all the mobile STAs send the packets with the same frame size and same data rate, DCF provides *throughput fairness* as well. But when some mobile STAs use lower bit rates than others, the performance of all the mobile STAs are considerably degraded [54] since the slow STAs capture the channel for a long time (i.e., no *time fairness*).

Typically, it is desirable to achieve a throughput proportional to the employed transmission rate (i.e., QoS requirements at application level). This can be done if we can provide *time fairness*. Several rate adaptation mechanisms were proposed to provide *time fairness*. Generally, we can decrease the time unfairness by selecting the best transmission rate and give more opportunities to the mobile STAs with higher transmission rates. For example, the OAR [50] and the MAD [46] mechanisms provide time fairness using this technique. Several other schemes, like burst transmission mode, are being standardized as a part of 802.11e MAC as well [55].

Typically, such mechanisms [50, 46] use a simple back-to-back burst transmission mode: the sender is allowed to send several frames back-to-back in a burst provided the entire frame exchange duration does not exceed a threshold value (e.g, proportion of transmission rate over basic rate). The MAD mechanism [46] has proposed a *packet concatenation* (PAC) mechanism as well. Since these mechanisms can be integrated in all rate adaptation mechanisms to provide time fairness, we have not focused on this issue in our solutions.

5.3.4 Power Control versus Rate Adaptation

One of the most important issue in the IEEE 802.11 WLANs is to reduce the power consumption of mobile STAs. *Transmit power control* (TPC) and *rate adaptation mechanisms* are the two most effective ways to achieve the economic power consumption. There are two major benefits of TPC: *Energy Saving* and *Spatial Reuse*. First by reducing transmit and receiving powers on wireless nodes, the mobile STA preserves energy and potentially extends the lifespan of an ad

hoc network. Second, a lower transmit power level leads to a shorter range of interference. As a result, multiple flows of transmission may occur in the vicinity of each other. This increase in spatial reuse could lead to increased capacity of the wireless networks as well.

Some of the rate adaptation mechanisms combine PHY rate adaptation and TPC to minimize the power consumption. These mechanisms can use the new structures provided by the new standard specifications like 802.11h to support intelligent TPC and rate adaptation simultaneously. In the next section, we explain one of the most famous algorithm that combined TPC and PHY rate adaptation named MiSer [49].

In summary, the mechanisms that adapt the transmission parameters to the channel conditions can be designed to optimize power consumption and/or throughput. Because we have focused on the task of maximizing the application-level throughput through rate adaptation mechanisms we do not use any TPC procedure in our proposed solutions.

5.3.5 Frequency allocation versus Rate Adaptation

The IEEE 802.11 standard provides multiple frequency channels beside multiple data rates (see Appendix B). For instance, the IEEE 802.11b provides 11 channels from 2.4 GHz to 2.483 GHz. Among these channels, only three of them are non-overlapping, i.e., in a network with IEEE 802.11b access points there can only be three bases stations that have overlapping coverage areas. Similarly, IEEE 802.11a provides 12 channels with 8 non-overlapping channels. Recently, some research works have focused on frequency allocation in IEEE 802.11 wireless LANs. Particulary in these algorithms, if the channel conditions on a selected frequency channel are not favorable, a mobile STA can skip to a better quality channel to send data with higher data rate. The rate adaptation mechanisms that employ frequency reallocations to provide better performance are out of scope of this paper. We are not aware of many research works in this field in the 802.11 WLAN networks (e.g., [56]). This issue can be considered as a hot open issue for IEEE 802.11 rate adaptation, specially for ad hoc network structures.

5.4 Survey of Rate Adaptation Mechanisms

In this section, we provide a detailed review of rate adaptation mechanisms that have been proposed for IEEE 802.11 WLANs. For each mechanism, we briefly summarize the key ideas and present their advantages and drawbacks considering the key parameters discussed in the previous section. Generally, we can divide rate adaptation mechanisms into two main groups, *non-feedback based* (or sender-based) and *feedback based* (or receiver-based) mechanisms. The mechanisms in the first group try to estimate the channel conditions without feedback from the receiver side [43, 28, 8], while those of the second group employ algorithms to get feedback

from the channel conditions at the receiver side [47, 44, 46, 50].

5.4.1 Auto Rate Fallback: ARF

ARF [43] was the first rate adaptation algorithm to be published. It was designed to optimize the application throughput in WaveLan II devices, which implemented the 802.11 DSSS standard. Rumors claim that ARF has been used in Agere and Intersil Prism II 802.11b devices but it is hard to come up with any meaningful information since this is considered as sensitive intellectual property. In ARF, each sender attempts to use a higher transmission rate after a fixed number of successful transmissions at a given rate and switches back to a lower rate after 1 or 2 consecutive failures. Specifically, the original ARF algorithm decreases the current rate and starts a timer when two consecutive transmissions fail in a row. When either the timer expires or the number of successfully received per-packet acknowledgements reaches 10, the transmission rate is increased to a higher data rate and the timer is reset. When the rate is increased, the first transmission after the rate increase (commonly referred to as the *probing* transmission or *probing* packet) must succeed or the rate is immediately decreased and the timer is restarted rather than trying the higher rate a second time.

Simulations and experimental results show that ARF outperforms other rate selection algorithms when the channel condition is suitable to send with the highest data rate [44, 57]. In fact, ARF finds the best rate (highest rate) very fast and without any feedback from the receiver side (i.e., adding complexity to implement the feedback algorithms). This scheme suffers from two problems:

- If the channel conditions change very quickly, it cannot adapt effectively. For example, in an ad hoc network where the interference bursts are generated by other 802.11 packet transmissions, the optimum rate changes from one packet to the next. Because ARF requires 1 or 2 packet failures to decrease its rate and up to 10 successful packet transmissions to increase it, it will never be synchronized with the sub-packet channel condition changes.

- If the channel conditions are not excellent and do not change at all, or change very slowly, the ARF mechanism will try to use a higher rate every 10 successfully transmitted packets. This results in increased retransmission attempts and thus decreases the application throughput.

5.4.2 Receiver Based Auto Rate: RBAR

Figure 5.2 shows the timeline of the enhanced DCF for the RBAR mechanism. The RBAR mandates the use of the RTS/CTS mechanism: a pair of RTS/CTS control frames are exchanged

between the source and the destination nodes prior to the start of each data transmission. The receiver of the RTS frame calculates the transmission rate to be used by the upcoming data frame transmission based on the SNR of the received RTS frame and on a set of SNR thresholds calculated with an *a priori* wireless channel model. The rate to use is then sent back to the source in the CTS packet. The RTS, CTS, and data frames are modified to contain information on the size and rate of the data transmission to allow all the nodes within transmission range, at both receiver and sender's side, to correctly update their *network allocation vector* (NAV).

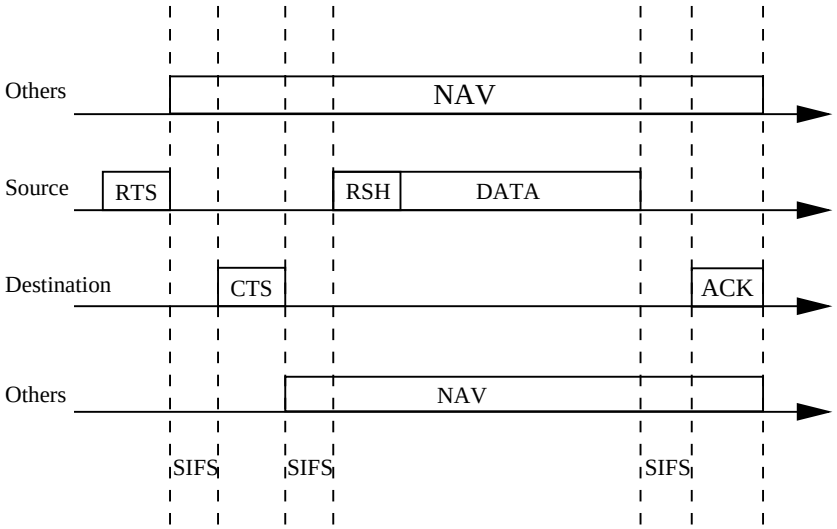


Figure 5.2: Timeline for enhanced DCF supporting RBAR mechanism.

While RBAR is of little practical interest because it cannot be deployed in existing 802.11 networks, it is of important theoretical interest because it can be used as a performance reference. However, the RBAR suffers from numerous flaws that are summarized below:

- This mechanism requires incompatible changes to the IEEE 802.11 standard: the interpretation of some MAC control frames is changed and each data frame must include a new MAC header field (RSH in Figure 5.2).
- The threshold mechanism used in each receiver to pick the best possible rate requires a calculation of the SNR thresholds based on an *a priori* channel model. RBAR has not considered the wireless channel parameters that are discussed in Section 5.3.1.
- The mechanism assumes that the SNR of a given packet is available at the receiver, which is not generally true: some (but not all) 802.11 devices provide an estimation of the SNR by measuring the energy level prior to the start of the reception of a packet and during the reception of the packet.

- The RTS/CTS protocol is required even though no hidden nodes are present. This can be a major performance problem.
- The interpretation of the RTS and CTS frames and the format of the data frames is not compatible with the 802.11 standard. Thus, RBAR cannot be deployed in existing 802.11 networks.

5.4.3 Full Auto Rate: FAR or Enhanced-RBAR

The FAR mechanism is proposed based on two key ideas [45]: adapting the transmission rate of all control and data frames and combining the sender and receiver based algorithms. First, FAR adapts the PHY transmission rate for RTS/CTS/ACK packets while other mechanisms suppose that all control packets should be sent with basic rate. Second in FAR, the rate adaptation of RTS/ACK packets is done at the sender while that of DATA/CTS packets is done at the receiver.

In order to solve the above issues and avoid changing the frame format as it is proposed in RBAR, the authors have proposed a new DCF timeline. Figure 5.3 shows the timeline for enhanced DCF called *modified virtual carrier sensing* (MVCS). With MVCS, the NAV is updated in 3 phases. Basically, the MVCS protocol reserves the medium only for the immediate next frame. For example, in Figure 5.3 after sending RTS by the source, the nodes in the range of the sender will defer their transmission by a duration to allow the CTS to pass. During the data transmission these nodes allow to pass data and ACK packets as well. Since DIFS is always bigger than SIFS, these nodes can never interrupt data transmission. For the nodes in the range of the destination, the NAV should be updated at the end of CTS transmission. MVCS forms the basis to propose a simple and efficient rate adaptation mechanism, i.e., FAR. Clearly MVCS reduces the overheads as well as the complexity of RBAR, which uses the *reservation subheader*.

On the other hand, FAR takes into account another characteristic of the 802.11 standard. As we discussed in Section 2.4, if a mobile STA detects an erroneous frame it will defer by an EIFS value. EIFS is equal to $\mathrm{SIFS} + \mathrm{ACK}_{\mathrm{time}} + \mathrm{DIFS}$ [1]. Considering EIFS time which is bigger than RTS/CTS/ACK transmission time with the lowest transmission rate (please see Table 2.1), mobile STAs will be able to send the control packets even with the highest rate.

FAR selects the rate for RTS, CTS, DATA, and ACK using a cached rate at the sender, cached rate or basic rate at the receiver, piggyback rate from CTS at the receiver, and piggyback rate from DATA at the sender respectively. Note that the duration fields in RTS, CTS, DATA, and ACK packets are data length in bytes, data length with chosen rate, ACK length with chosen rate, and 0 respectively. These duration fields will be used to update the NAV as discussed before for MVCS protocol. Analytical and simulation results show that the FAR mechanism significantly outperforms RBAR [45], however it suffers from three important flaws:

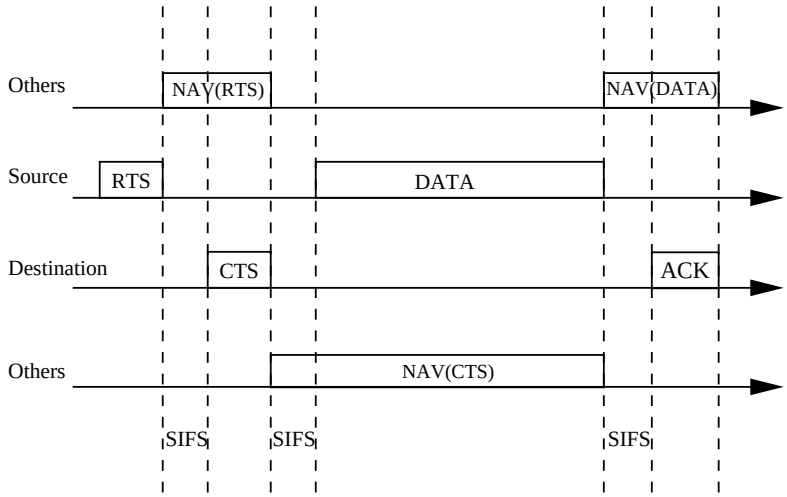


Figure 5.3: Timeline for enhanced DCF supporting FAR mechanism.

- Like RBAR, the RTS/CTS protocol is required even though no hidden nodes are present.
- The mechanism required to update the NAV should be modified for all the mobile STAs in the network. Thus FAR can not be deployed easily in existing 802.11 WLANs. In other words, the 802.11 compliant devices can not co-exist with FAR devices without employing MVCS.
- The authors do not explain the functionalities of MVCS when the sender uses the fragmentation option. Further modifications are necessary for the fragmentation option.

5.4.4 Opportunistic Auto Rate: OAR

The OAR protocol is an enhancement to any automatic rate adaptation mechanisms to provide *time fairness* [50] (see Section 5.3.3). This mechanism tries to better exploit durations of high-quality channels. The key idea of the OAR protocol is to send multiple back-to-back data packets whenever the channel quality is good. The timeline of this mechanism is shown in Figure 5.4 for a network where the quality of channel for node1 is better than node2 (2 Mbps versus 1 Mbps). This figure compares the timeline of OAR with RBAR as well.

In this example OAR gives two transmission opportunities to node1 to send two packets at 2 Mbps. This provides *time fairness* between node1 and node2. Note that the maximum granted time to a sender is bounded by the transmission time at the basic rate, e.g., 6 Mbps for 802.11a. For instance, if the sender can send with 36 Mbps, it can send up to $[36/6] = 6$ packets, whenever it has channel access. Using the OAR mechanism, all the nodes ensure that they are granted channel access for the same time-shares as single-rate. This protocol achieves

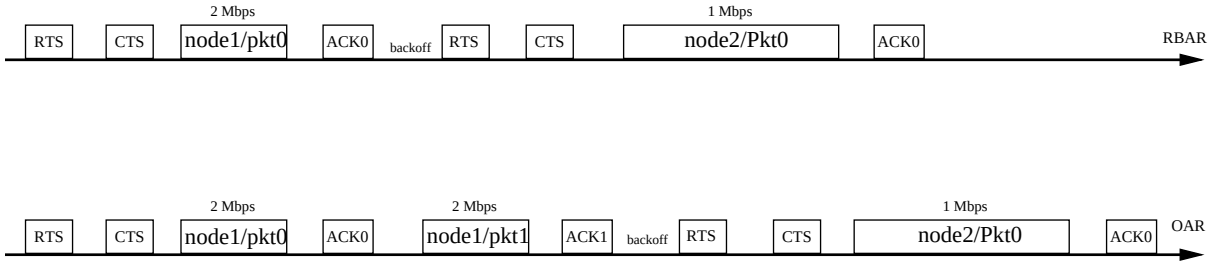


Figure 5.4: Timeline for enhanced DCF supporting OAR mechanism. node1 is in a better channel condition (2 Mbps) than node2 (1 Mbps).

significant throughput gains compared to RBAR (if it is implemented over RBAR), but it suffers from several flaws:

- The mechanism is addressed and evaluated over the RBAR mechanism. Thus it suffers from all the problems mentioned for RBAR.
- It is possible that channel conditions significantly vary during a burst transmission. This causes additional channel probing and RSH overhead to inform others about changing the rates during burst transmission. It is not standard compliant as well.
- The OAR needs some modifications in the fragmentation protocol of IEEE 802.11 standard to integrate in the existing devices.

5.4.5 Medium Access Diversity: MAD

The key idea of the MAD mechanism [46], which is an extension of RBAR and OAR, is to obtain instantaneous physical channel conditions information from several receivers and selectively send the packet to the mobile STA which improves the total throughput of the network. Figure 5.5 shows the timeline of the MAD mechanism for k mobile STAs. In MAD, the sender sends a packet called GRTS (*group RTS*) to all potential receivers. The difference between GRTS and normal RTS is that a GRTS control packet contains a list of receiver addresses pulled by sender. This allows the sender to query multiple receivers about their channel conditions at the same time. Using the rank of the receiver addresses in GRTS packet, the receivers can make the synchronization for sending back their CTS packets as shown in Figure 5.5.

Similarly to RBAR, the receivers select the best transmission mode using a simple threshold rate selection algorithm and send it to the sender by CTS packets. The MAD mechanism uses an estimation of the average SNR via the following equation as well:

$$\overline{SNR_i} = (1-\alpha)\overline{SNR_i} + \alpha SNR_i, \tag{5.5}$$

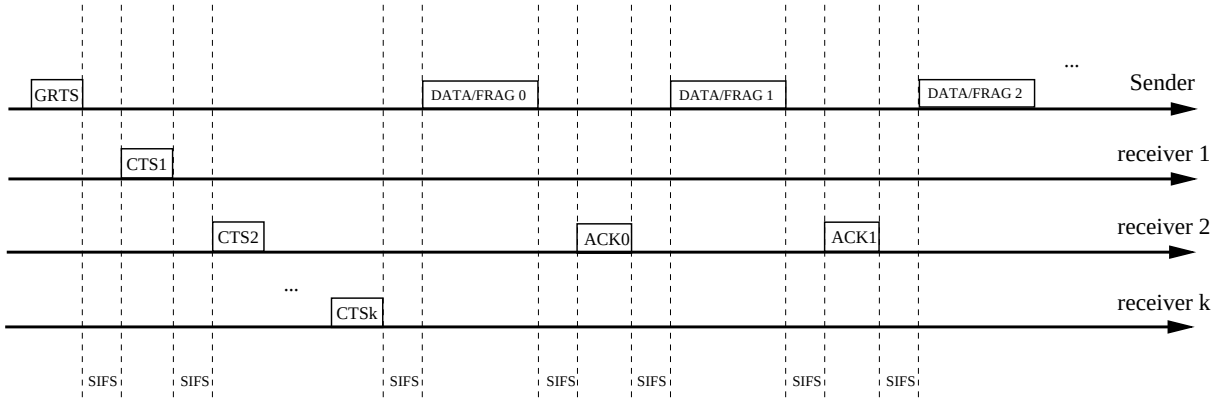


Figure 5.5: Timeline for enhanced DCF supporting MAD mechanism.

where $\overline{SNR_i}$ is the average SNR for STA i, SNR_i is the measured SNR for the last GRTS packet and α is a constant (0.2 in [46]). This helps the receiver to make a more precise estimation of the channel conditions since the average SNR can be updated even though the mobile STA is not in the list of the receivers. MAD completes the channel probing after receiving the last CTS from the last receiver in the list. Then the sender selects one of the receivers based on its scheduling scheme (e.g., receiver 2 in Figure 5.5) and uses the OAR algorithm to send back-to-back data packet. The scheduler aims to improve the utilization of the channel while maintaining temporal fairness among multiple back-logged flows. More information about scheduler and optimum number of receivers can be found in [46].

Contrary to OAR, where the maximum number of packets transmitted back-to-back can not exceed the ratio between current rate and basic rate, the sender can send data packets until a packet is lost or it uses up its time share. In the MAD mechanism, the sender can send a *packet concatenation* (PAC) as well: it transmits a superframe (i.e., control frame), immediately followed by a chain of data packet frames to decrease the overhead of PLCP and MAC header. The analytical and simulation results presented in [46] show that MAD can improve the overall throughput of network by 50% compared to OAR. However MAD suffers from the following flaws:

- It inherits all the drawbacks of RBAR and OAR rate adaptation mechanisms.

- The definition of the new RTS packet type (GRTS) as well as the new frame format for CTS and DATA packets are not standard compliant. Thus, MAD can not be implemented in current WLAN devices.

5.4.6 Link Adaptation: LA

The *link adaptation* (LA) mechanism [28], is the first rate adaptation mechanism that uses the *received signal strength* (RSS) measured from the AP frames to find the best transmission rate. Figure 5.6 shows the structure of the LA mechanism.

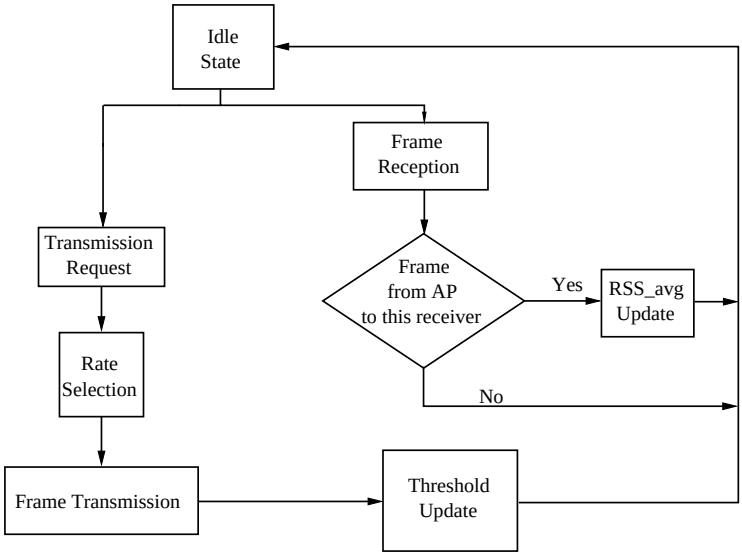


Figure 5.6: Structure of link adaptation algorithm.

With LA the mobile STAs use the measured RSS of all packets sent by AP that are addressed to itself or broadcast/multicast addressed frame from AP. LA mechanisms computes the RSS_{avg} by:

$$RSS_{avg} = a \cdot RSS_{avg} + b \cdot RSS \tag{5.6}$$

where $a \geq 0$, $b \geq 0$, and $a + b = 1$. The mobile STA also uses several thresholds to select the best transmission rate. These thresholds are function of the frame length[1] and should be updated by:

$$Th[i,j] = c \cdot Th[i,j] + d \cdot RSS \tag{5.7}$$

where i and j represent the transmission mode and length interval respectively, $c \geq 0$, $d \geq 0$, and $c + d = 1$.

For instance, if a frame is dropped at a particular transmission rate, the threshold for this rate should increase using the RSS values stored in the station. The authors claim that for

[1]LA mechanism classifies the frames within three intervals: 1-100 bytes, 100-1000 bytes, and 1000-2400 bytes

the rate selection algorithm, a STA will consider the values of RSS_{avg}, thresholds, frame size and number of retransmission attempts. However the details of the LA mechanism are not released publicly. They have only provided some simulation results, thus it is not clear how their algorithm behave in practice. For this reason it is almost impossible to evaluate the performance of the LA mechanism.

5.4.7 Hybrid Rate Control: HRC

Hybrid rate controller (HRC) [47, 48] is the first rate adaptation mechanism which makes a differentiation between *short-term* and *long-term* modification of the channel conditions (see Section 5.3.1). The main goal of this algorithm is to meet the strict latency requirements of streaming applications. The authors use a simple statistic based algorithm along with the information from signal strength indication of the acknowledge (SSIA) to select the best rate.

Figure 5.7 shows the structure of the HRC mechanism. Basically, the core of HRC is a throughput-based rate controller. This controller is the feedback loop at the bottom of Figure 5.7, which probes adjacent rates to determine if a rate switch is necessary or not. In this mechanism, implemented on the AR5000 chipset by Atheros, 10% of the data is sent at next higher or lower available data rate. At the end of a specified decision window which is about one second, the performances of all three data rates are determined and the mobile STA switches to the rate with the highest throughput.

The second loop (i.e., SSI-rate lookup table at the top of the Figure 5.7) bounds the acceptable range of the SSIA frames values for each rate selected in the core loop. These bounds can be implemented as a simple lookup table.

In addition, based on the information received from the rapid SSI change detector module, the lookup table uses two different sets of thresholds, named stable and volatile low thresholds. When the channel conditions change rapidly, the table lookup uses the volatile thresholds. The adaptive adjustment logic module is responsible for the update of the values in the lookup table according to the recent history of channel conditions. Further information about functionality and the implementation issues of this algorithm can be found in [48].

The authors presented the results of an experimental real-time video streaming with two moving laptops equipped with 802.11a wireless network interface and showed that HRC effectively prevented many packet losses, achieving a much higher quality than static-based algorithms. The authors conducted a series of measurements to compare the signal behavior at the sender and at the receiver. The results show that the SSI at the receiver and at the sender are strongly correlated, e.g., for 74% of the packets, the difference between SSI at the sender and at the receiver falls within the stationary ± 2 dB noise range. Note that we will use this result in one of our solution, i.e., CLARA, there we use ACK SSI to get feedback for channel condition at receiver [7]. The performance of this mechanism has only been compared with the algorithm

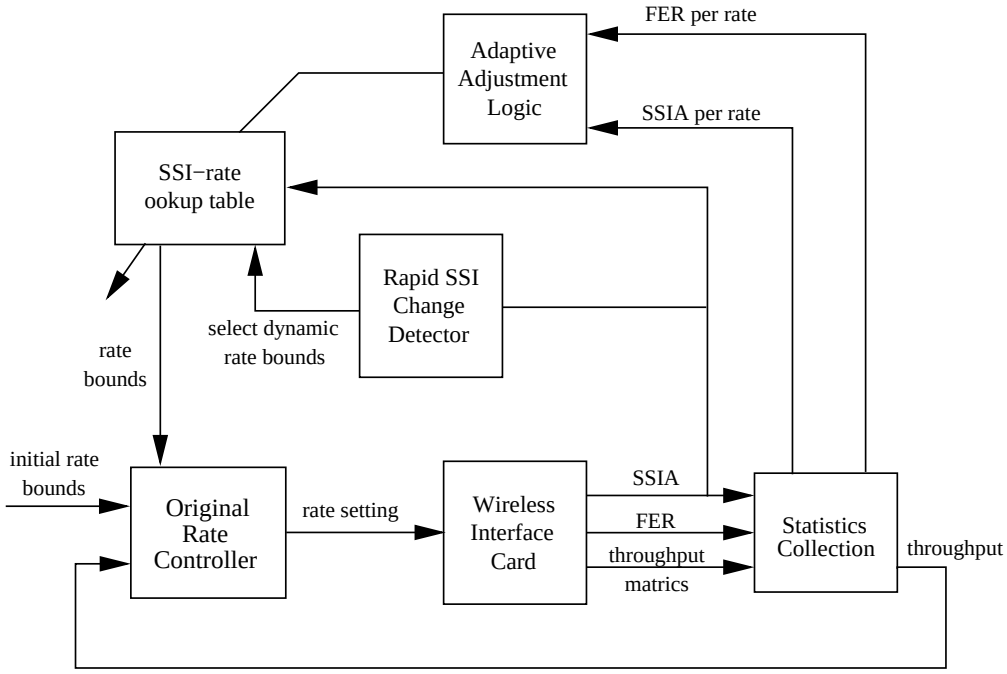


Figure 5.7: Structure of hybrid rate controller.

implemented for the R5000 chipset by Atheros.

5.4.8 Minimum-energy Transmission Strategy: MiSer

As explained in Section 5.3.4, MiSer [49] is an algorithm based on the 802.11a and 802.11h[2] standards whose goal is to optimize the local power consumption (and not the application throughput which is our stated goal). To achieve this, it adapts both the transmission rate and the transmission power. The set of optimal rate/transmission power pairs is calculated offline with a specific wireless channel model. At runtime, the wireless nodes execute simple table lookups to pick the optimum rate/transmission power combination. The main problems of this algorithm (other than mandating the use of the RTS/CTS protocol) are twofold:

- It requires the choice of an *a priori* wireless channel model for the offline table calculation. Thus it is not a practical approach considering the implementation issues addressed in Section 5.3.1.
- It requires *a priori* knowledge of the number of contending stations on the wireless network.

[2]802.11h is an extension of the current 802.11 MAC and the high-speed 802.11a PHY, to implement an intelligent transmission power control.

5.4.9 MADWIFI

In this section we explain one of the implemented rate adaptation algorithm to which we had access (i.e., MADWIFI algorithm for AR5212 chipset named multi rate retry). A complete Linux driver for the AR 5212 chipset is available from the *multiband atheros driver* for WiFi (MADWIFI) project, hosted on SourceForge [8]. This project contains a binary-only *hardware abstraction layer* (HAL) which hides most of the device-specific registers, a 802.11 MAC implementation imported from the BSD kernel and a Linux AR 5212 driver, heavily inspired by a BSD AR 5212 kernel driver.

The HAL exports a very classic interface to the AR 5212 chipset. It allows the user to create up to 9 unbounded FIFOs (First In First Out queue) of transmission descriptors to schedule packets for transmission. Each descriptor contains a status field that holds the transmission status of the descriptor, a pointer, and the size of the data to be transferred. Each transmission descriptor also contains an ordered set of 4 pairs of rate and transmission count fields (r_0/c_0, r_1/c_1, r_2/c_2, r_3/c_3).

To schedule the transmission of a data buffer, the software driver inserts in one of the FIFOs a properly initialized transmission descriptor. Whenever the wireless medium is available for transmission, the hardware triggers the transmission of the descriptor located at the head of the FIFO. To do so, it transfers the descriptor and the data pointed to by the descriptor from the system RAM to its local RAM and then starts the transmission of the data with the rate r_0 specified in the descriptor. If this transmission fails, the hardware keeps on trying to send the data with the rate r_0, $c_0 - 1$ times. If the transmission keeps on failing, the hardware tries the rate r_1, c_1 times then the rate r_2, c_2 times and finally the rate r_3, c_3 times. When the transmission has failed $c_0 + c_1 + c_2 + c_3$ times, the transmission is abandoned: the status field of the descriptor is updated and it is transferred back from the local RAM to the system RAM. This process is summarized in Figure 5.8.

When the transmission is finally completed, or finally abandoned, the hardware also reports in the transmission descriptor the number of missed ACKs for the transmission of this descriptor. It is interesting to note that this number indirectly indicates the final transmission rate of the packet as well as the transmission rate of each retry.

For example, if $c_0 = 1$, $c_1 = 1$, $c_2 = 1$, and $c_3 = 1$, and if the number of missed ACKs is zero, it means that the transmission completed successfully at the first rate r_0. If the number of missed ACKs is 1, it means that the first transmission failed and the second one was completed successfully. If the number of missed ACKs is 3, it means that the first 3 transmissions failed and the fourth one succeeded. Finally, if the number of missed ACKs if 4, it means that all transmissions failed.

The existing MADWIFI driver implements rate control with a two-stage process which is quite natural given the capabilities exported by the HAL. The short-term variations are handled

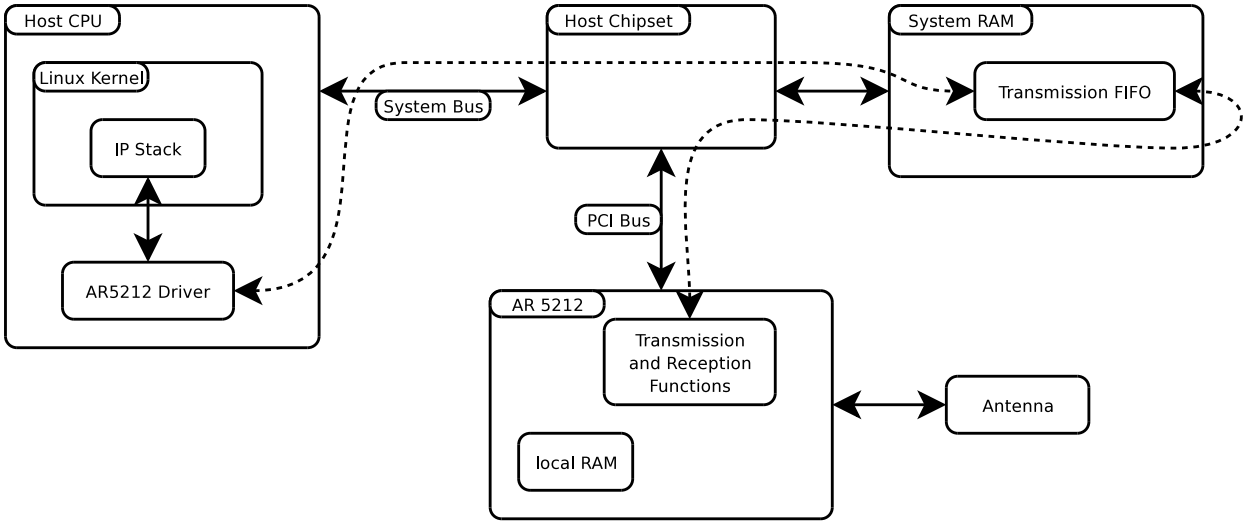


Figure 5.8: System architecture of an AR 5212-based device.

by the Multi Rate Retry mechanism described in the previous section while the long-term variations are handled by changing the value of the r_0/c_0, r_1/c_1, r_2/c_2 and r_3/c_3 pairs at regular fixed intervals (from 0.5 to 1 second intervals). We will discuss the performance evaluation of this mechanism in the next section when we introduce our practical solution using this device.

5.4.10 SampleRate

The SampleRate mechanism [58, 59] is proposed based on the following key observations for lossy link in outdoor environment[3]:

- The rate adaptation mechanism can not predict the performance of higher physical data rate based on low data rate performance. In other words we can not conclude that higher data rates perform poorly since low data rates perform poorly.
- It is possible that the physical data rate with highest throughput suffers from high packet losses (i.e., high data rate, high throughput, and high loss rate)
- The rate adaptation mechanism should avoid measuring the performance of all available data rate as much as possible and limit it to a small set of data rate. As sending the packet with at low data rate wastes the transmission time.

Based on the above key ideas and observations, SampleRate sends packets at different data rate periodically to gather the information about other data rates. Then it switches to a different

[3]Note that the above observations are based on some measurements for lossy links in outdoor environment [51]

data rate if recorded loss rate shows that new data rate has higher throughput. SampleRate also eliminates the data rate that experienced several successive failures.

Another important aspect that SampleRate uses is the average transmission time metric. SampleRate uses this metrics to shift between non-consecutive data rates. SampleRate gets feedback (i.e., successfully transmission packet or number of retransmission) from the wireless card and calculate the transmission time for each data rate considering the packet length as well. Then it makes an average over its recorded transmission time for each data rate. SampleRate uses an averaging windows equal to 10 seconds and makes the average over this window to avoid using stale information.

Initially, SampleRate starts the transmission with the highest data rate. Then, it uses the data rates which are eligible and has the smallest average transmission time (highest throughput) values. This mechanism suffers from a high number of packet losses although it can provide higher throughput for certain network conditions.

5.4.11 Miscellaneous

There are also several research works about analytical rate adaptation algorithms. In [57] and [60], the authors present a complete analytical evaluation of throughput for link adaptation in IEEE 802.11a taking into account the transmission modes and the frame length. [61] presents and evaluates SNR employing to select the best transmission rate. The authors have also modified the MAC header and reservation scheme because of the multi-rate hidden terminal problem [61].

5.5 Our Approaches: AARF and CLARA

Our solution contains three major parts. We first propose a modification to ARF, named *adaptive ARF* (AARF) [6], which can be used for the stable channel condition. Then we propose a mechanism, named CLARA [7], which is a culmination of the ARF and RBAR with additional practical features, that facilitates multipath fading sensing and feedback control signaling. Finally in the separate section, we present our practical rate adaptation mechanism named *adaptive multi rate retry* (AMRR) [6].

5.5.1 Adaptive Auto Rate Fallback: AARF

ARF was designed for a low-latency system based on the second generation of WaveLAN devices. While it is reasonably good at handling the short-term variations of the wireless medium characteristics in an infrastructure network, it fails to handle stable conditions that are the

norm efficiently. Typically, office workers setup their laptop, sit in a chair or at their desk and work there for a few hours. They rarely walk around while typing on their computer keyboard!

In this environment, the *best* rate to choose to optimize the application throughput is the highest rate whose PER is low enough such that the number of retransmissions is low. Typically, higher rates can achieve higher application-level throughput but their higher PERs generate more retransmissions, which then decreases the application-level throughput. Typically, ARF can recognize this *best* rate and use it extensively but it also tries constantly (every 10 successfully transmitted consecutive packet) to use a higher rate to be able to react to channel condition changes. This process however can be costly since the regular transmission failures generated by ARF decrease the application throughput.

The inability of ARF to stabilize for long periods is a direct consequence of the belief that the long-term variations of the wireless medium can be handled by the same mechanism used to handle its short-term variations. While this is true, there is no reason for it to be very efficient.

To avoid the scenario described above, an obvious solution is to increase the threshold used to decide when to increase the current rate from 10 to 40 or 80. While this approach can indeed improve performance in certain scenarios, it does not work in practice since it completely disables the ability of ARF to react to short-term channel condition changes.

This problem led us to the idea that forms the basis of AARF: *the threshold is continuously changed at runtime to better reflect the channel conditions.* This adaptation mechanism increases the amount of history available to the algorithm, which helps it to make better decisions. In AARF, we have chosen to adapt this threshold by using a *binary exponential backoff* (BEB, as first introduced in [62]).

When the transmission of the probing packet fails, we switch back immediately to the previous lower rate (as in ARF) but we also multiply by two the number of consecutive successful transmissions (with a maximum bound set to 50) required to switch to a higher rate. This threshold is reset to its initial value of 10 when the rate is decreased because of two consecutive failed transmissions. Detailed pseudo code that describes formally the behavior of ARF and AARF is included in the Appendix C.

The effect of this adaptation mechanism is to increase the period between successive failed attempts to use a higher rate. Fewer failed transmissions and retransmissions improves the overall throughput. For example, Figure 5.9 shows a period of time where the most efficient transmission mode is mode 3. ARF tries to use mode 4 after 10 successful transmissions with mode 3 while AARF uses the history of the channel and does not increase the rate at each 10 successful packet boundary.

5.5.2 Closed-Loop Adaptive Rate Allocation: CLARA

The key idea of the CLARA is based on the following observation [7]:

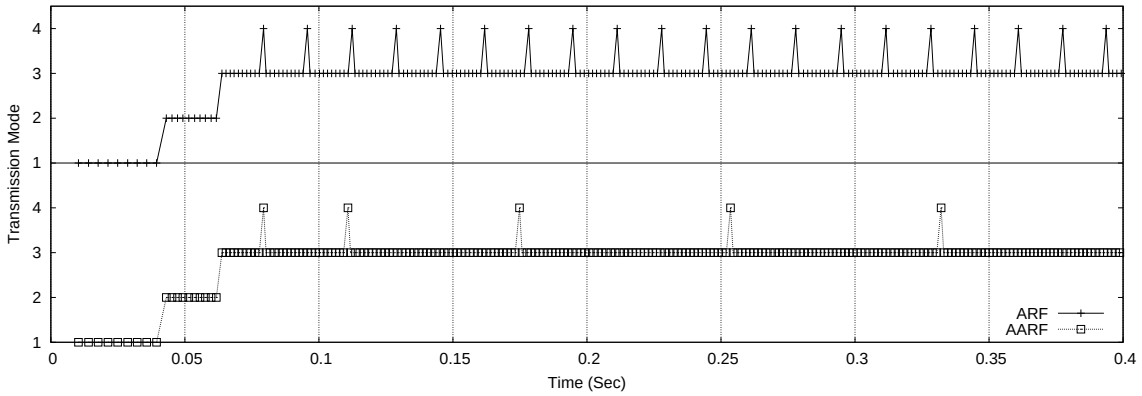


Figure 5.9: Mode selection comparison between ARF and AARF.

- The frame loss in IEEE 802.11 is resulting from a MAC collision or PHY corruption.
- We can differentiate the cause of frame corruption to stable versus unstable channel state.
- MAC collisions can be eliminated by RTS/CTS handshake.
- The stability of the channel can be probed by partitioning a data frame into many smaller fragments. Fragmentation also allows a finer level of channel sensing.

We almost explained all the above observations in the previous chapters except the last one. So, here we first address the last observation. Basically, the bit errors occur if a frame that is transmitted requires a higher target SNR than the instantaneous SNR offered by the channel at that time. It is also possible that due to non-stationarity of the multipath fading, time varying channel, bit errors occur in bursts during periods of *deep fade*. This is an example where a longer frame duration is more susceptible to data corruption due to fluctuation in the received signal strength. The stationarity of a channel is commonly measured in terms of its coherence time T_c. For the successful delivery of a MAC frame, we must consider the relative duration of the channel coherence time over the entire MAC frame duration T_f.

Consider an example, illustrated in Figure 5.10, where the non-stationary channel has two different signal strengths in intervals $\Delta_1 = 0 \leq t \leq T_1$ and $\Delta_2 = T_1 \leq t \leq T_2$. In a unique situation where the initial and final time epochs of the data frame are located in different Δ intervals $T_i \in \Delta_1$ and $T_i + T_f \in \Delta_2$, two scenarios are possible. If a lower rate mode is selected such that the channel is stable for the entire frame duration, the frame is received error-free but at a sub-optimal transmission rate. On the other hand, if a higher rate mode is selected such that the channel is unstable, the frame is corrupted and must be retransmitted with a reduction in overall throughput. Throughput is maximized if DATA can be transmitted in two different modes. Such a mode selection procedure can be achieved by the appropriate fragmentation of

the data frame. As shown in Figure 5.10, each fragment duration T_{frag} is carefully chosen to retain channel stability during the transmission of each fragment.

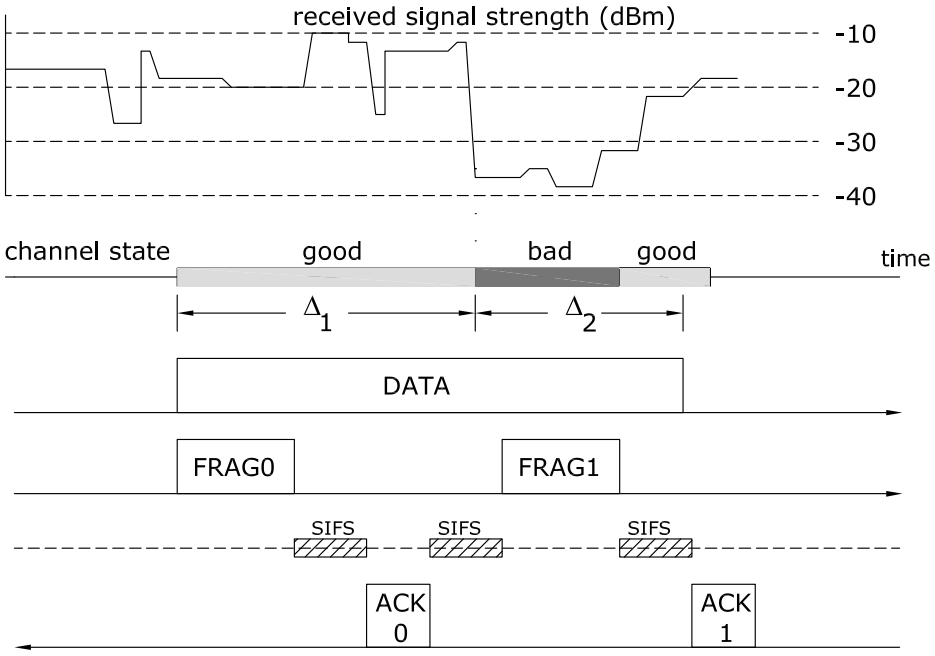


Figure 5.10: DATA or fragment delivery over a non-stationary channel.

Now we describe how all these functionalities are incorporated into CLARA. With every connection, we propose the use of the RTS/CTS protocol for the initial hand-shake and channel sensing if the frame duration is above the RTS threshold. If the RTS/CTS handshake is successful, the channel is reserved for the entire duration of the data frame. As discussed in Section 5.4.2, in RBAR, the RTS/CTS handshake is used as a two-step process to probe the physical layer channel condition. In RBAR, channel probing via RTS/CTS is mandatory regardless of the frame size or data rate. If an ACK is not received, it assumes the cause is a bad channel state (due to non-stationarity of the PHY channel) or poor mode selection (due to inaccurate receiver feedback measurement). However, it is not possible to differentiate one type of frame loss from the other in RBAR. This is because RTS/CTS serves only as initial probing of the PHY channel. During the transmission of DATA, the channel statistics may have changed and the initial estimate based on RTS reception may not be adequate.

In our proposed scheme, RTS/CTS is used in the conventional sense of channel reservation and hidden node identification. Unlike RBAR, we do not measure channel statistics and SNR based on RTS reception. Therefore, the CTS duration/ID field need not be altered. The transmission mode for data delivery is selected a priori based on feedback data over previous CTS and ACK frames. In this respect, our rate selection method is similar to ARF since the

history of past channel access statistics are used in rate selection. Unlike ARF, our scheme has a much broader knowledge of the channel statistics by using feedback reserved bits as well as an increase in the number of ACKs due to fragmentation. Unlike RBAR, there is no mandatory requirement for RTS/CTS handshake in our scheme.

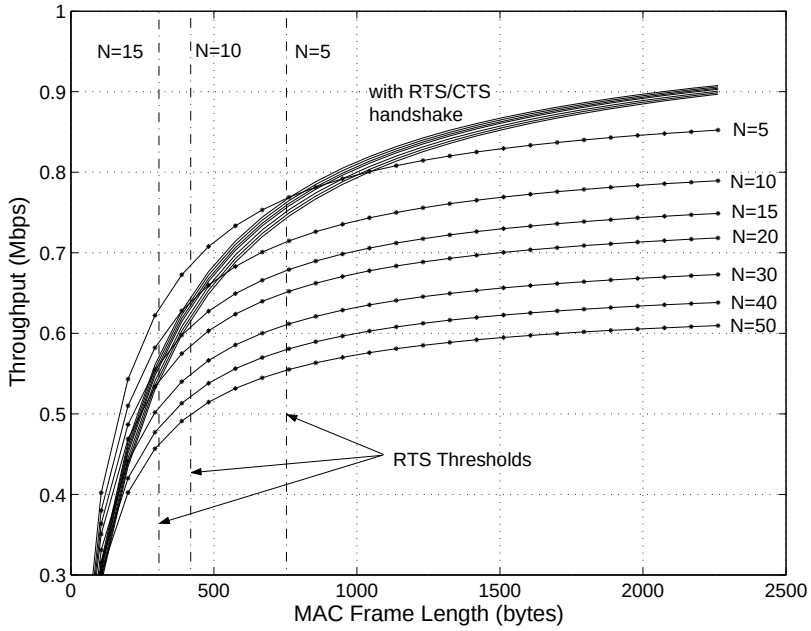
The use of data fragmentation is also an option in our scheme. Based on the history of RSSI (similar to [28] and [48]) from previous fragment ACKs, we can deduce the channel coherence time. Unlike the RTS/CTS handshake, we use fragments and their corresponding ACKs for the sole purpose of PHY channel probing. It is stated in the 802.11 standard [1] that a data fragment and its ACKs serve the role of a virtual RTS/CTS. We should, however, be aware that fragmentation is used to probe the PHY channel stability. It is more appropriate to view the combination of RTS/CTS and fragmentation as the creation of a virtual duplex PHY channel.

It should be noted that the drawback of using RTS/CTS is its overhead equal to $T_{RTS} + T_{CTS} + 2SIFS$, where T_{RTS} and T_{CTS} are frame durations of RTS and CTS, respectively and SIFS is the duration of short inter-frame spacing. In deciding when to invoke RTS/CTS, we must compare this overhead loss against the potential gain offered by RTS/CTS by avoiding MAC collisions.

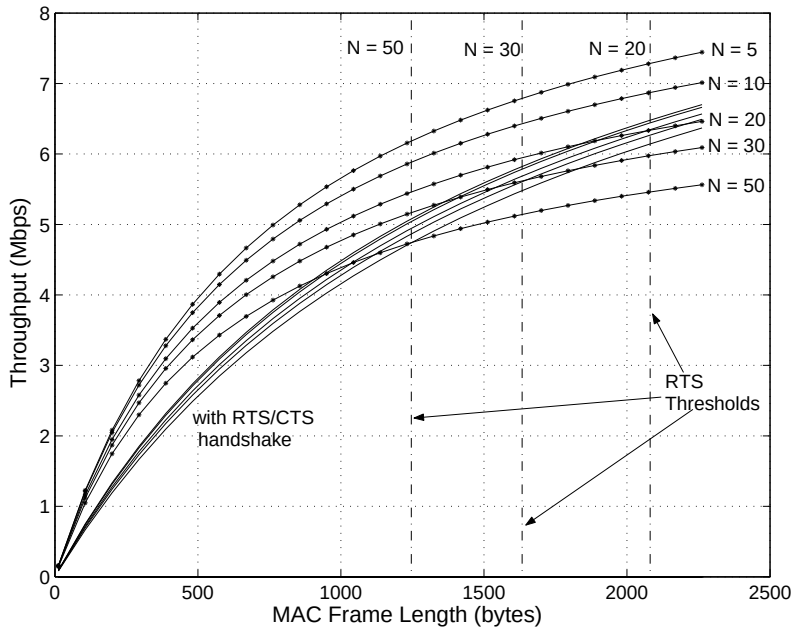
It can be seen from Figures 5.11, 5.12, and 5.13, that the RTS threshold Δ_{RTS} is highly sensitive to the number of contending STAs (denoted by N in the figures), preamble length, and data rate. The plots are generated using a modified Bianchi's model [30] presented in [33] (see Chapter 4). The beaded curves represent throughput plots in basic CSMA mode without RTS/CTS. In the highest rate mode (e.g., 11 Mbps in Figure 5.11) with a long preamble, RTS/CTS handshake reduces goodput for any practical frame size since $\Delta_{RTS} > 1500$ bytes for $N < 36$. However, it is shown in Figure 5.12 that when a short (96 μs) preamble is used, $\Delta_{RTS} \approx$ 500 bytes and the RTS/CTS mechanism should be invoked. We also note that based on similar plots for 802.11a shown in Figure 5.13, RTS/CTS can be used for $N \geq 15$ for all Tx modes (6 to 54 Mbps).

By invoking RTS/CTS we avoid MAC collisions and the unnecessary dead time of collided data frames. A similar situation is observed in the PHY layer. If the DATA frame is long, it may be corrupted due to bad channel states or poor rate selection. By invoking fragmentation, we avoid PHY corruption and unnecessary dead time of a corrupted DATA frame. The price for this improvement is the overhead: $(N_{frag} - 1)\left[T_{ACK} + 2SIFS\right]$; T_{ACK} is the ACK duration and N_{frag} is the number of fragments per channel access. Analogous to the RTS/CTS, the fragment threshold is highly sensitive to the channel statistics (in particular, its stationarity) and frame duration. As we will show, we may still want to invoke fragmentation (despite its small loss in goodput) since it improves latency and data flow.

Finally, it should be noted that by utilizing reserved bits in the SERVICE field of the PLCP headers for every type of frame, feedback control information can be exchanged in each transmitter receiver pair. The PLCP header structure for 802.11a and 802.11b modes are shown

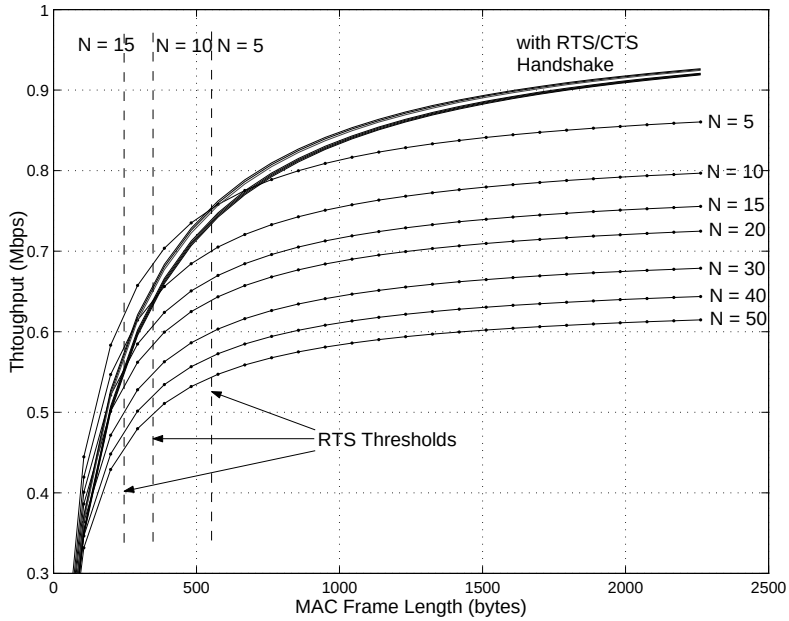


(a)

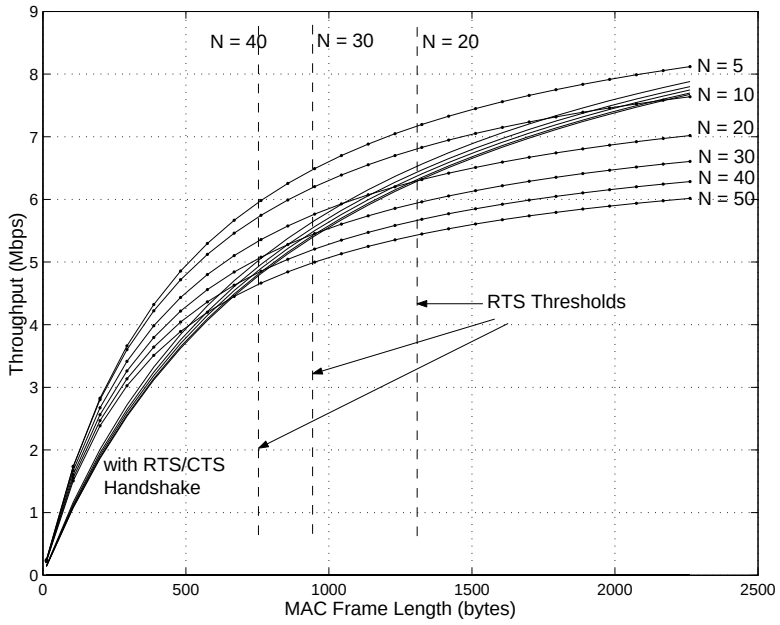


(b)

Figure 5.11: RTS Thresholds for IEEE 802.11b (a) 1 Mbps and (b) 11 Mbps modes with long (192 μsec) PLCP preamble and header.

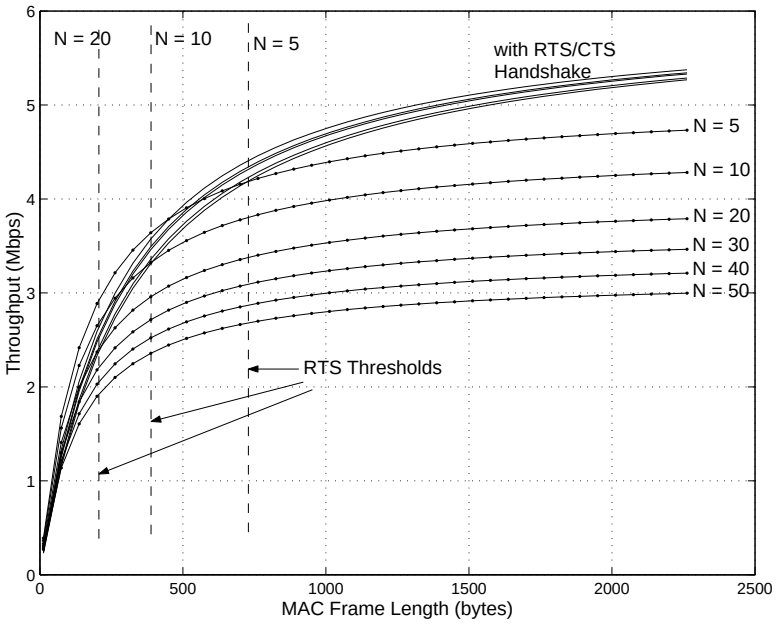


(a)

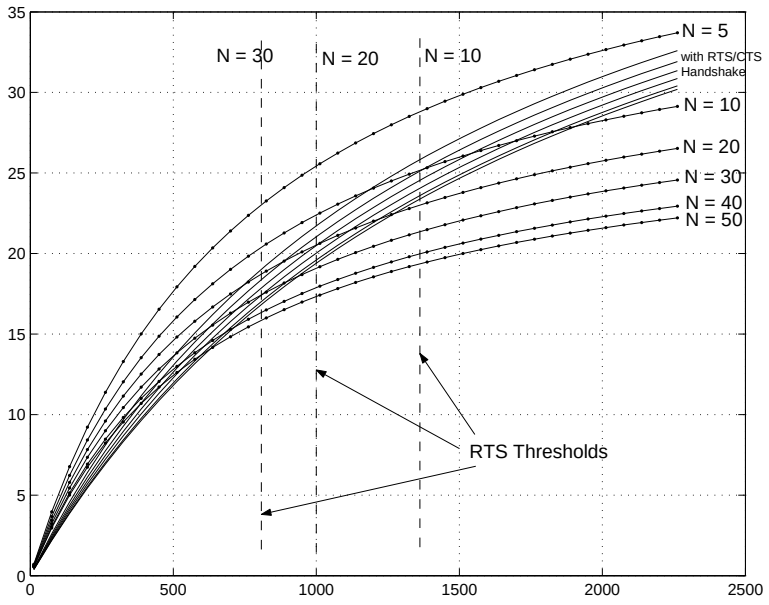


(b)

Figure 5.12: RTS Thresholds for IEEE 802.11b (a) 1 Mbps and (b) 11 Mbps modes with short (96 μsec) PLCP preamble and header.



(a)



(b)

Figure 5.13: RTS Thresholds for IEEE 802.11a (a) 6 Mbps and (b) 54 Mbps modes.

in Figure 5.14. There are 5 and 9 reserved bits available in b and a modes for carrying feedback information. In practice, these bits will be used to carry RSSI and other relevant control information.

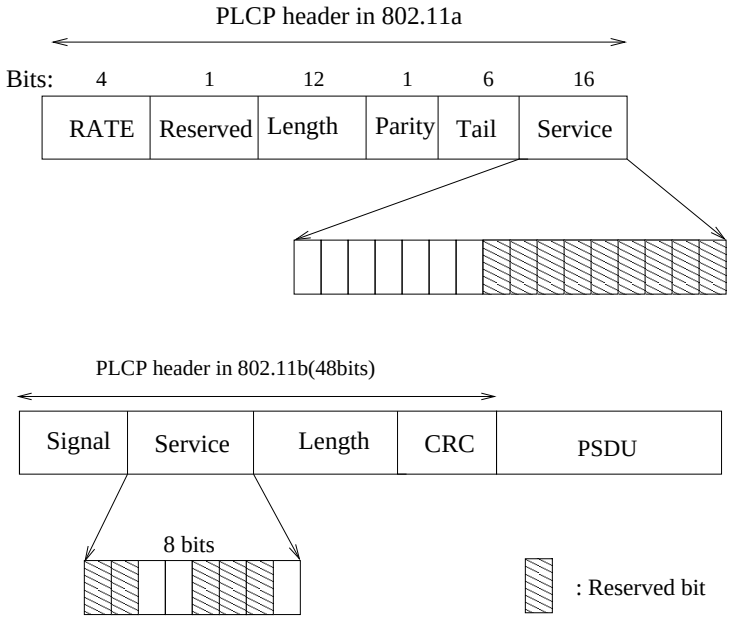


Figure 5.14: Reserved bits in PLCP header at IEEE 802.11a/b standard.

5.6 AARF and CLARA Performance Evaluation

In this section, we present the performance evaluation of our proposed rate adaptation mechanisms with simulation. We have used NS-2 [63] simulations to evaluate and compare the performance of AARF, ARF, and RBAR. Then we have conducted some simulations in MATLAB to evaluate the performance of CLARA in different wireless channel conditions, particularly for stationary and non-stationary wireless channel and have compared it with ARF.

5.6.1 Performance Evaluation of AARF

Because we specifically designed AARF to work well in an infrastructure network[4], we focused on comparing its performance in this environment with ARF and RBAR. To do so, we

[4]AARF is based on ARF which requires at least 10 packets to switch to a higher rate when the transmission conditions improve. In a dense ad hoc network, the wireless medium characteristics can vary many times during the transmission of 1 or 2 packets mainly because of the high collision probability. In this context, it is impossible for ARF or AARF to adapt to the channel characteristics correctly which is why we do not present any ad hoc simulation results in this chapter.

performed simulations based on the simulation environment described in [42] that uses the NS-2 network simulator, with extensions from the CMU Monarch project and the RBAR implementation from [44].

The version of NS-2 on which the RBAR implementation is based does not directly support infrastructure networks. It only supports ad hoc networks and offers the choice of numerous ad hoc routing algorithms. As such, it was impossible to evaluate this algorithm in a multi-node infrastructure network. We thus decided to use the methodology described in [44]: infrastructure networks were simulated with a 2-node ad hoc network. One of the motivations for doing this was also to be able to reproduce the simulation conditions of the results published in this paper exactly and thus to be able to achieve a fair comparison with RBAR.

Our network contains two stations. Station A remains static while station B moves toward station A. The movement of station B is not continuous: it stays static for 60 seconds before moving 5 meters towards station A. Whenever station B stops, a single CBR (Continuous Bit Rate) data transmission towards station A is started. Each CBR packet is 2304 bytes long. Each CBR flow attempts the transmission of 30000 packets 0.8 msec apart which generates a 24 sec continuous data flow. Because the simulations that do not use the RTS/CTS mechanism can achieve a higher throughput peak than what these default CBR flows provide, for these simulations, we used a CBR flow of 50000 packets, each 0.46 msec apart.

As shown in Figure 5.15, the transmission modes 24 Mbps and 48 Mbps always perform worse than all other modes during our simulations. We thus chose not to use them in all further experiments and simulations. We also removed the 9 Mbps mode because its coverage range is always worse than that of the 12Mbps transmission mode as suggested by [42], [60], and [48].

To analyze the influence of the AARF algorithm parameters, we ran a set of simple simulations which kept all parameters except one constant. The default fixed values as well as the variation range of these parameters are shown in Table 5.2.

Table 5.2: AARF parameters

Parameter	*Default*	*Variation Range*
TimerTimeout (number of packets)	*15*	*11-100*
MinSuccessThreshold (number of packets)	*10*	*1-49*
MaxSuccessThreshold (number of packets)	*50*	*11-100*
SuccessFactor (no unit)	*2*	*1.01-5*

The results of these simulations are shown in Figures 5.16. As detailed in [57], we used a packet-based timer for ARF and AARF rather than the time-based timer originally described

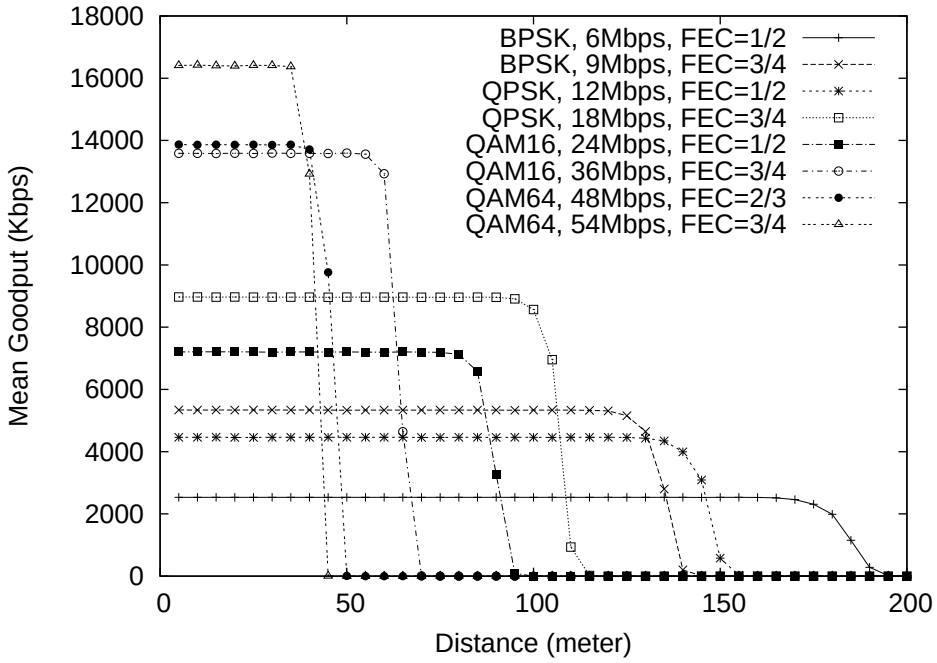


Figure 5.15: Mean goodput for a single hop with the IEEE 802.11a transmission modes.

in [43]. The authors of [44] and [57] had already established that its value had little influence on the behavior of ARF (be it time-based or packet-based) and Figure 5.16-a shows that it also has no noticeable influence on the behavior of AARF. The variations of the number of packets transmitted during one simulation represent less than 0.2% of the minimum number of packets transmitted. Similarly, the variations of *MinSuccessThreshold* in Figure 5.16-b represent less than 0.4% of the minimum number of packets transmitted and those of *SuccessFactor* shown in Figure 5.16-c less than 0.45% of the minimum number of packets transmitted.

MaxSuccessThreshold is the only parameter that has a relatively important influence on the performance of the algorithm (see Figure 5.16-d). The performance of the algorithm obviously reaches a plateau towards the values of 80 to 90. We chose 50 since it seems to offer a good compromise between the performance obtained and the ability of the algorithm to increase its rate within a reasonable amount of time when the user moves toward the access point. The results for the single rate transmissions are presented in Figure 5.15. In these simulations, RTS frames, CTS frames, ACK frames, and PLCP headers are sent with BPSK modulation with a FEC (Forward Error Correction) rate equal to 1/2 and a 6 Mbps data rate (basic mode). Note also that all throughput shown in this section exclude the MAC and PHY headers.

Figure 5.17 shows the mean goodput (the goodput represents the application throughput) achieved by ARF, AARF, and RBAR in the same conditions. These results show that ARF fails to perform as well as the fixed rates for mode 2, 3, and 4. The main reason for this was explained

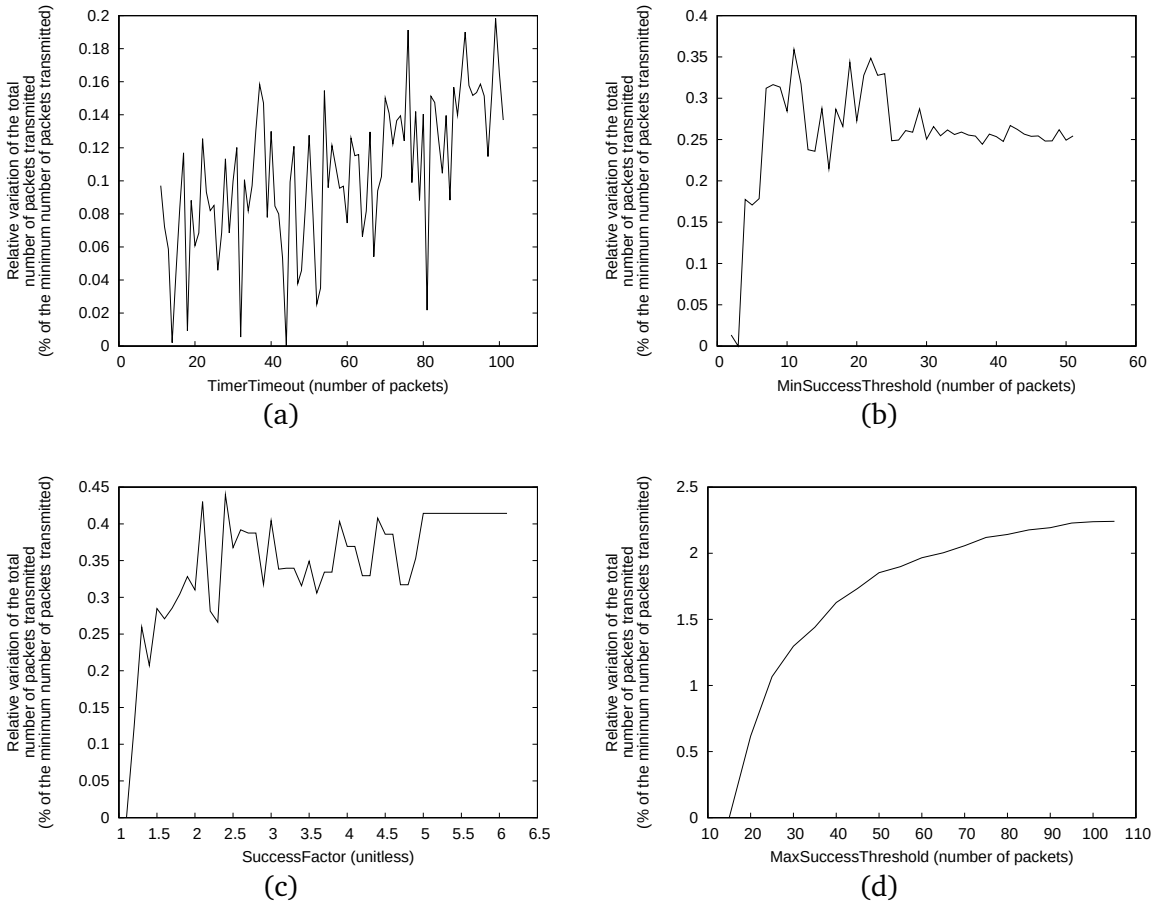


Figure 5.16: Influence of the value of (a) *TimerTimeout*, (b) *MinSuccessThreshold*, (c) *SuccessFactor*, and (d) *MaxSuccessThreshold* on the performance of AARF.

in Section 5.5.1: ARF periodically generates transmission failures. RBAR always picks the best available rate which means that the number of transmission failures is much lower. Its mean goodput is thus much higher. Figure 5.17 shows that AARF performs on average the rate selection as well as RBAR and better than ARF. One of its main advantage over RBAR is that it does not require the use of the RTS/CTS protocol. In this case, its performance, as expected, is much higher than that achieved with RBAR as shown in Figure 5.17.

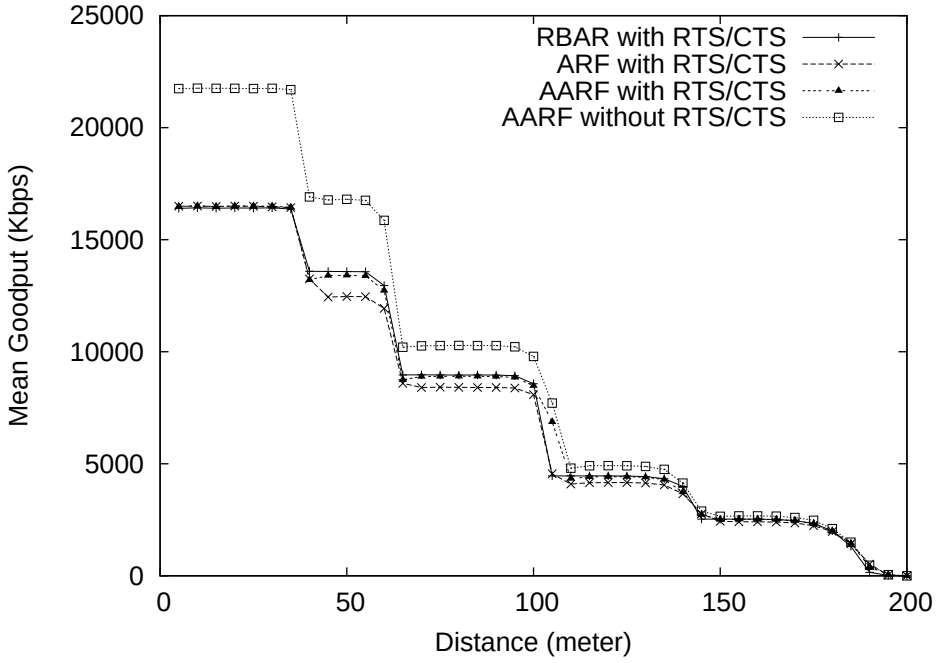


Figure 5.17: Mean goodput for a single hop with three different automatic rate selection algorithms.

The simulation results clearly show the performance improvement offered by AARF over ARF: it can reach on average the near-optimum performance of the RBAR algorithm without requiring any incompatible changes to the 802.11 protocol. Furthermore, all it requires from the hardware is a low communication latency between the block which implements the rate control algorithm and the transmission block which handles the ACK timeouts. This new algorithm can thus be easily and incrementally deployed in existing infrastructure networks with a simple firmware or driver upgrade on each node.

5.6.2 Performance Evaluation of CLARA

To evaluate the performance of CLARA we used event-driven simulations implemented in MATLAB. Here we compare the performance with ARF. We do not compare CLARA with RBAR since RBAR is a subset of CLARA as we explained in Section 5.5.2. In fact, our approach is

a practical and improved RBAR since feedback information is piggy-backed through both CTS and fragment ACKs.

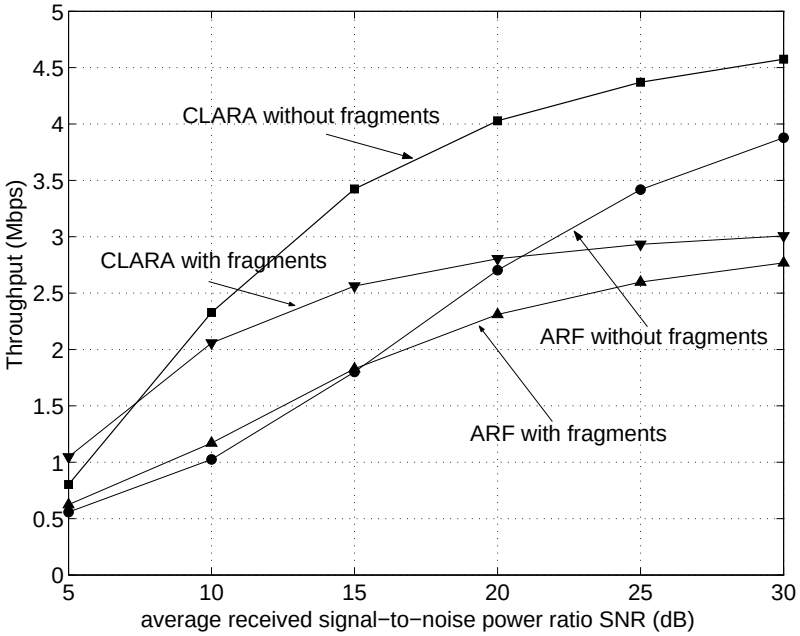
It should be noted that we use the Rayleigh distribution to model the severity of received signal amplitude, quantified in terms of the Rayleigh parameter σ (i.e. the mean value is $\sqrt{\pi/2}$ σ and the received signal power is chi-square distributed). The stationarity of the channel is modelled as a Poisson arrival process with arrival rate λ which is a parameter equivalent to the channel coherence time T_c.

The simulation is carried out in MATLAB. Since in this evaluation, we are studying the PHY layer behavior, there are only two STAs in the system as with the AARF evaluation. This time for each channel access the MAC frame size is 1500 bytes. If fragmentation is used, the maximum number of fragments is 4. The STAs are operating in 802.11b mode with long PLCP preamble and are capable of communicating in all 802.11b modes. In each event 10,000 data frames or 40,000 fragments are exchanged. The sender selects the best mode based on real-time feedback information. For the transmission of RTS, it uses the mode selected for the frame or last fragment from the most previous channel access.

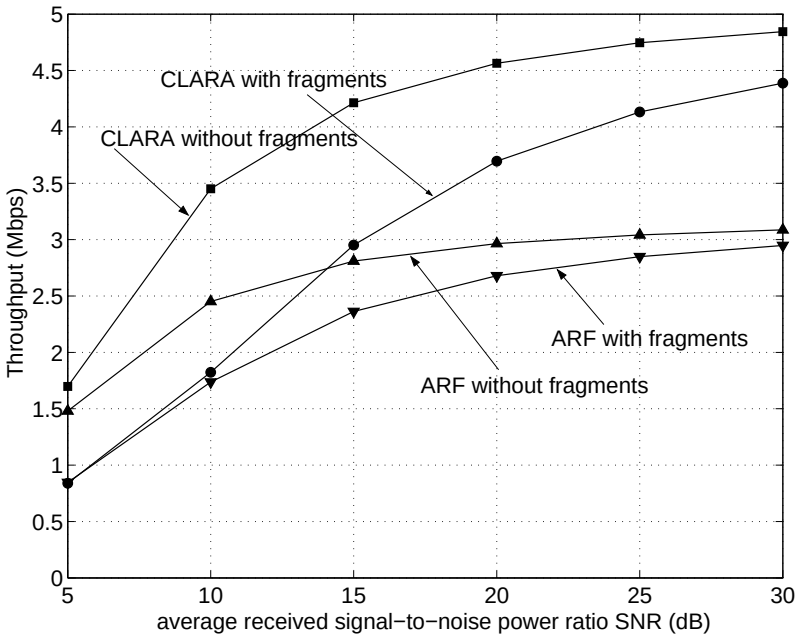
In Figure 5.18, the throughput performance of CLARA and ARF with and without fragmentation is shown. Over all SNR range CLARA outperforms ARF; this is expected. For low SNR (5–15 dB), fragmentation is preferred. The gap between CLARA and ARF closes as the channel becomes more stationary (from 2 to 8 ms). The remaining gap is ARF's inability to adapt received signal fluctuations. For high SNR, fragmentation is not recommended since finer channel sensing is not required and the overhead loss of fragmentation reduces the throughput. Additionally, when the channel is more stationary, there is little performance gain by fragmentation.

In order to show the benefits of using fragmentation in non-stationary channels, we run a series of simulations for T_c=0.4 and 10 ms as shown in Figure 5.19. We see that with fragmentation, we are able to better sense the channel and as a result, the throughput is not susceptible to deep channel fades and variations, resulting in better QoS in terms of smaller delay and steady data flow. In particular, ARF with its slow rate adaptation leads to choppy data flow and long delays. Note that for low received SNR, CLARA with fragments outperforms its cousin without fragmentation. This trend is reversed for either more stationary or less severe channel condition. Regardless, we see that fragmentation in both ARF and CLARA provides a smoother throughput.

And finally to demonstrate the efficiency of CLARA, compared to ARF, we compare their respective time snapshots of rate adaptation in Figure 5.20. Each MAC frame is partitioned into four fragments. The retransmission count starts from 1 for the first attempt. A fragment is dropped after 5 attempts. Transmission modes start from 1 (b/1) to 4 (b/11). Mode 0 denotes a dropped MAC frame. As we can see from the figures, CLARA is able to quickly adapt its mode selection to received SNR variations.

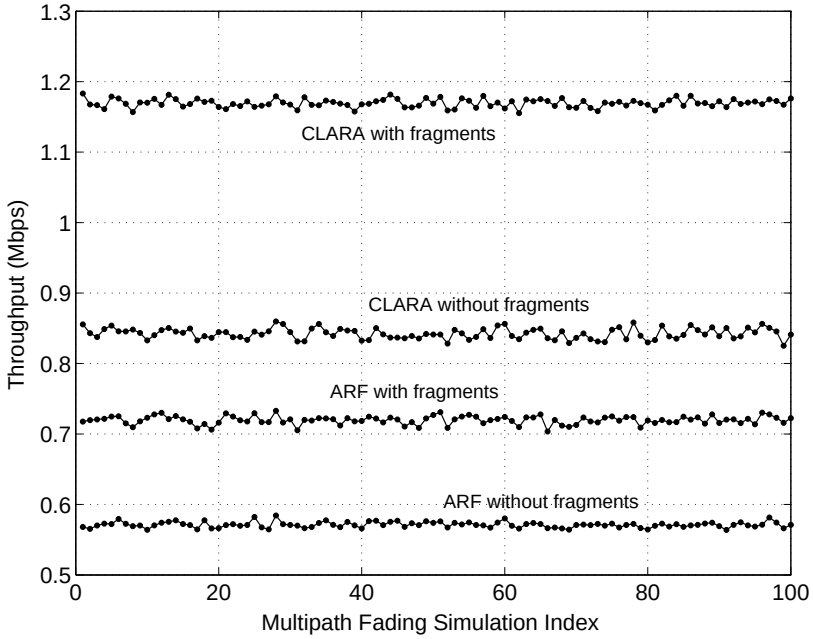


(a)

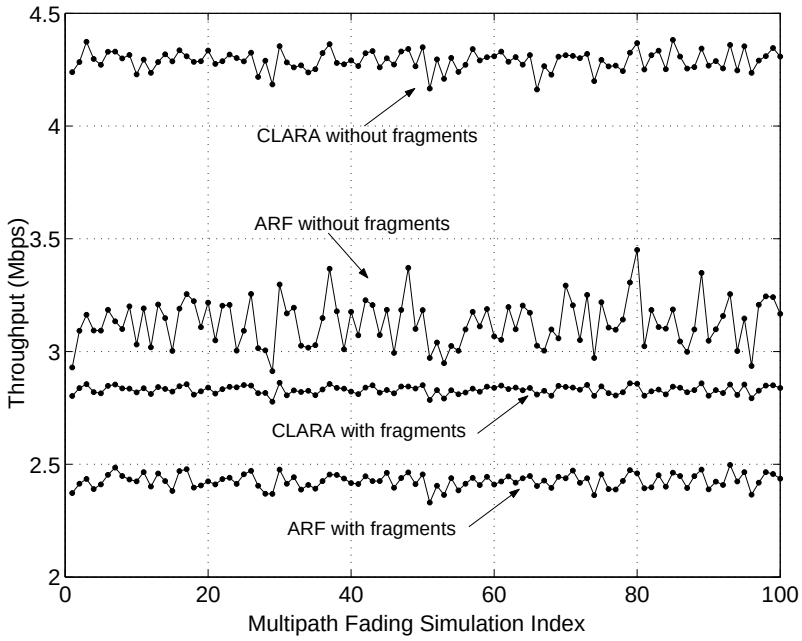


(b)

Figure 5.18: Plot of Throughput vs. received $\overline{\text{SNR}}$ with data fragmentation; (a) coherence time T_c = 2 ms, (b) 8 ms.



(a)



(b)

Figure 5.19: Plot of Throughput for each fading realization; (a) T_c=0.4 ms, $\overline{SNR}$=10 dB , (b) T_c=10 ms, $\overline{SNR}$=15 dB.

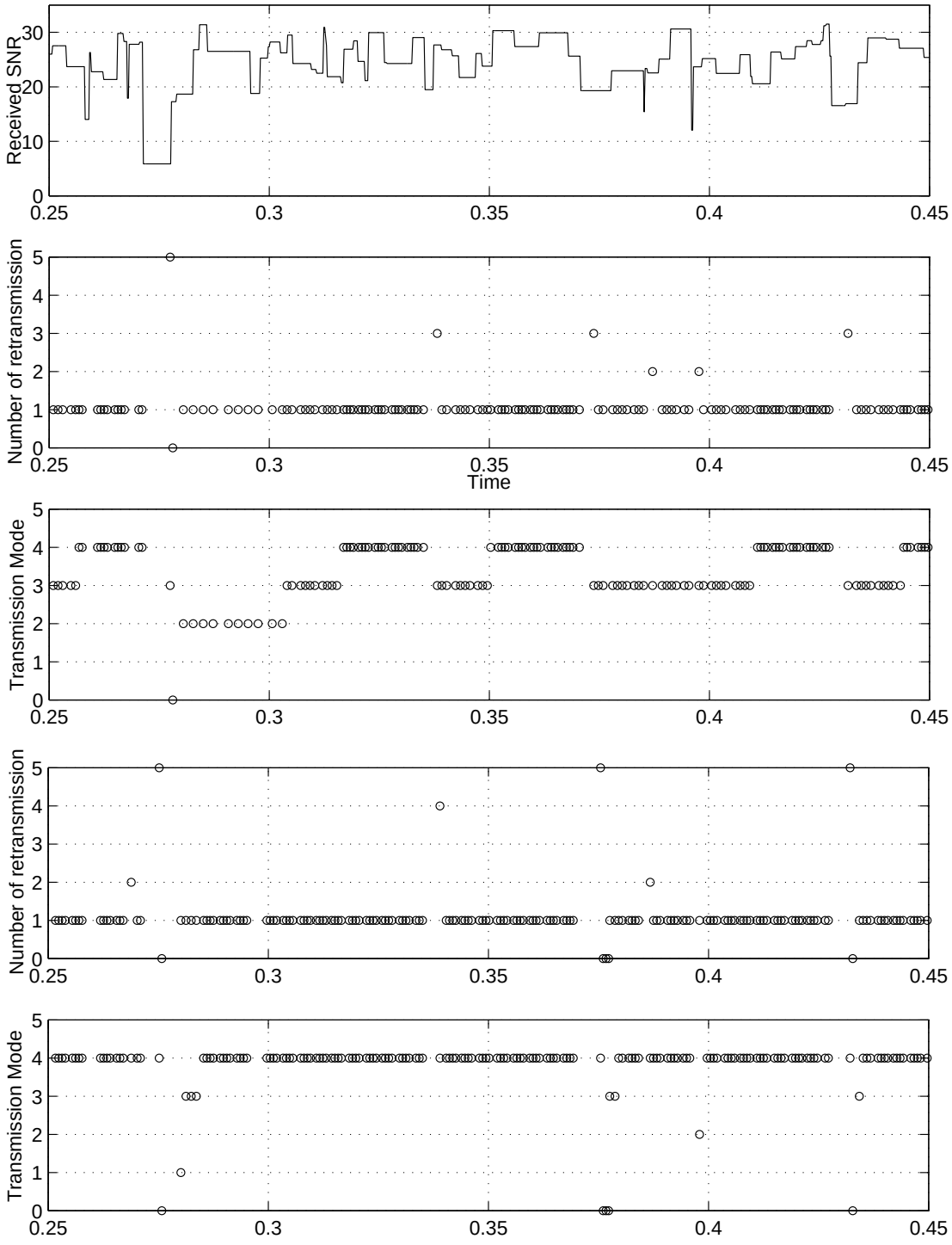


Figure 5.20: A time snapshot of rate selection in ARF (top) and CLARA (bottom) in Rayleigh fading

5.7 Our Practical Approach to Rate Adaptation: Adaptive Multi Rate Retry

Following all performance evaluations for CLARA and AARF, in this section we present some experimental results to improve the performance of MADWIFI algorithm (see Section 5.4.9). While AARF had been designed to work in a *low-latency* system, the AR5212-based 802.11 device to which we had access falls in the *high-latency* group.

A natural way to introduce a Binary Exponential Backoff in the MADWIFI algorithm is to adapt the length of the period used to change the values of the rate/count pairs and this is exactly what *adaptive multi rate retry* does. To simplify the logic of the code, we also decided to use heuristics simpler than those in the MADWIFI algorithm to choose the rate/count pairs at the period boundaries.

To ensure that short-term variations of the wireless medium are quickly acted upon, we chose $c_0 = 1$, $c_1 = 1$, $c_2 = 1$ and $c_3 = 1$ (while MADWIFI uses $c_0 = 4$, $c_1 = 2$, $c_2 = 2$ and $c_3 = 2$). The rate r_3 is always chosen to be the minimum available rate (typically, 6Mbps in 802.11a networks). The rates r_1 and r_2 are determined by r_0: we implemented the simplest heuristic possible by setting r_1 and r_2 to the immediately lower available rates. Finally, our rate control algorithm determines r_0 from the previous value of r_0 and the transmission results for the elapsed period. The Appendix D describes the exact heuristics used.

We used the simulation environment described in Section 5.6 to evaluate the performance of the AMRR algorithm and compared it to that of the MADWIFI and RBAR algorithms. The MADWIFI algorithm we simulated is a trivial copy of the code available in the MADWIFI driver, slightly modified for the simulation environment to use only the 5 transmission modes chosen for our 802.11a networks. Our implementation of the MADWIFI algorithm in the simulator and of the AMRR algorithm both in the simulator and in the driver is straightforward except for the way the transmission FIFO, which is shared between the AR5212 chip and the Linux kernel driver, is handled.

More specifically, the original MADWIFI driver initialized the transmission descriptors present in the FIFO only once, when they were inserted into the FIFO. A rather annoying consequence of this behavior is that it can generate wide oscillations of the algorithm due to the different rates of the packets located at the head and at the tail of the FIFO. For example, when the user application generates a 15 Mbps data flow and if the wireless channel conditions allow the 802.11a 12 Mbps transmission mode with a reasonable PER ($r_3 = 12$, $r_2 = 6$, $r_1 = 6$ and $r_0 = 6$), the source buffers quickly fill (the transmission descriptor FIFO is thus full) and the user application encounters a lot of packet drops at the source. If the PER is low-enough at this rate set, the rate control algorithm will try to increase the rate set to $r_3 = 18$, $r_2 = 12$, $r_1 = 6$, and $r_0 = 6$, this means that every new packet that enters the FIFO uses this new rate set.

However, at the next decision period boundary, the transmission statistics used to adapt the current rate set are those generated by the transmission of the packets whose rate set is $r_3 = 12$, $r_2 = 6$, $r_1 = 6$, and $r_0 = 6$ and that are still present in the FIFO. Because the PER of this rate set is low enough, the rate control algorithm thus will try to increase the rate set again, yielding something like ($r_3 = 24$, $r_2 = 18$, $r_1 = 12$, and $r_0 = 6$).

At one point, the packets whose rate set is high will reach the front of the FIFO and will be treated by the hardware: they are likely to fail which will make the rate control algorithm drop the current rate set quickly. However, it is likely to decrease the rate set too much for the same reasons it increased it too much previously. We have observed this phenomenon during preliminary experiments and we have reproduced it in simulation as shown in the curve named *Original MADWIFI* in Figure 5.21.

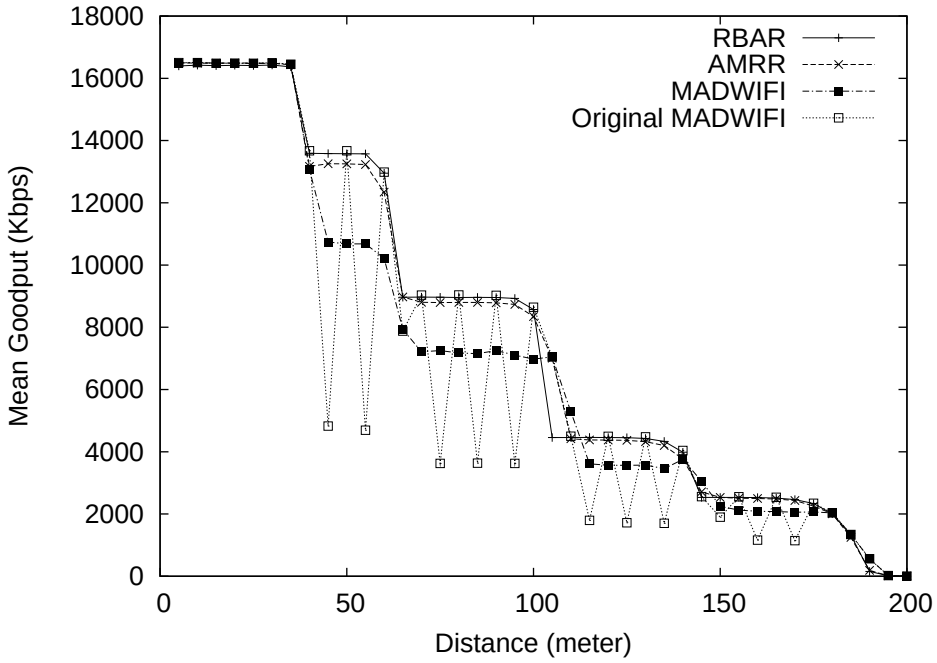


Figure 5.21: Mean goodput for a single hop with RBAR, MADWIFI, and AMRR mode selection.

We chose to avoid this problem by modifying the MADWIFI driver to parse the transmission FIFO each time a rate change happens to apply the rate change to each transmission descriptor concerned immediately. All further simulations and experiments (unless explicitly stated) were conducted with this modified version of the MADWIFI algorithm. It should be noted that the painful and costly process of parsing the transmission descriptor FIFO whenever a rate change needs to be applied could be alleviated if proper hardware support for this had been provided. For example, it should be possible to include in each transmission descriptor a pointer to the rate/count pairs rather than the rate/count pairs themselves. Alternative designs that use an

on-chip cache of rate/count patterns on a per-destination basis would also be possible.

Because the AMRR algorithm is based on the same set of ideas developed for AARF, that is, the use of a BEB to adapt the success threshold, similar parameters can be tweaked. Among these, the *MinSuccessThreshold* and the *MaxSuccessThreshold* parameters are the two most important parameters. We did not bother to evaluate the influence of *MinSuccessThreshold* since increasing it would further decrease the ability of the AMRR algorithm to react to channel condition changes. Figure 5.22 shows how *MaxSuccessThreshold* influences the output of the AMRR algorithm.

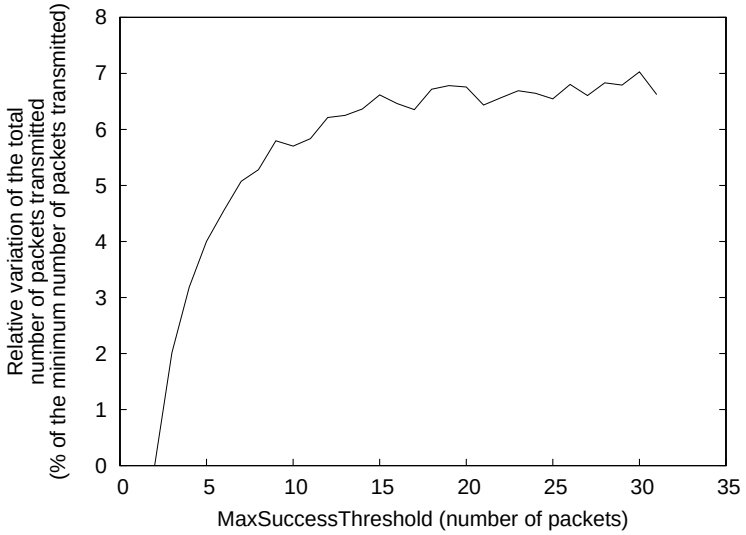


Figure 5.22: Influence of the value of *MaxSuccessThreshold* on the performance of AMRR

As expected, *MaxSuccessThreshold* follows the same pattern observed in Figure 5.16-d: the throughput increases with its increase. As in Section 5.6.1, we do not choose the highest value possible to avoid decreasing its ability to react to channel condition changes rapidly and settle for the value of 15 which is quite close to the plateau maximum.

The simulation results, which are summarized in Figure 5.21, clearly show that AMRR performs much better than the original rate control algorithm used in the MADWIFI driver and that it achieves similar performance to RBAR on average. Here again, the BEB-based adaptive mechanism is the main reason for this throughput improvement: the probability of trying a rate set which requires numerous retransmissions is greatly diminished.

We conducted some experimental evaluations for AMRR as well. Our test setup was created to approximate as closely as possible real-world use cases. As such, we chose a typical office environment with many people walking from one office to the other: a 802.11b/g Access Point (a Netgear WG602) was setup with a private SSID in an office and a laptop with a *Proxim Orinoco Gold* pcmcia card based on the AR5212 chipset was setup in another office approximatively 10

meters away from the Access Point. We first installed an unmodified 2.6.5 Linux kernel and a RedHat 8.0 Linux distribution on the test laptop and then tested three versions of the Madwifi driver:

- *Original Madwifi* : the original unmodified Madwifi driver.
- *Madwifi* : the Madwifi driver modified to apply immediately rate changes on its transmission FIFO as described above in this section.
- *AMRR* : the Madwifi driver modified to apply immediately rate changes and implement the AMRR rate control algorithm.

To mitigate the variations of the transmission conditions with time, we ran three sets of experiments whose results are shown in Figures 5.23. The goal of each of the three sets of experiments was to compare the average throughput achieved by two of the three drivers.

For each set of experiments, we loaded in the Linux kernel alternatively each of the two selected drivers and started a 600 seconds continuous 30 Mbps UDP stream from the laptop to the only machine located on the 100 Mbps ethernet link of the Access Point. We executed this test 5 times for each of the two selected drivers and recorded the average throughput achieved during each experiment.

Despite the variability of the experiments, we can observe the clear performance improvement achieved by *AMRR* over both *Original Madwifi* and *Madwifi* in Figure 5.23. The results show that *AMRR* reached on average 24Mbps while *Original Madwifi* and *Madwifi* reached on average 20Mbps. Figure 5.23-c shows that *Original Madwifi* and *Madwifi* do not have a significative average throughput difference even though we could observe a clear throughput oscillation for *Original Madwifi* during these experiments.

5.8 Conclusions

In this chapter, we identified the key parameters that have to be considered when designing rate adaptation mechanisms. Differences between two classes of 802.11 devices and its influence on the design of practical rate adaptation algorithms have been described. The key parameters of wireless channel and some issues like transmit power control and fairness which are related to rate adaptation mechanisms have been addressed. A detailed survey for existing rate adaptation mechanisms along with their advantages and drawbacks has been presented. Three novel mechanisms, AARF based on ARF [43], CLARA based on ARF and RBAR [44], and AMRR based on MADWIFI [8] have been proposed and evaluated. AARF and AMRR have been respectively designed for *low latency* and *high latency* communication systems. Simulations showed that they both perform close to the optimum in the case of infrastructure networks.

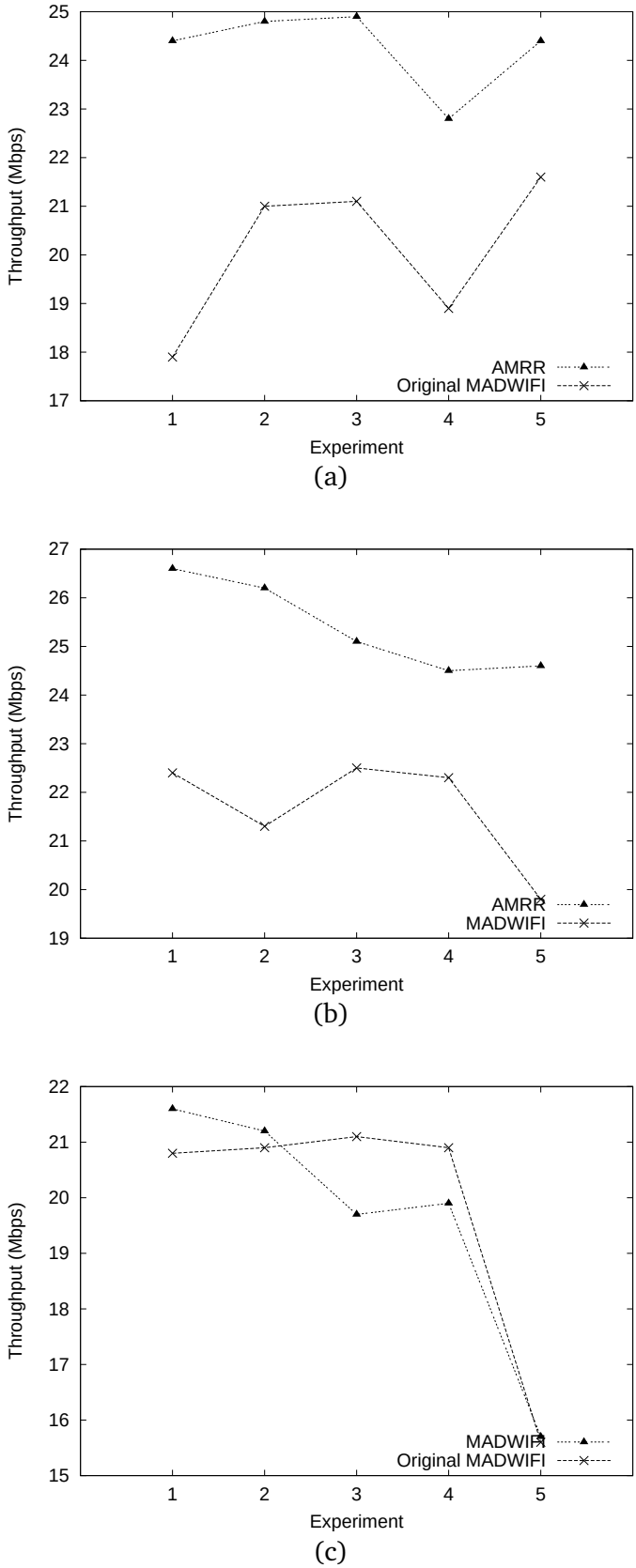


Figure 5.23: Experimental results for (a) AMRR versus Original MADWIFI, (b) AMRR versus MADWIFI, (b) MADWIFI versus Original MADWIFI.

An implementation of the AMRR algorithm in a Linux kernel driver for AR5212-based [64] devices brought further evidences that these algorithms improve the achievable performance and can be readily implemented in existing devices. Simulation results for CLARA showed its advantages over other schemes that incorporate non-adaptive blind-mode rate selection procedures. Since CLARA rate adaptation is implemented at the PHY layer, it is MAC independent and therefore it can be implemented in all existing and emerging 802.11 WLAN standards.

6

MEDIA-ORIENTED RATE SELECTION ALGORITHM: MORSA

6.1 Summary

Current wireless LANs treat multimedia flows and classical data flows alike. Typically, the same error control mechanisms are used for video flows which are generally error-tolerant but delay-sensitive, and TCP flows which are error-intolerant and delay insensitive. The performance of multimedia applications can be significantly improved by some degree of cross-layer awareness. In this chapter, we focus on the optimization of real-time multimedia transmission over 802.11-based wireless networks. In particular, we propose a simple and efficient cross-layer mechanism for dynamically selecting the transmission mode that considers both the channel conditions and characteristics of the media. This mechanism called *media-oriented rate selection algorithm* (MORSA) targets loss-tolerant applications such as audio/video conferencing or VoD that do not require full reliable transmission. We provide an evaluation of this mechanism for MANETs using simulations with NS-2 and analyze the video quality obtained with a fine-grain scalable video encoder based on a motion-compensated spatio-temporal wavelet transform. Our results show that the proposed cross-layer approach achieves up to 4 Mbps increase in throughput and that the routing overhead decreases significantly. The transmission of a sample video flow over an 802.11a wireless channel has been evaluated with MORSA and compared with the traditional approach. Significant improvement is observed in throughput, latency, and jitter while keeping a good level of video quality.

6.2 Introduction

The emerging widespread use of real-time voice, audio and video applications generates interesting transmission problems to solve over MANETs. Many factors can change the topology of MANETs such as the mobility of nodes or the changes of power level. For instance, power control done at the physical (PHY) layer can affect all other nodes in MANETs, by changing the levels of interference experienced by these nodes and the connectivity of the network, which impacts routing. Therefore, power control is not confined to the physical layer, and can affect the operation of higher level layers as well. This can be viewed as an opportunity for cross-layering design and poses many new and significant challenges with respect to wired and traditional wireless networks. Indeed a cross-layering approach becomes necessary to optimize data transmission according to both the characteristics of the data and to the varying channel conditions. Numerous cross layer protocols have already been proposed in the literature [65, 66, 67, 68, 69]. They focus on the interactions between the application, transport, network and link layers. With the recent interest on software radio designs [70], it becomes possible to make the PHY layer as flexible as the higher layers. Adaptive and cross layering interactions can now affect the whole stack of the communication protocol. Consequently, the classical OSI approach of providing a PHY layer as reliable as possible independently of the type of data transmitted becomes questionable.

In this chapter, we focus on the optimization of real time multimedia transmission over 802.11 based MANETs. In particular, we propose a simple and efficient cross layer protocol which dynamically adjusts the transmission mode, i.e., the physical modulation, rate and possibly the *forward error correction*. This protocol called MORSA (*media-oriented rate selection algorithm*) is convenient for *loss tolerant* (LT) applications such as video or audio codecs that do not require 100% transmission reliability (i.e., a certain level of *packet error rate* or *bit error rate* can be concealed at the receiver). Contrary to mail and file transfer applications, several multimedia applications, such as audio and video conferencing or *video on demand* (VoD) can tolerate some packet loss. For example, an MPEG video data flow can contain three different types of packet, *intra-Picture* (I) frames, *prediction* (P) frames and *bi-prediction* (B) frames. I-frames are more important for the overall decoding of the video stream, because they serve as reference frames for P- and B-frames. Therefore, the loss of an I-frame has a more drastic impact on the quality of the video playback than the loss of other types of frames. In this respect, the frame loss requirement of I-frames is more stringent than those of P- and B-frames. Furthermore, as described in Section 6.6, some multimedia applications implement their own error control mechanisms [71, 72], making it inefficient to provide full reliability at the link layer.

MORSA takes into account both the intrinsic characteristics of the application and varying

conditions of the channel. It selects the highest possible transmission rate while guaranteeing a specific bit error rate: the selected transmission mode varies with time depending on the PER or BER tolerance and on the *signal-to-noise ratio* measured at the receiver. We show in this chapter that by adaptively selecting the transmission mode according to both loss tolerance requirements of the application and varying channel conditions, the application-layer throughput can be significantly increased and more stability can be achieved in ad hoc routing. Finally, we evaluate the quality of a sample video transmitted over a wireless 802.11a channel using MORSA and compare it with the quality obtained when we do not take into account characteristics of the application (i.e. using the standard approach). Our results show that MORSA can reach a comparable video quality than the one obtained with the standard mechanism while using only a very low (5%) FEC overhead at the application level instead of the physical layer FEC (50% or 25%). This significantly decreases transmission delay of the application.

Throughout this chapter, we assume that wireless stations use the *enhanced distributed channel access*, proposed in the IEEE 802.11e [13] to support different levels of QoS (see Section 2.4.1). We have modified the NS-2 simulation tool to evaluate the overall system efficiency when considering the interaction between layers in the protocol stack.

This chapter is structured as follows. In Section 6.3 we review related works concerning cross layer protocols in ad hoc networks. The MORSA scheme and a possible implementation within a 802.11-compliant device are presented in Section 6.4. Simulation results with NS-2 are analyzed in Section 6.5. We evaluate quality of a sample video transmission over a wireless channel in Section 6.6. Finally, the conclusion is presented in Section 6.7.

6.3 Cross Layer Algorithms in 802.11 WLAN: Related Works

Several cross layer mechanisms such as mechanisms for TCP over wireless links [65, 69], power control [73], medium access control [66], QoS providing [74], video streaming over wireless LANs [75], and deployment network access point [65] have been proposed.

The Mobileman European project [76] introduced inside the layered architecture the possibility that protocols belonging to different layers can cooperate by sharing network status information while still maintaining separation between the layers in protocol design. The authors propose applying triggers to the network status such that it can send signals between layers. In particular, This cross-layering approach addresses the security and cooperation, energy management, and quality-of-service issues.

The effect of such cross layer mechanisms on the routing protocol, the queuing discipline, the power control algorithm, and the medium access control layer performance have been studied in [66].

A cross-layer algorithm using MAC channel reservation control packets at the physical layer

is described in [68]. This mechanism improves the network throughput significantly for mobile ad hoc networks because the nodes are able to perform an adaptive selection of a spectrally efficient transmission rate.

[75] describes a cross-layer algorithm that employs different error control and adaptation mechanisms implemented on both application and MAC layers for robust transmission of video. These mechanisms are *media access control* (MAC) retransmission strategy, application-layer *forward error correction* (FEC), bandwidth-adaptive compression using scalable coding, and adaptive packetization strategies.

Similarly a set of end-to-end application layer techniques for adaptive video streaming over wireless networks is proposed in [77].

In [78], the *adaptive source rate control* (ASRC) scheme is proposed to adjust the source rate based on the channel conditions, the transport buffer occupancy and the delay constraints. This cross layer scheme can work together with hybrid ARQ error control schemes to achieve efficient transmission of real-time video with low delay and high reliability.

However, none of these algorithms have tried to adapt the physical layer transmission mode in 802.11 WLANs. More examples could be cited, but we are not aware of any cross layer algorithm that takes into account the physical layer parameters (e.g., physical level FEC) as explained in Section 2.5.

It should be noted that standardization efforts are in progress to integrate various architectures. The important co-design of the physical, MAC and higher layers have been taken into account in some of the latest standards like 3G standards (CDMA2000), BRAN HiperLAN2, and 3GPP (High Speed Downlink Packet Access) [65]. IEEE has also considered a cross-layer design in the study group on *mobile broadband wireless access* (MBWA).

6.4 Cross Layer Mode Selection Protocol

This section describes the MORSA mechanism and discusses implementation issues.

6.4.1 Algorithm Description

As we already mentioned, real-time multimedia applications can be characterized by their tolerance to a certain amount of packet loss or bit errors. These losses can be ignored (if they are barely noticeable by human viewers) or compensated at the receiver using various error concealment techniques. In our scheme, the sender is able to specify its *loss tolerance* (LT) such that the receiver uses both this information and the current channel conditions to select the appropriate transmission mode (i.e., rate, modulation, and FEC level).

More precisely, the sender includes the LT information in each RTS packet to allow the

receiver to select the best mode. The LT information is also included in the header of each data packet such that the receiver can decide whether or not to accept a packet. While receiving the RTS, the receiver uses the information concerning the channel conditions along with the information related to LT to select the best data rate for the corresponding packet. The selected rate is then transmitted along with the packet size in the CTS back to the sender, and the sender uses this rate to send its data packets. When a packet arrives at the receiver side, if the receiver is able to decode the PLCP header, it can identify the BER tolerance for the encoded payload. If the packet can tolerate some bit errors, it has to be accepted even if its payload contains errors. As will be shown later, our mechanism makes it possible to define new transmission modes that do not use FEC but that exhibit comparable throughput performance.

To take into account both the SNR and the LT information, we have modified the RBAR threshold mechanism. These thresholds are used to select the best transmission mode in the receiver (see Section 5.4.2). For 802.11a, we assume that the receiver uses FEC Viterbi decoding. The upper bound on the probability of error provided in Section 3.5.4 is used under the assumption of binary convolutional coding and hard-decision Viterbi decoding. Note that on this chapter we use AWGN channel model for our wireless channel model.

Figure 6.1 shows an example of the modifications made for the SNR threshold in RBAR with and without the media-oriented mechanism. Commonly, a BER at the physical layer smaller than 10^{-5} is considered acceptable in wireless LAN applications. By using theoretical graphs of BER as function of the SNR for different transmission modes on a simple additive white Gaussian noise channel (see Figure 6.1), we can compute the minimum SNR values required. Now if a particular application can tolerate some bit errors (e.g. a BER up to the 10^{-3} as shown in Figure 6.1), the receiver can select the highest rate for the following data transmission corresponding to this SNR. For example in Figure 6.1, when the SNR is equal to 5 dB, the receiver can select a 9 Mbps data rate instead of a 6 Mbps data rate if it is aware that the application can tolerate a BER less that 10^{-3}.

We have calculated the thresholds using Equations (3.26), (3.27), and (3.28) for an application that can tolerate up to 10^{-3} BER (see Table 6.2 in Section 6.5.). The receiver can use arrays of thresholds that are pre-computed for different LTs. In the following sections, we describe how such a mechanism can be implemented in 802.11-based WLANs.

6.4.2 Implementation issues

We propose to implement MORSA with the help of the EDCA protocol [11, 12, 17]. As we explained, EDCA is one of the features that has been proposed by IEEE 802.11e to support QoS in WLANs [13] (see Section 2.4.1). In this protocol each QoS-enhanced station (QSTA) has 4 queues to support up to 8 User Priorities (UP). Figure 6.2 shows the QoS control field that is added to the MAC header in the 802.11e specification [13] (see Section 2.9). Bits 6 and 7 of

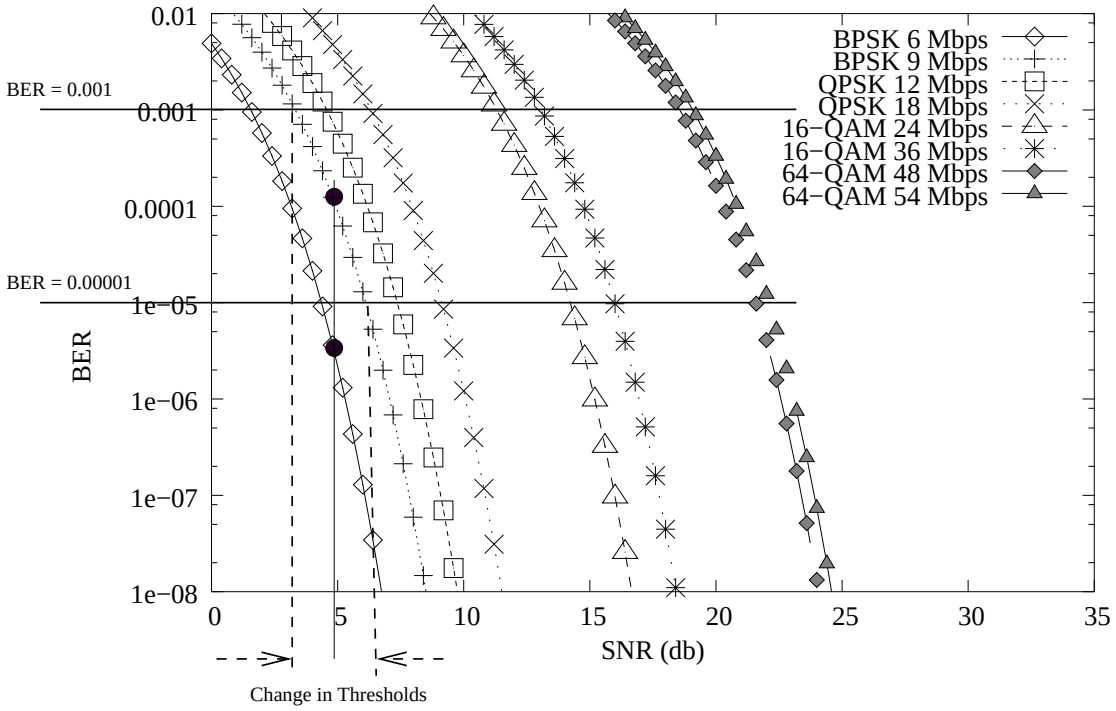


Figure 6.1: Bit error rate versus SNR for various transmission modes (802.11a).

this header can be used to indicate the loss tolerance information. Table 6.1 shows a possible meaning for these two bits in our media-oriented mechanism that should be defined in the process of connection setup. LT information is sent to the receiver by adding one byte to the RTS packets as illustrated in Figure 6.3.

Bits 0-3	Bit 4	Bit 5	Bits 6-7	Bits 8-15
Traffic ID	*Schedule Pending*	*Ack Policy*	*Reserved*	*TXOP duration*

Figure 6.2: QoS control field in the 802.11e.

To make our mechanism operational, it is crucial to let the packets with corrupted payload reach the receiver's application layer. As such, some modifications of the standard are necessary. First, the CRC at the MAC layer should no more cover the payload but only the MAC, IP, UDP, and possibly the RTP headers. Second, the optional UDP checksum must be disabled, as described in the UDP Lite proposal [79]. UDP Lite is a lightweight version of UDP with increased flexibility in the form of a partial checksum. The coverage of the checksum is specified by the sending application on a per-packet basis. This protocol can be profitable for MORSA. Furthermore, to make our mechanism more robust against bit errors, the headers of the different layers (MAC,

Table 6.1: Loss Tolerance classification.

Bit 6-7	*Application Sensitivity*
00	No tolerance in payload
01	Low loss tolerance in payload
10	Medium loss tolerance in payload
11	High loss tolerance in payload

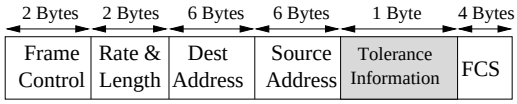


Figure 6.3: Modifications to the RTS header.

IP, UDP, and RTP) have to be sent with the basic rate (see Figure 6.4 and Section 2.9). This is somewhat similar to the reservation sub-header used in [44] as explained in Section 5.4.2. The corresponding bandwidth overhead is investigated in the next section.

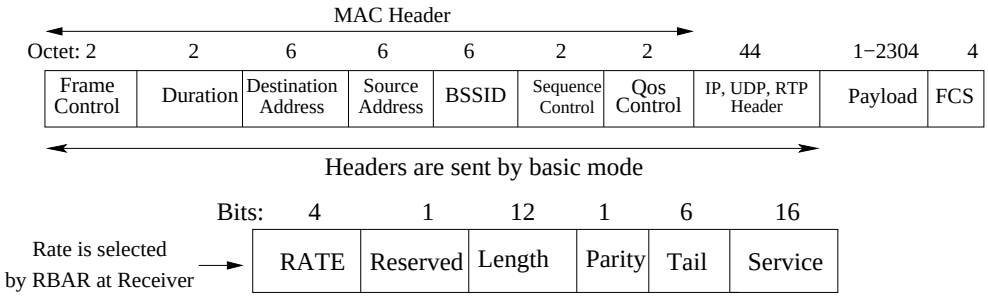


Figure 6.4: Proposed Frame format.

6.5 Simulation Results

Our simulations are based on the simulation environment described in Chapter 4 and Chapter 5 [42] which uses the NS-2 network simulator, with extensions from the CMU Monarch project [80] to simulate multi-hop wireless ad hoc networks. In order to obtain more realistic results, Cisco Aironet 1200 Series parameters are used in our simulations [81]. Note that in the following simulations similar to Chapter 5 , CTS and RTS control packets and PLCP headers

are sent with the basic mode, i.e., BPSK modulation, FEC rate equal to 1/2 and 6 Mbps data rate. All throughput shown in the following figures exclude the MAC and PHY headers; they are denoted as goodput for the remainder of this chapter.

To evaluate the perceived quality for the user using our protocol, we have taken an example of video application that can tolerate 0.1% of bit errors (see Section 6.6.2). Thus, we have investigated the throughput performance of MORSA when the BER is equal to 10^{-3} in the following simulations. Of course other values of the BER can be chosen to perform simulations with similar results.

In our simulation, we assume that bit errors are distributed in a packet according to a binomial distribution. This is an acceptable assumption since the position of the bit errors are not taken into account by NS-2. In Section 6.6, we will provide more precise models for the distribution of bit errors in our data stream. Let n represent the number of bit errors in a packet of N bits, and p be the probability of bit error. The probability of having less than L bit errors can be calculated by:

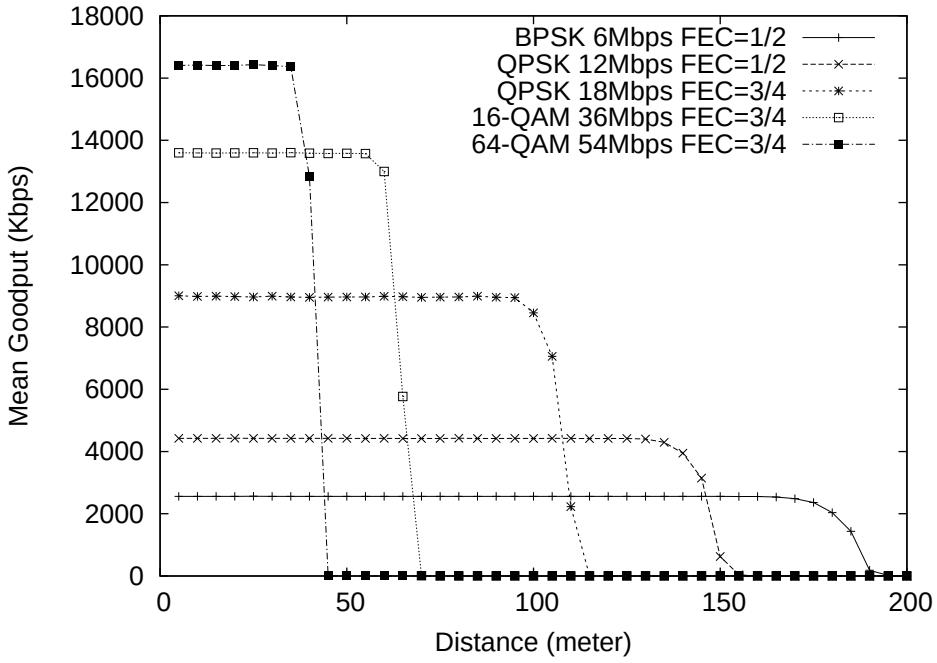
$$P(n \leq L) = \sum_{i=0}^{L} \binom{N}{i} \cdot p^i \cdot (1-p)^{N-i} \tag{6.1}$$

We first evaluate our mechanism in a simple ad hoc network that contains two wireless stations. These wireless stations communicate on a single channel. Station A is fixed and station B moves toward station A. Station B moves in 5 meters increments over the range of mobility (0-200) meters and is held fixed for a 60 seconds transmission of CBR data towards station A. In each step, 30000 CBR packets of size 2304 bytes (including physical layer FEC) are sent.

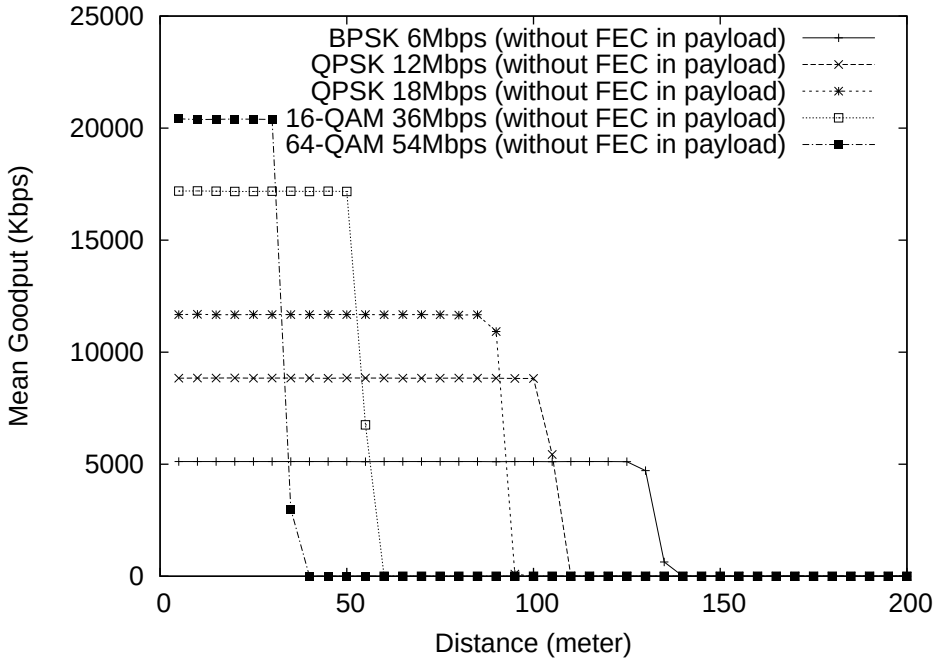
Figure 6.5 shows the mean goodput of this single CBR connection between two wireless stations versus the distance between them for different transmission modes with and without media-oriented mechanism. Based on our simulation study for 802.11a, we have selected five efficient transmission modes out of the 8 possible transmission modes in 802.11a [42] (see Section 5.6.1).

Since no payload FEC is used in our media-oriented protocol, the mean goodput is increased significantly compared to the standard transmission modes. For example, we can observe that the media-oriented mechanism achieves a 4 Mbps mean goodput improvement at the highest rate mode. However, this has a cost in coverage range: in the same example, it is 50 meters less. It should be noted that if an application can tolerate more bit errors, the coverage range will be larger than for the standard transmission modes [11].

We have also evaluated the extra bandwidth overhead of the modified frame format. This overhead is caused by having to send the MAC header at the basic mode and by the additional byte in the RTS packet. Figure 6.6 compares the mean throughput for the traditional RBAR



(a)



(b)

Figure 6.5: Mean goodput versus distance for (a) Standard transmission modes and (b) Media-oriented with 0.1% bit errors.

and for RBAR with the modified frame format. The worst-case overhead at the maximum rate is about 1 Mbps, but the coverage range does not change much compared to the standard specification.

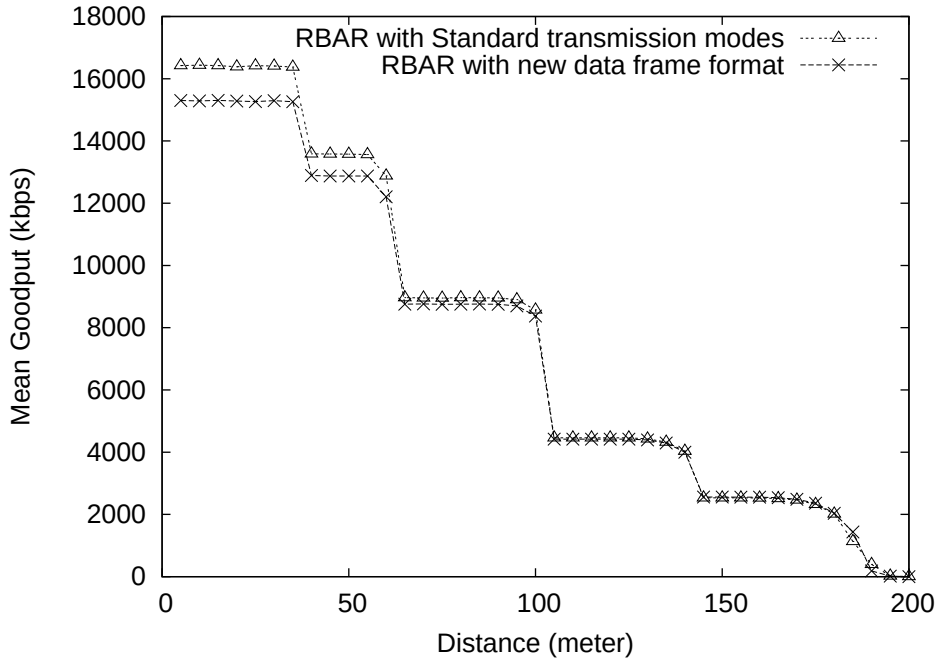


Figure 6.6: Overhead of the modified frame format.

To evaluate the performance of RBAR under different mode selection mechanisms, we need to calculate arrays of thresholds for each mechanism (see Section 6.4). Table 6.2 shows these threshold values for RBAR and MORSA[1]. These results show that if we can tolerate loss we will be able to send data with a higher rate.

Table 6.2: SNR(dB) threshold values to select the best transmission mode

Data Rate *Mbps*	*Standard* *(With FEC)*	*Media-oriented* *(No LT)*	*Media-oriented* *(0.1% LT)*
12	0.68	6.12	4.94
18	4.75	7.37	6.18
36	11.39	14.22	13.5
54	17.29	21.58	20.3

Figure 6.7 illustrates the performance of RBAR and MORSA. Since the standard mode selec-

[1]For a SNR smaller than these values, data will be sent with the basic mode which is 6 Mbps.

tion mechanism can achieve the maximum coverage range and the media-oriented mechanism obtains the maximum mean goodput, we have defined a new media-oriented mode selection mechanism called *hybrid transmission mode selection* or *H-MORSA*, to achieve both objectives at the same time (see Figure 6.8). The five PHY transmission modes that are used for the hybrid mode selection mechanism do not use FEC.

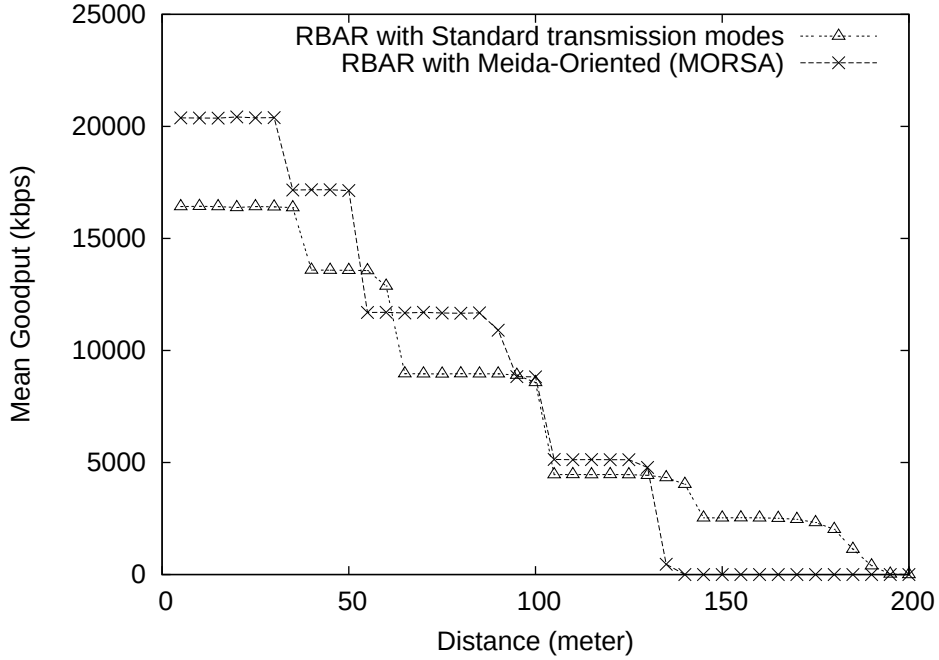


Figure 6.7: RBAR performance for standard and media-oriented protocols (MORSA).

Then, we evaluate the two media-oriented mechanisms (MORSA and H-MORSA) in ad hoc networks. Figure 6.9 shows an example of network configuration for 20 nodes which are commonly used for ad hoc network evaluation [44, 14, 80]. In our simulation, each ad hoc network consists of 20 mobile nodes that are distributed randomly in a 1500 $\times$ 300 meters arena. The speed at which nodes move is uniformly distributed between $0.9v$ and $1.1v$, for different speeds of v. We use the following speed values 2, 4, 6, 8 and 10 $\mathrm{m/s}$. The nodes choose their path randomly according to a random waypoint mobility pattern. The same movement patterns are used in all experiments whatever the mean node speed. For example, if node A moves from point a to point b with a speed of 2 $\mathrm{m/s}$, it will take the same route with 4, 6, 8, and 10 $\mathrm{m/s}$ in the other scenario patterns but with different delays. All the results are based on an average over 30 simulations with 30 different scenario patterns.

In each simulation, a single UDP connection sends data between two selected nodes. Other nodes can forward their packets in the ad hoc network. The data is generated by a CBR source at saturated rate. In other words, there are always packets to send during the whole simulation

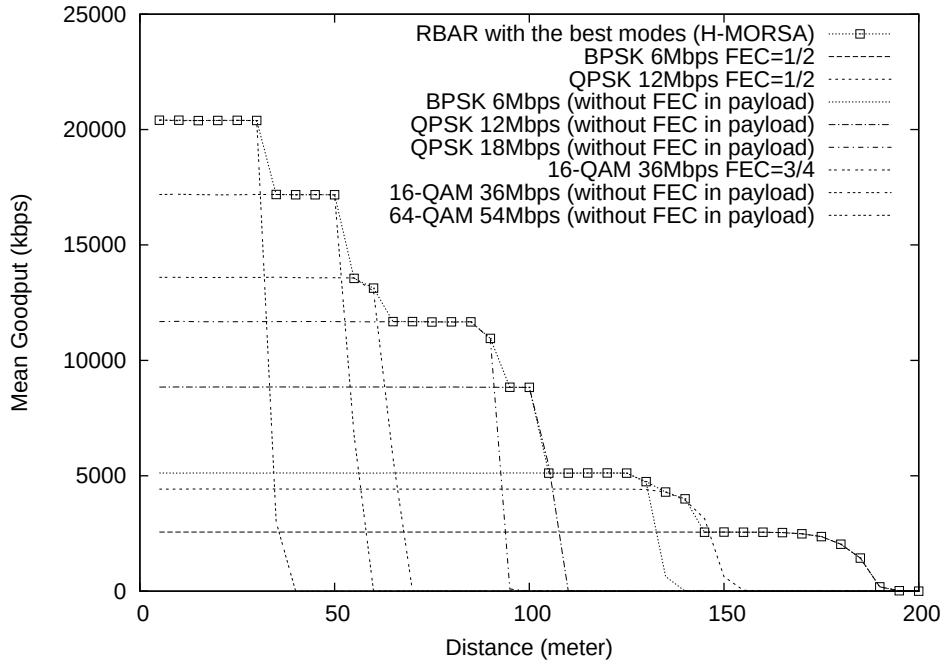


Figure 6.8: RBAR performance using standard or media-oriented protocols (H-MORSA).

time. Unlike in the simple network topology with 2 nodes where we used static routing, here the *dynamic source routing* (DSR) [14] protocol has been used. DSR is a simple and efficient routing protocol designed specifically for use in multi-hop ad hoc networks. It should be noted that routing packets are sent using the basic transmission mode like the RTS, CTS, and ACK control packets.

We use three automatic mode selection mechanisms defined in our previous simulations (see Figures 6.7 and 6.8). In the standard mode selection mechanism (RBAR) and hybrid mode selection mechanism (H-MORSA), we may have a hop in the route between source and destination that uses a physical FEC equal to 1/2. Thus, we have to use packets with a payload length equal to 1152 bytes for these simulations. However, with MORSA we are able to send packets with 2304 bytes since no physical layer FEC is used in this mechanism.

Figure 6.10 shows the mean goodput of a single CBR connection versus the different mean node speeds. For an application that can tolerate a BER of 10^{-3}, the mean goodput is about 25% higher when we take into account the application's characteristics.

Figure 6.11 shows the number of delivered bits for 30 scenario patterns[2] with mean speed equal to $2\mathrm{m/s}$. In the scenarios where the number of delivered bits is zero, DSR was not able to find a route between the source and the destination during the whole simulation time. As ex-

[2]Scenarios are sorted by the number of delivered bits obtained with the standard mode selection mechanism.

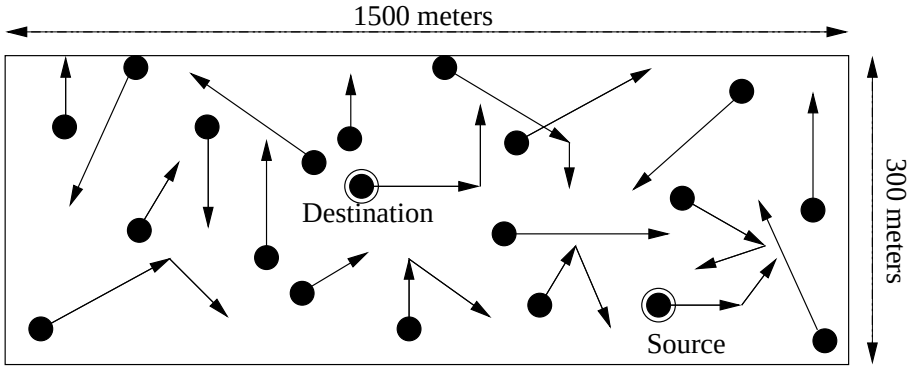


Figure 6.9: Example of ad hoc network topology scenario.

pected, in most of the scenario patterns MORSA can deliver more data bits to the receiver. One interesting observation is that in some scenario patterns (less than 15% of them) the number of delivered bits with the standard RBAR and H-MORSA is more than the one in MORSA. The rationale behind this is that DSR packets can be sent with the maximum coverage range in the standard and the hybrid mode selection mechanisms. As a result, the source can find a route to the destination faster than MORSA. Thus, the number of delivered packets in the standard RBAR and the H-MORSA is more than that of MORSA (e.g., scenario number 20).

We have also evaluated the overhead of the DSR routing protocol in different cases. The DSR algorithm has two different phases called *route discovery* and *route maintenance* to manage the routes in ad hoc networks. In *route Discovery*, ad hoc nodes need to find a route between the source and the destination. This is performed only when the source attempts to send a packet to the destination and does not already know a route. In *route maintenance*, DSR detects changes in the network topology such that the source can no longer use the current route to destination. This can occur if a link along the route is not usable anymore.

Figure 6.12 shows the number of routing overhead packets generated by DSR, which have been sent in ad hoc networks according to different mean speed of the nodes. In order to evaluate this overhead, we have considered all DSR routing packets that should be sent before making a connection and during data transmission. So this overhead includes *route discovery* and *route maintenance* overheads. These results show that routing overhead decreases significantly when we use MORSA. We believe this is a consequence of having more stable connection when MORSA is used.

We have done different simulations to evaluate the performance of our mechanism in the presence of interference for ad hoc networks. For all these simulations, 20 nodes are distributed in an area of 500×100 meters which is 9 times smaller than previous simulation scenarios. In this simulation, 6 UDP connections are set up between 12 different nodes. Data is generated

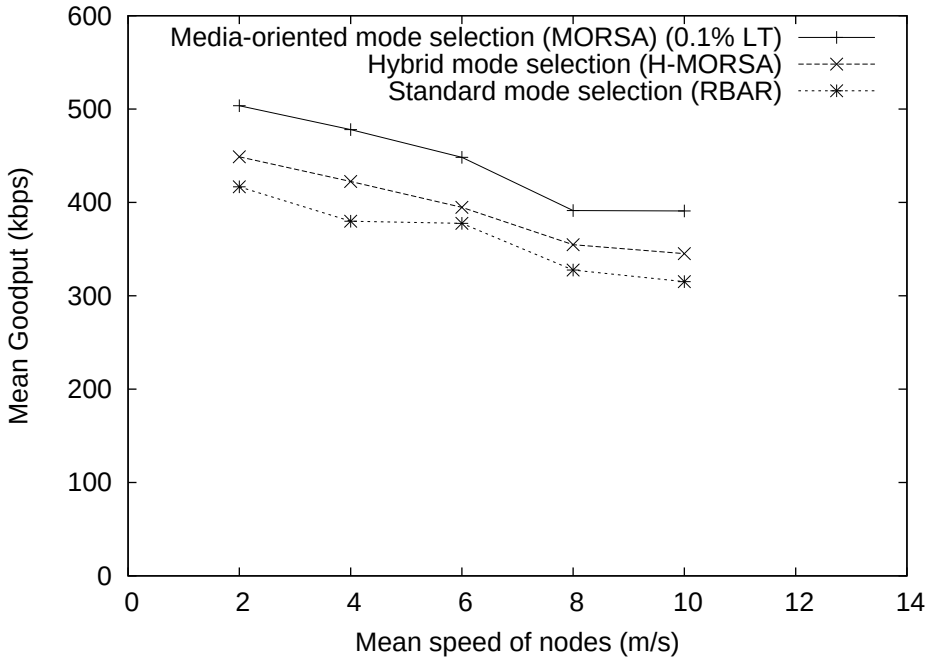


Figure 6.10: Performance comparison for a single CBR connection in a multihop network, with and without MORSA.

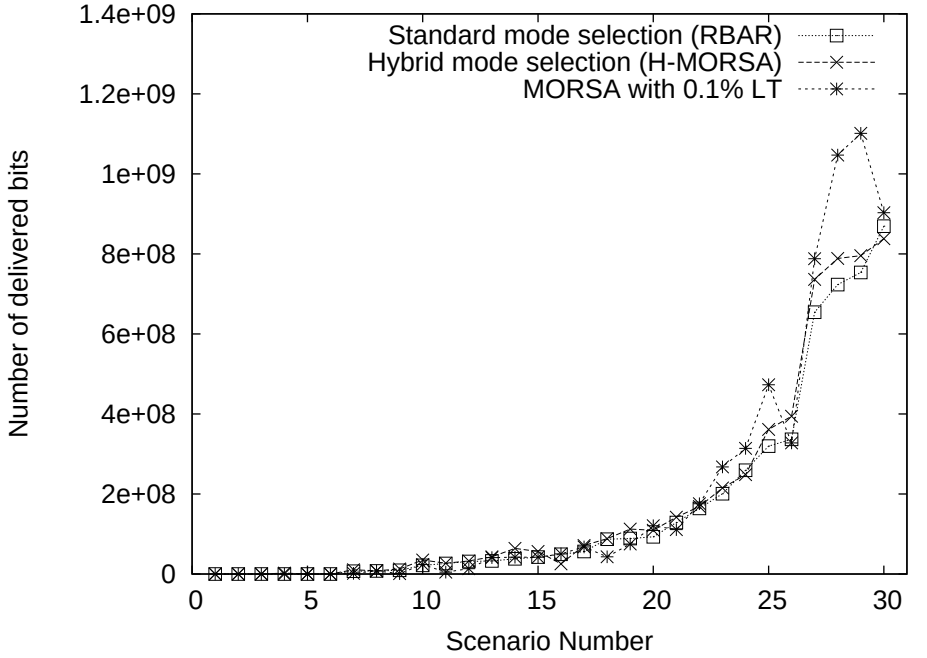


Figure 6.11: Number of delivered bits to the application (speed $= 2$ m/s).

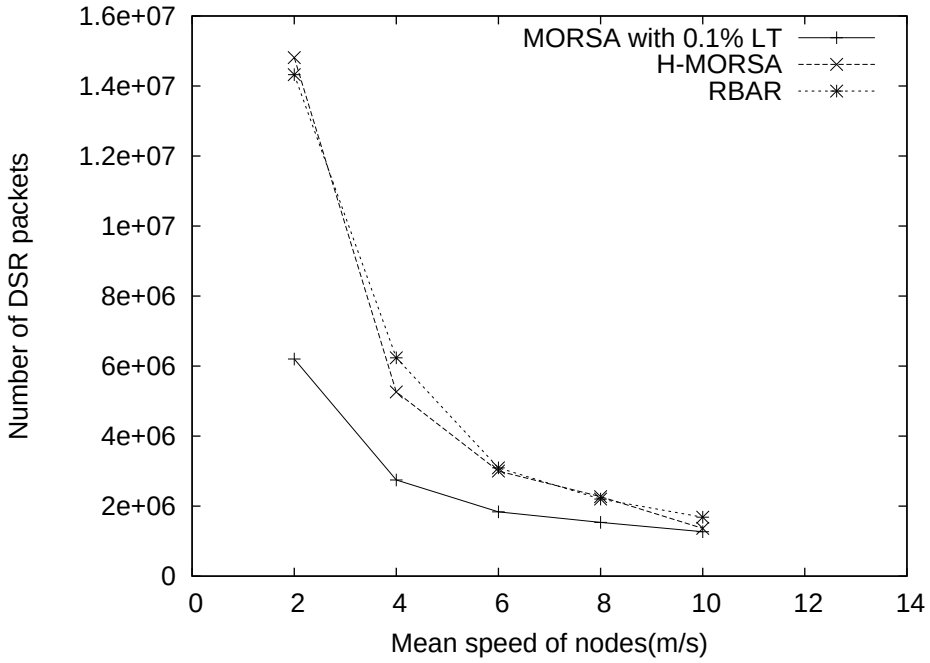


Figure 6.12: DSR routing overhead in multihop network.

by CBR sources at a saturation rate. The first source starts data transmission at time 3.12 and the last one at 25.12. For this simulation, nodes are fixed and DSR does not need to use *route maintenance*. The results are averaged over 30 different scenario patterns. Figure 6.13 shows the performance of MORSA in these experiments. Clearly, MORSA outperforms the standard mode selection (RBAR) and the hybrid mode selection (H-MORSA) mechanisms. This is because the media-oriented mechanism considers the application's characteristics and does not use FEC at the physical layer when the channel condition is good.

6.6 Evaluation of Video Quality

Simulation results in NS-2 have shown a significant improvement in throughput when considering the loss requirements of the application to select the transmission mode. In this section, we evaluate the effectiveness of the proposed media-oriented mechanism using the simulation of a video transmission over a 802.11a wireless channel. Our previous observations about the performance of the media-oriented mechanism can be further justified by the evaluation of the video quality obtained at the receiver when we employ the media-oriented mechanism. In the following sections, we describe a wireless channel model that can estimate the position and the length of burst error bits in 802.11a. Then, we present a video application that can tolerate a BER equal to 10^{-3} by using an application level FEC whose overhead is only 5%. Finally, we

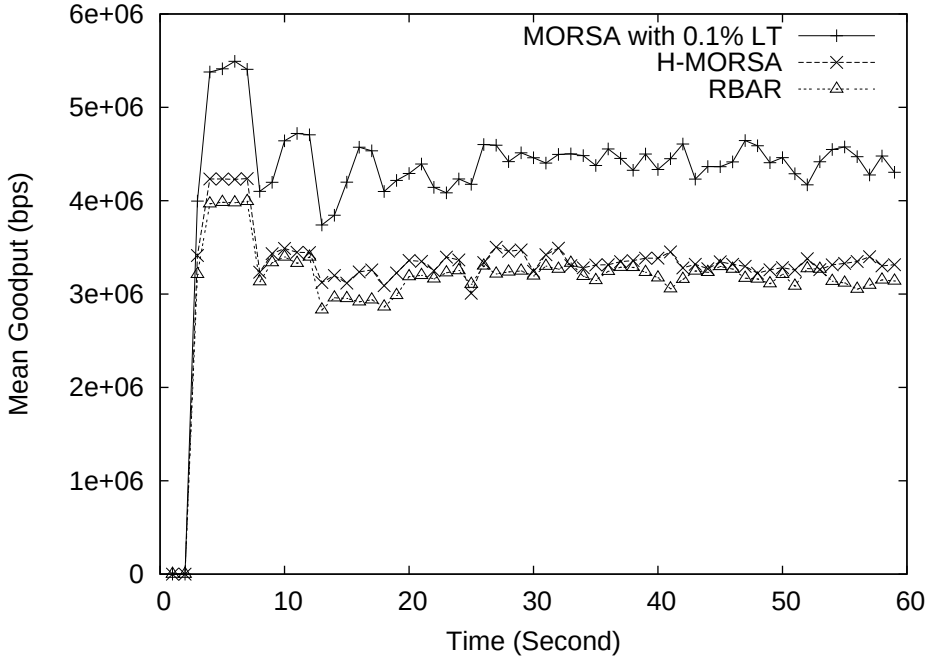


Figure 6.13: Performance comparison for a several CBR connection in multihop network, with and without media-oriented mechanism.

compare the transmission delay and the video quality (*peak signal-to-noise ratio*) with standard and media-oriented transmission mechanisms.

6.6.1 802.11a Channel Model

Wireless channel models can be divided into two main groups: *memoryless models* and *models with memory*. In memoryless models, corrupted bits are produced by a sequence of independent trials. Each trial has the same probability p of producing a correct bit and probability $q = 1 - p$ of producing a bit error. However, in a real communication environment, links have memory and errors often occur in isolated bursts because of multipath fading, impulsive noise, or switch transients. A classic method to model a wireless channel with memory is using a Markov chain. In this model, the probability of bit error depends on the state of the model. We have considered in this section a model with memory, which is based on the model proposed in [82] for 802.11a WLANs.

As we discussed in Chapter 3, in the 802.11a physical layer, the data field shall be encoded with a standard convolutional encoder of different coding rate $R = 1/2$, $2/3$, or $3/4$, depending on the data rate. The $1/2$ convolutional encoder uses the generator polynomials, $G_0 = 1338$ and $G_1 = 1718$, and simple puncturing is applied to derive higher convolutional rates [2].

Regarding convolutional decoding, it is usually implemented using the Viterbi algorithm.

In this section, we use the derivation for distribution of error events obtained in these convolutional codes at the output of the Viterbi decoder. We estimate the position and the length of bit errors at the output of the decoder with this method. We use asymptotic bounds to analyze the distribution of error event lengths at the output the Viterbi decoder. We also consider the relationship between the error probability of a random convolutional code and the error probability of a particular block code (termed *code termination* technique is presented in [83]). The tail of the distribution that is otherwise difficult to estimate with classical techniques can be estimated with this method.

Then, we use the error event length distribution and the distribution of errorless periods to derive a simple model which describes the residual error at the output of the soft decision Viterbi decoder. In the next section, we use this model to compute the distribution of corrupted bits for different transmission modes.

6.6.2 Video Encoder

The concept of *fine grain scalability* (FGS) has been introduced in order to allow for dynamic rate adaptation to varying bandwidth and receiver capabilities. Compression solutions based on motion-compensated spatio-temporal signal decomposition have thus gained attention as viable alternatives to classical predictive techniques for scalable video representation. The video codec that has been used in the experiments reported here, referred to as WAVIX in the sequel, has been developed in this framework. Figure 6.14 shows the structure of WAVIX video encoder.

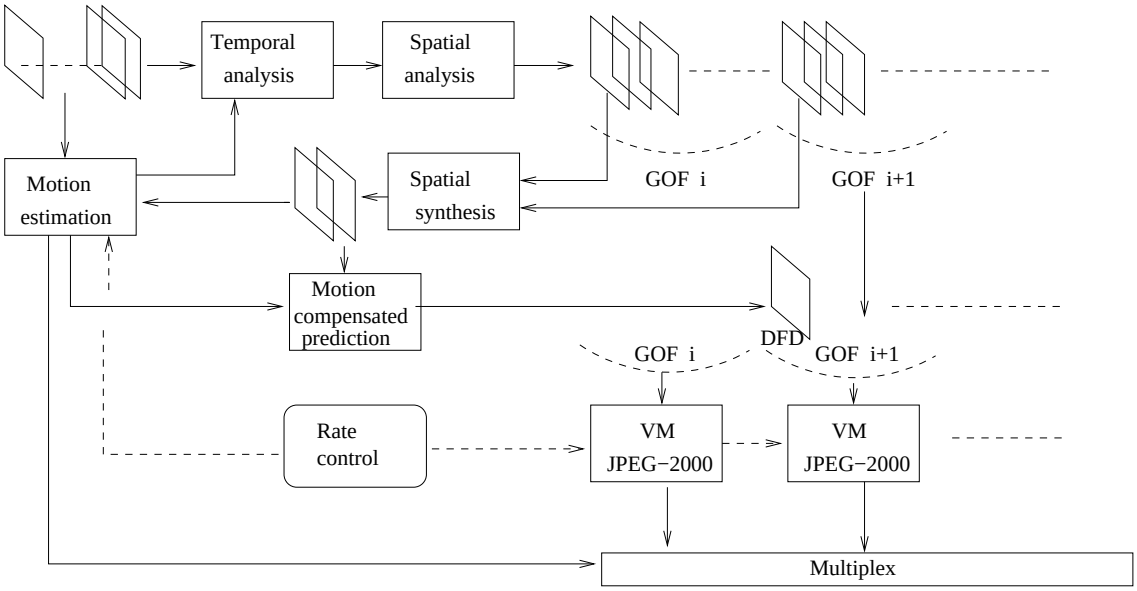


Figure 6.14: WAVIX structure

A *group of frames* (GOF) is fed into the coding system. In order to fine tune the bit rate allocated to the motion fields, the block matching motion estimation makes use of a rate-constrained adaptive tree structure. The block size is thus adapted to local motion characteristics in a rate-distortion sense. The rate here refers to the bit rate allocated to encode the motion vectors and the distortion relates to the prediction error. The estimation itself, to save computation time, relies on a hierarchical approach. The motion vectors obtained in the first steps of the quadtree are used to initialize the search in the subsequent steps. The motion vectors are then predictively coded. The predictor is given by the median value of neighboring vectors. The prediction error is then coded using Huffman codes.

The GOF is fed to the motion-compensated temporal transform which is based on a two-taps Haar wavelet transform. The temporal decomposition is applied iteratively on pairs of images within the GOF. The advantage of the Haar wavelet transform is to achieve a fairly good temporal energy compaction with a limited number of motion fields (8 motion fields for a 3-stage temporal decomposition of a group of 8 images). Each temporal subband is then further decomposed by a bi-orthogonal 9-7 wavelet filter in the horizontal and vertical direction. In the experiments, 3 levels decompositions are being used. The subbands resulting from the spatio-temporal decomposition are then quantized with a uniform quantizer and encoded with a context-based bit plane arithmetic coding as used in the JPEG-2000 standard [84]. The algorithm optimizing the truncation points in a rate-distortion sense handles groups of spatio-temporal subbands. The truncation point rate-distortion optimization leading to quality layers is well suited to fine tune the rate allocated to the texture information, hence to support fine grain scalability.

An inter-GOF temporal prediction is also used as an option in the above coding system. The inter-GOF temporal prediction leads to GOFs of type INTRA and of type INTER. The inter-GOF temporal prediction requires one additional motion field. This temporal prediction and corresponding motion estimation are realized in a closed loop. The closed-loop prediction is done by taking as reference information the corresponding image coded at a lower rate, as used in a base layer of a scalable representation. A more detailed description of this video codec can be found in [85].

Arithmetic codes are widely used in coding systems due to their high compression efficiency. They are however very sensitive to bit errors. A single bit error may lead to a complete de-synchronization of the decoder. In order to make the WAVIX codec robust to errors, an error-resilient arithmetic codes decoding technique [86] has been integrated in the video decoder. The technique consists in exploiting the residual redundancy in the bitstream by using soft-decision decoding procedures. The term *soft* here means that the decoder takes in input and supplies not only binary (*hard*) decisions but also a measure of confidence (a probability) on the bits. One can thus exploit the so-called *excess-rate* (or sub-optimality of the code), to reduce

the catastrophic *de-synchronization* effect of VLC decoders, hence to reduce the residual symbol error rates. This amounts to exploiting inner codeword redundancy as well as the remaining correlation within the sequence of symbols (remaining inter-symbol dependency).

In practice, the decoding algorithm can be regarded as a soft-input soft-output sequential decoding technique run on a tree. The complexity of the underlying Bayesian estimation algorithm growing exponentially with the number of coded symbols, a simple, yet efficient, pruning method is integrated. It allows the user to limit the complexity within a tractable and realistic range, at a limited cost in terms of estimation accuracy.

In order to increase the re-synchronization capability, a *soft synchronization* mechanism has been added. This mechanism relies on both the use of *soft synchronization* markers and of forbidden symbols. The *soft synchronization* markers are patterns, inserted in the symbol stream at some known positions, which serve as anchors for favoring the likelihood of correctly synchronized decoding paths. This *soft synchronization* idea augments the auto-synchronization power of the chain at a controllable loss in information rate. The forbidden symbols, when used, provide additional error detection and correction capability [87].

The bitstream generated by WAVIX is split into motion vectors and texture. The texture is encoded with the EBCOT algorithm. Hence it has the same properties as a JPEG 2000 bitstream. The corresponding bitstream is separated into header and entropy coded data. The header contains high level information, like GOF sizes, and provides a description of the structure of the entropy coded data. As this information is essential to the decoder, it is protected by a Reed-Solomon block code with high redundancy (127/255 for instance).

6.6.3 Multimedia Transmission over 802.11a Wireless Channel

In this section, we evaluate the quality of the video bitstream at the receiver side when the media-oriented mechanism is used. In our experiments, the WAVIX video encoder is configured to encode a sample of 300 CIF video frames. The video bit rate is 2 Mbps and each frame is a YUV image[3]. The number of frames in each GOF is 8. The activation of the WAVIX error resilience options corresponds to the addition of a 127/255 Reed-Solomon block code for header protection and of synchronization markers as explained in Section 6.6.2. The overhead of the header protection represents about 5.2% of the video stream while the overhead of the synchronization markers is negligible.

The transmission delay is calculated by considering the number of retransmissions and the value of the backoff timer [1]. The retransmission limit is defined in the IEEE 802.11 MAC standard specification with the help of the two following counters: The *short retry count*(SRC) and the *long retry count* (LRC). These counters are incremented and reset independently. The

[3]The foreman CIF (352$\times$ 288 pixels) video sequence has been used.

SRC is incremented every time an RTS fails and LRC is incremented when data transmission fails. Both the SRC and the LRC are reset to 0 after a successful data transmission. Data frames are discarded when SRC (LRC) reaches dot11ShortRetryLimit (dot11LongRetryLimit). The default values for dot11ShortRetryLimit and dot11LongRetryLimit are 7 and 4 respectively.

Furthermore, we consider the backoff timer period after each retransmission. For each retransmission, we select a random backoff which is drawn from a uniform distribution over the interval $[0, CW]$. In each retransmission, CW is updated to either $2 \times (CW + 1) - 1$ or its maximal value aCW_{max} (see Section 2.4). Let $\overline{T}_{backoff}(i)$ denote the average backoff interval after i consecutive unsuccessful transmission attempts. It can be calculated by [60]:

$$\overline{T}_{backoff}(i) = \begin{cases} \frac{2^i(aCW_{min}+1)-1}{2} \times aSlotTime & 0 \leq i \leq 6 \\ \frac{aCW_{max}}{2} \times aSlotTime & i \geq 6 \end{cases} \tag{6.2}$$

Where aCW_{min}, aCW_{max} and $aSlotTime$ are 15, 1023 and 9 μs for IEEE 802.11a WLANs [2]. We have chosen 4 SNR corresponding to 4 different transmission modes (see Table 6.2). Using the 802.11a channel model described in Section 6.6.1, we can find the distribution of bit errors for each SNR and transmission mode at the output of Viterbi decoder. The bit errors are distributed over the packets of length 1000 Bytes.

In the standard transmission mode, we only accept packets without corrupted bits. The error resilience options of the application layer are not employed for the standard transmission mechanism. However, we activate the WAVIX error resilience options and we accept packets with corrupted payload for the media-oriented mode selection mechanism.

Figures 6.15-6.18 show the PSNR, transmission delay, and interval jitter performance for 4 transmission modes with both the standard and the media-oriented mechanisms. Table 6.3 also shows the overall duration of the transmission for this video stream. As expected, the media-oriented mechanism (with LT = 0.1% and 5.2% FEC overhead at the application layer) significantly decreases the overall duration of the transmission (see Table 6.3).

Table 6.3: Transmission time comparison for video transmission with and without media-oriented mechanism

Modulation	*Data Rate (Mbps)*	*FEC Rate*	*SNR (dB)*	*Transmission duration for Standard (Sec)*	*Transmission duration for Media-oriented (Sec)*
BPSK	6	1/2	-1.6	8.00	6.92
QPSK	12	1/2	1.3	4.14	3.57
16-QAM	36	3/4	8.5	1.09	0.96
64-QAM	54	3/4	17.3	0.81	0.72

We made the following observations from Figures 6.15-6.18. The packet transmission time is almost fixed with the media-oriented mechanism while it continuously changes with the number of retransmissions using the standard mechanism. When the media-oriented mechanism is used, the PSNR of the decoded video is equivalent to the standard transmission mode, except for the *drops* that correspond to GOFs where errors occur. In this case, error resilience options allow us to decode the GOFs with the best achievable visual quality. The corrupted frames exhibit a lower quality, but their visual content is preserved. When the PSNR remains above 30 dB, the degradation is generally unnoticeable for a human viewer. When the PSNR falls as low as 25 dB, the decoded frames are severely degraded but still acceptable by a human viewer. The impact of errors on the visual quality depends on the characteristics of the current frame (in particular, the number and positions of errors, and the video content). In applications involving real time constraints, as for instance visiophony or streaming, it may be preferable to receive a degraded frame rather than losing it entirely or slowing down the video playback because of packets retransmission.

Another observation from the PSNR calculation is that after 4 consecutive retransmissions, (i.e. when a packet is lost for good), the standard transmission mechanism can not decode the rest of the video frame (e.g., this occurs at the frame number 220 in the Figure 6.15). However, this problem can be solved at the transmitter side with a more intelligent packetization scheme, or by adding resynchronisation patterns within the data flow. Nonetheless, in case of packet drop, the visual content of a full GOF may be lost.

Figures 6.15-6.18 also show the jitter for the standard and the media-oriented mode selection mechanisms. It is obviously and logically correlated to transmission delay. In the media-oriented mechanism, the jitter is much less important than with the standard mode. This is a very desirable property in the case of video transmission. Having a constant time interval between packets arrivals is equivalent to having a constant time slot available to decode each GOF. Therefore, complexity can be managed easily without the need for excessive buffering.

We have simulated the same scenarios for 10 different channel characteristics (different distributions of corrupted bits over data flow) in order to calculate the confidence interval of the PSNR with the media-oriented transmission mode. For each transmission rate, the 95% confidence intervals on the mean PSNR are computed. The intervals for the various rates are displayed by horizontal lines as shown in Figure 6.19. The results show an acceptable PSNR in all transmission modes. Figure 6.20 shows two samples of video streams transmitted with the media-oriented algorithm at 12 Mbps.

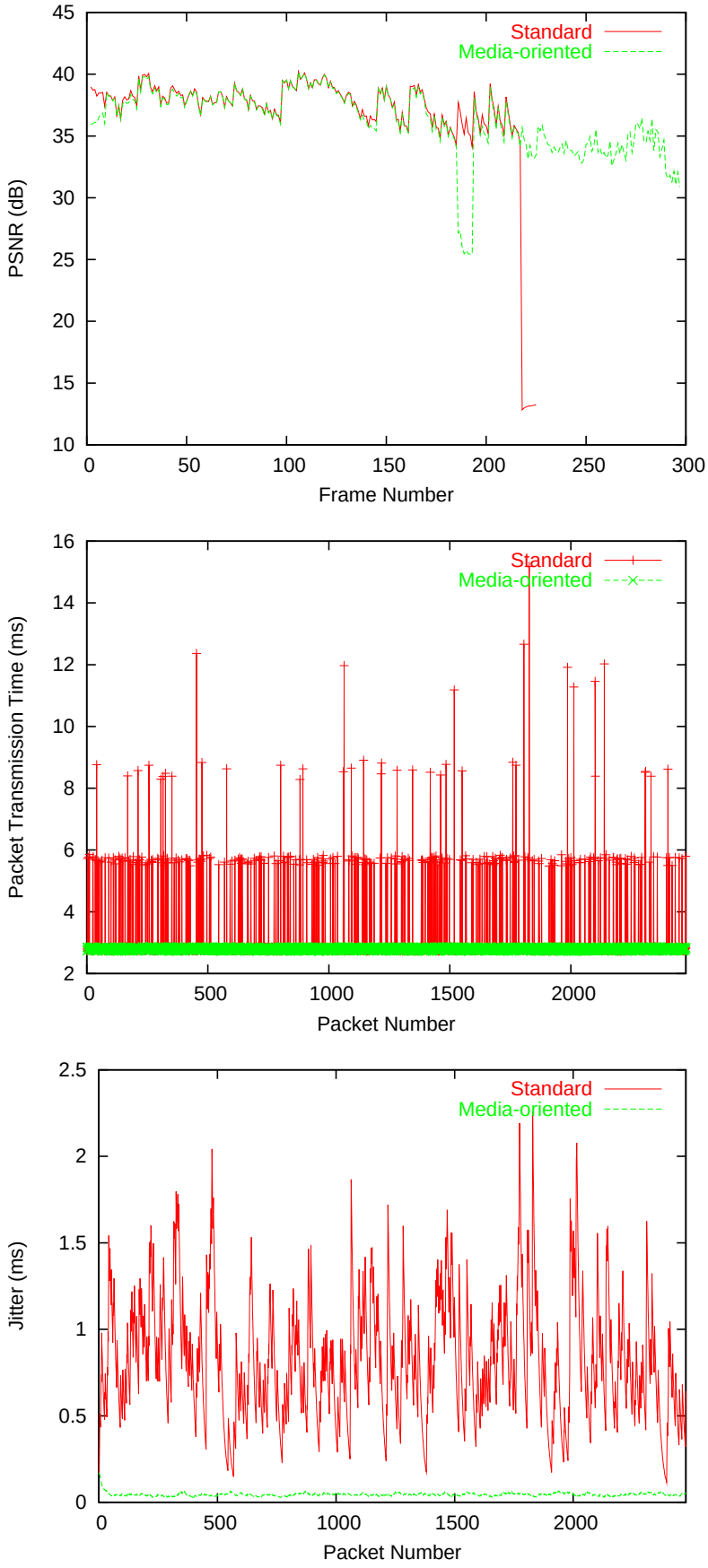


Figure 6.15: PSNR, transmission delay, and jitter comparison (SNR = −1.6dB, 6 Mbps, FEC=1/2, BPSK).

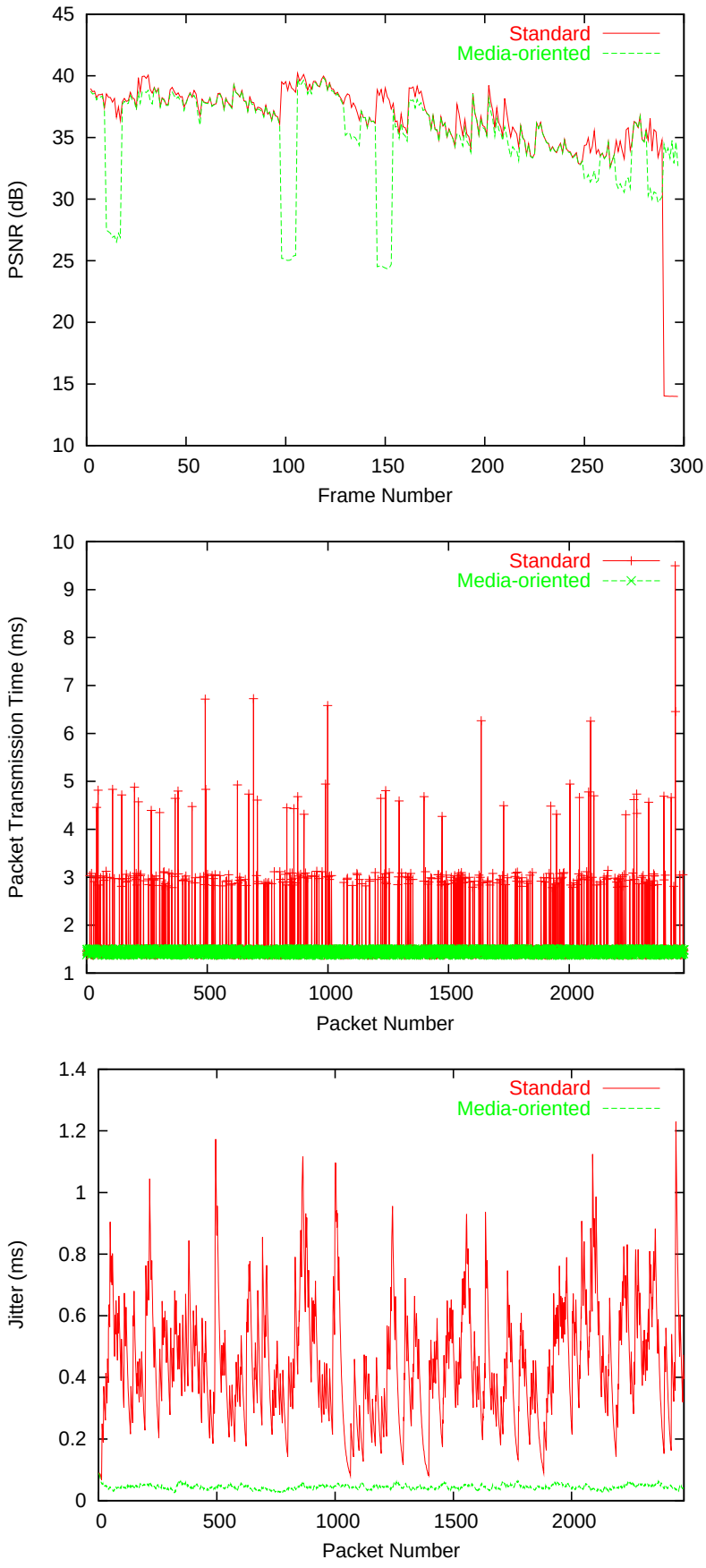


Figure 6.16: PSNR, transmission delay, and jitter comparison (SNR = 1.3dB, 12 Mbps, FEC=1/2, QPSK).

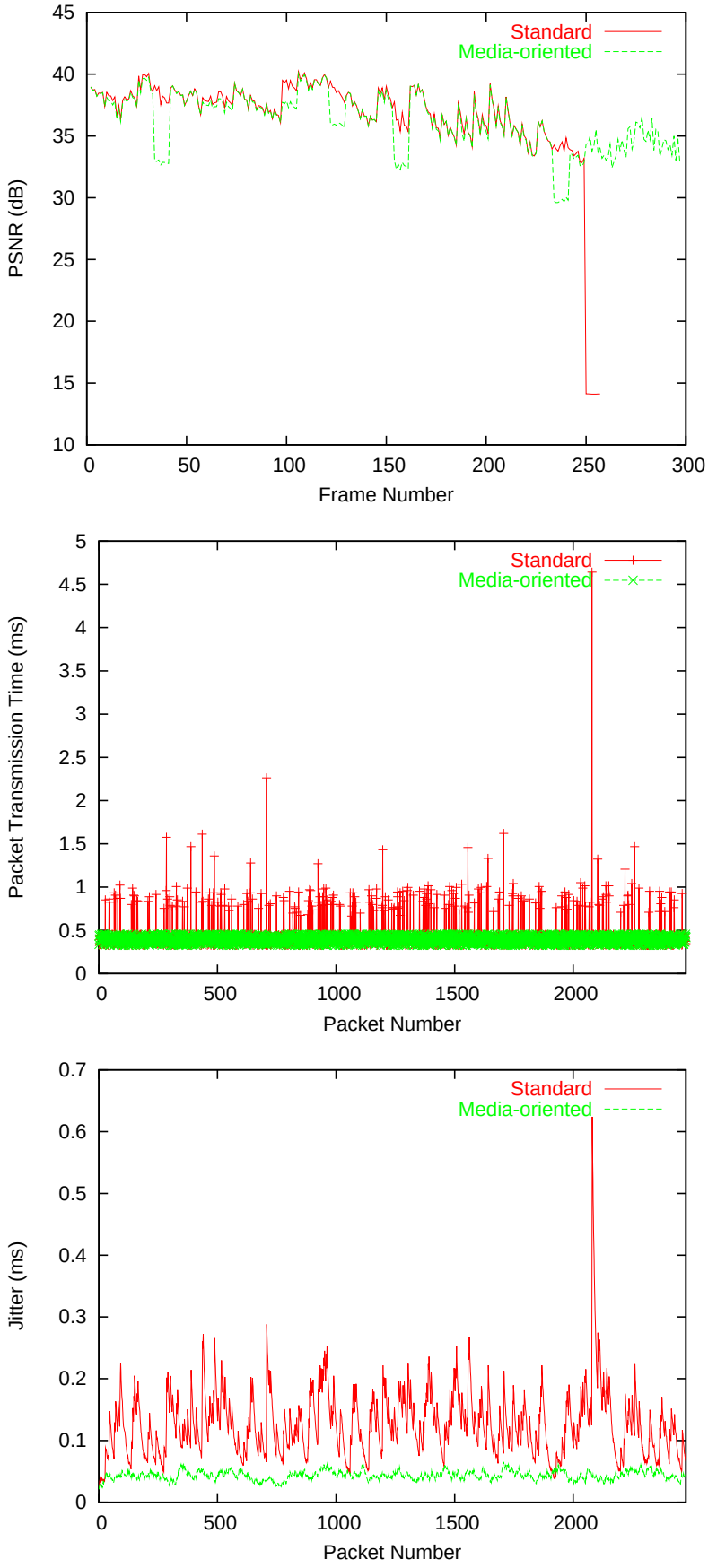


Figure 6.17: PSNR, transmission delay, and jitter comparison (SNR = 8.5dB, 36 Mbps, FEC=3/4, 16-QAM).

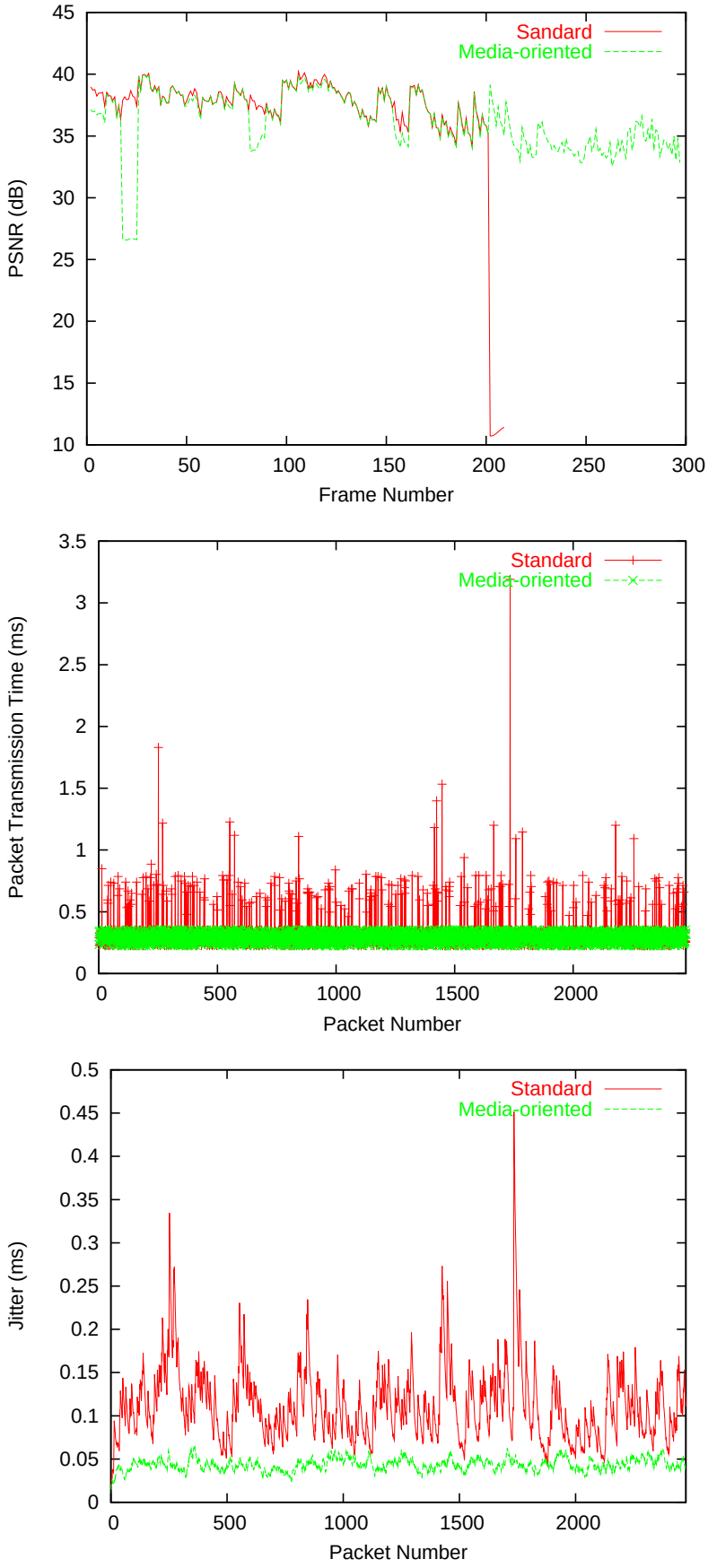


Figure 6.18: PSNR, transmission delay, and jitter comparison (SNR = 17.3dB, 54 Mbps, FEC=3/4, 64-QAM).

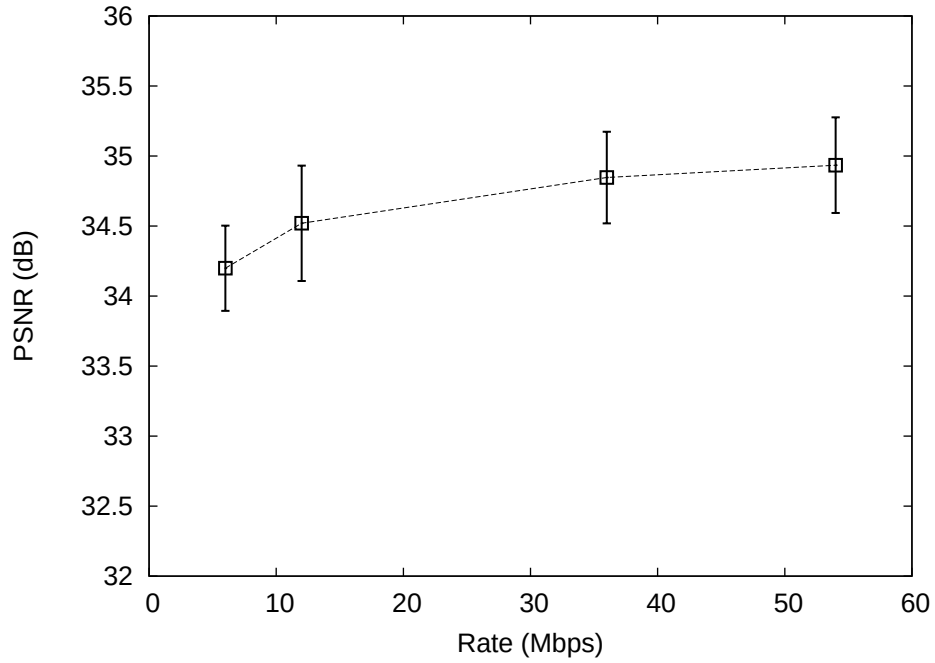


Figure 6.19: 95% confidence intervals of PSNR for different transmission modes with media-oriented mode selection mechanism.

6.7 Conclusions

In this chapter, we have presented a novel cross layer mechanism in MANETs to select the best transmission mode which takes into account some characteristics of the application. This mechanism, which we believe to be easy to implement in actual devices, uses information from the physical channel and the loss tolerance requirements of the application to select the optimal PHY rate, modulation and FEC transmission parameters. We have proposed new transmission modes which do not use physical level FEC and which significantly increase the application throughput. NS-based simulation results in ad hoc networks show that our mechanism achieves up to 4 Mbps increase in throughput in MANETs. The gain obtained from the application point of view has been evaluated with the help of the WAVIX video encoder, which can tolerate a BER equal to 10^{-3} with only 5% of FEC overhead at the application level. The results show significant improvements in throughput, latency and jitter.

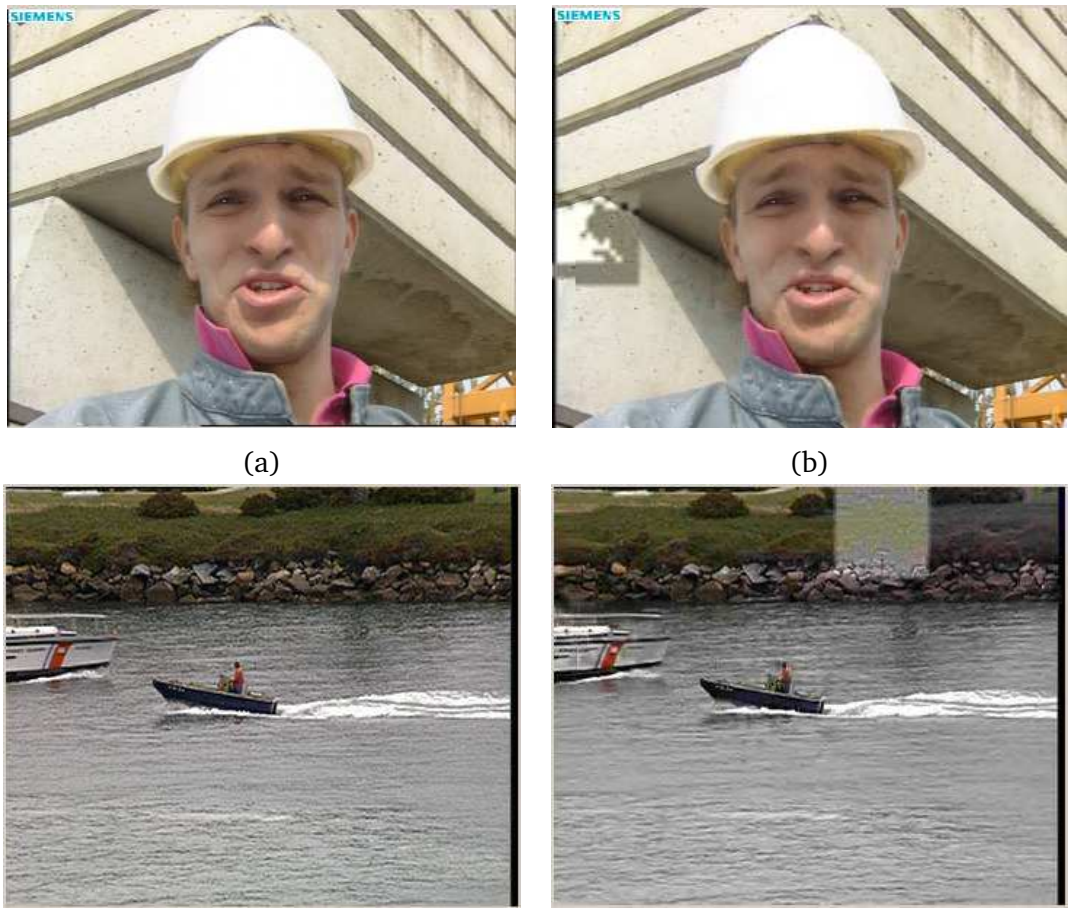


Figure 6.20: The samples of video stream at the receiver, (b) and (d) are transmitted using the Media-oriented algorithm (accepting the packets with 0.1% bit errors, SNR=1.3, and Rate = 12 Mbps), (a) and (c) are original video streams.

7

CONCLUSIONS

7.1 Summary of Contributions

This dissertation has examined the design of several cross layer interaction mechanisms and adaptive communications for IEEE 802.11 wireless LANs. The ideas presented in this dissertation can significantly improve the performance of IEEE 802.11 wireless LANs. In this section we summarize the contributions of this research work.

In Chapter 2, we presented an overview of 802.11 CSMA/CA MAC layer, 802.11e MAC layer, and several transmission techniques defined by 802.11 standard specifications including IR, DSSS, and OFDM. We then presented a detailed evaluation of physical transmission modes for 802.11a/b in Chapter 3. The goals we seek to address in this chapter are: the effect of wireless channel characteristics (i.e., AWGN versus fading, and multipath channels), modulations, multipath receiver structure, binary convolutional coding, Viterbi decoding, and physical data rate on the performance of 802.11 devices.

In Chapter 4 we presented DAW, a distance aware model which allows modelling the performance of the 802.11 devices considering both MAC and PHY layer characteristics. This model is based on Bianchi's model for CSMA/CA 802.11 MAC layer. DAW considers the interference and background noise to compute the packet loss probability, hence taking into account the capture effect [31]. We computed the performance of 802.11 WLANs in fixed and random topologies scenarios using the DAW model. Using NS-2 simulations, we validated our model and showed that DAW obtains more realistic results compared to the models that only consider the 802.11 MAC layer.

Chapter 5 concerned physical rate selection mechanisms. In this chapter, the large body of

related works for rate selection mechanisms are surveyed. We presented the key ideas of each algorithm along with its pros and cons. We also pointed the key parameters to design physical rate selection mechanisms. We then described our solutions named, AARF, CLARA, and AMRR and provided a performance analysis. The key contributions of these algorithms are:

- AARF and AMRR, designed for low latency and high latency communication systems respectively, both perform close to the optimum case considering the simulation and experimental results.
- AMRR is implemented in a linux driver for AR5212-based devices. The implementation showed that AMRR improves the performance and can be easily implemented in existing devices.
- CLARA is a smart rate selection mechanism that can adaptively select the best transmission mode at any given time based on measured and estimated system and channel parameters.
- CLARA could be implemented at the PHY layer and so it is MAC independent. Therefore it can be implemented in 802.11e mobile STAs as well.

In Chapter 6 we presented MORSA, a media oriented rate selection algorithm that takes into account the application characteristics to select the physical transmission rate. We addressed the issues concerning its implementation on an 802.11e station. We showed that the performance of the mobile ad hoc networks can be significantly increased using this simple cross-layer mechanism. We also evaluated the quality of video application using the WAVIX encoder when it employs the MORSA mechanism. We showed that the application can obtain significant improvements in throughput, latency, and jitter when it uses the MORSA.

7.2 Future Works

At the time of this writing, cross layer interaction mechanisms in WLANs are being actively addressed in the computer network research areas. There are some standardization efforts in progress at BRAN HiperLAN2, 3GPP (High Speed Downlink Packet Access), and IEEE as addressed in Section 6.3. Most of the research works presented in this thesis are focused on cross layer interaction mechanisms and related topics. Following we address possible future works in several research areas presented in this thesis.

Future works on the cross layer modelling will first involve proposing a more precise model. This could be the integration of fading and multipath wireless channel models into the DAW model or evaluation of the model with different random topology distribution. We can also

calculate the packet delay. Another modification could be the duplex link modelling (i.e., uplink and downlink connections). This model could be helpful to elaborate many topics related to ad hoc networks as well. For example the capacity, connectivity, power control, and pricing of ad hoc networks can be evaluated more precisely using our model since it considers both 802.11 MAC and PHY layers. DAW model can be potentially used to design the WLANs, determining the optimal placement of access points with some elaborations. This aims providing the maximum coverage for WLAN service areas as well as maximum overall throughput. We can also evaluate the performance of different application (e.g, TCP) over 802.11 wireless LANs using the DAW model.

Future works on physical rate adaptation algorithm can involve evaluating the proposed mechanisms over multihop connections. Generally, the physical rate adaptation mechanisms are proposed and evaluated for single connection and single hop scenarios. However, the real network topologies consist several connections and multihop. At this time, we are not aware of any research work that takes into account the interactions between several connections to select the best physical transmission rate.

Another issue with physical rate adaptation mechanisms is the frequency allocation and power control. As we addressed in Chapter 5, proposing a joint frequency/power allocation and rate selection mechanisms will be a hot topic, particularly in ad hoc networks. Evaluation of rate adaptation mechanisms over MIMO receiver structures and sensor networks could be interesting as well. Finally, the implementation issues for CLARA should be investigated with more detailed. We can also propose hybrid rate selection mechanisms using AARF and CLARA for different channel characteristics similar to HRC [47, 48]. We can also obtain better performance using simple and efficient algorithms like FAR [45] along with AARF and CLARA, to select the data rate for control frames (i.e., RTS, CTS, and ACK).

Regarding to adaptive communications and MORSA mechanism, the future works can involve evaluating the mechanism with other applications like voice over IP or different video encoders. For example, the quality of corrupted audio flows could be assessed using the E-Model. Since MORSA encourages the users to send with higher data rates, it will decrease the effects of the time unfairness in the multirate network scenarios [54]. This issue can be investigated for MORSA in the future works. A real implementation of MORSA over a *software defined radio* (SDR) platform could be performed and evaluated as well. This can be performed by 802.11a SDRs devices provided by Cisco [88].

A

IEEE 802.11 WORKING GROUP

The 802.11™ *working group* (WG) is responsible for developing *wireless local area network* (WLAN) standards under the auspices of the IEEE® Project 802 LAN/MAN Standards Committee (802 LMSC). The 802.11™ WG is one of several WGs reporting to the 802 LMSC. The 802.11 WG operates under the Policies and Procedures of IEEE® Project 802, LAN/MAN Standards Committee and the hierarchy of rules under which IEEE® Project 802 operates. Table A.1 shows the list of all 802.11 WGs and their tasks.

Table A.1: IEEE 802.11 working groups.

IEEE 802.11 working group	*Tasks (Functionalities)*
IEEE 802.11	The original 1 Mbit/s and 2 Mbit/s 2.4 GHz RF and IR standard (1999)
IEEE 802.11a	54 Mbit/s, 5 GHz Standard (1999), Shipping products in (2001)
IEEE 802.11b	Enhancements to 802.11 Support 5.5 and 11 Mbit/s (1999)
IEEE 802.11c	Bridge operation procedures Included in the IEEE 802.1D standard (2001)
IEEE 802.11d	International (country-to-country) Roaming extensions (2001)
IEEE 802.11e	Enhancements: QoS, Including packet bursting
IEEE 802.11f	Inter-Access Point Protocol (2003)
IEEE 802.11g	54 Mbit/s, 2.4 GHz backwards compatible with b (2003)
IEEE 802.11h	5 GHz spectrum, dynamic channel/frequency selection (DCS/DFS) and transmit power control (TPC) for European compatibility (2004)
IEEE 802.11i	Enhanced security (2004)
IEEE 802.11j	Extensions for Japan (2004)
IEEE 802.11k	Radio resource measurement enhancements
IEEE 802.11m	Maintenance of the standard: odds and ends
IEEE 802.11n	Higher throughput improvements
IEEE 802.11p	WAVE - Wireless Access for the Vehicular Environment
IEEE 802.11r	Fast roaming
IEEE 802.11s	ESS Mesh Networking
IEEE 802.11T	Wireless performance prediction (WPP) test methods and metrics
IEEE 802.11u	Interworking with non-802 networks (e.g., cellular)
IEEE 802.11v	Wireless network management
IEEE 802.11w	Protected Management Frames

B

CHANNELS FOR 802.11A AND 802.11B

The channel identifiers, channel center frequencies, and regulatory domains of each IEEE 802.11a 20-MHz-wide channel are listed in Table B.1. All channel sets are restricted to indoor usage except the Americas, which allows for indoor and outdoor use on channels 52 through 64 in the United States [2].

The channel identifiers, channel center frequencies, and regulatory domains of each IEEE 802.11b 22-MHz-wide channel are listed in Table B.2. France is included in the European regulatory domain; however, only channels 10 through 13 can be used in France. Users are responsible for ensuring that the channel set configuration complies with the regulatory standards of France.

Table B.1: Channels for IEEE 802.11a.

Channel Identifier	*Frequency (MHz)*	*America*	*Japan*
34	5170	–	×
36	5180	×	–
38	5190	–	×
40	5200	×	–
42	5210	–	×
44	5220	×	–
46	5230	-	×
48	5240	×	–
52	5260	×	–
56	5280	×	–
60	5300	×	–
64	5320	×	–
149	5745	–	–
153	5765	–	–
157	5785	–	–
161	5805	–	–

Table B.2: Channels for IEEE 802.11b.

Channel Identifier	*Frequency (MHz)*	*America*	*Japan*	*Europe*
1	2412	×	×	×
2	2417	×	×	×
3	2422	×	×	×
4	2427	×	×	×
5	2432	×	×	×
6	2437	×	×	×
7	2442	×	×	×
8	2447	×	×	×
9	2452	×	×	×
10	2457	×	×	×
11	2462	×	×	×
12	2467	–	×	×
13	2472	–	×	×
14	2484	–	×	–

C

AARF Pseudo Code

The following variables are defined:

- *timer*: incremented for each packet transmitted (regardless of success or failure). Reset to zero when two or more consecutive retries are done or when 10 consecutive packets have been successfully transmitted.
- *success*: number of consecutive successful packet transmissions.
- *recovery*: if it is set to TRUE, it means that we have just sent a probing packet to try a higher rate. Otherwise, it is set to FALSE.
- *retry*: number of consecutive retries for a given packet.
- *success_threshold*: number of consecutive successful transmissions required to send a probe packet (to try a higher rate).
- *timeout*: if *timer* reaches this value, a probe packet is sent.
- *retry_threshold*: number of consecutive failures for one packet required to drop this packet.

- *min_success_threshold*: value used to initialize the variable *success_threshold* when the rate is decreased because of two consecutive transmission failures.
- *max_success_threshold*: maximum possible value of *success_threshold*.
- *success_k*: multiplicative factor used to calculate the new value of *success_threshold* when the probe packet fails.
- *min_timeout*: minimum value of *timeout*.
- *timeout_k*: in AARF, *timeout* is set to either *min_timeout* or *success_threshold* $\times$ *timeout_k*.

```
static int g_success = 0; static int g_timer = 0; static int
g_recovery = 0; static int g_success_threshold = 10; static int
g_timeout = 15; static int g_retry_threshold = 4;

#ifdef AARF static int g_min_success_threshold = 10; static int
g_max_success_threshold = 50; static int g_min_timeout = 15;
static double g_success_k = 2; static double g_timeout_k = 1.5;

#define report_recovery_failure() \ { \
  g_success_threshold = min (g_success_threshold *
  g_success_k, g_max_success_threshold); \
  g_timeout = max (g_timeout_k *
  g_success_threshold, g_min_timeout); \
}

#define report_failure() \ { \
  g_success_threshold = g_min_success_threshold; \
  g_timeout = g_min_timeout; \
} #else /* AARF */ #define report_recovery_failure() #define
report_failure() #endif /* AARF */

int send_one_packet (Packet packet, int rate) {
  int retry = 0;
  Status status;
```

```
while (retry < g_retry_threshold) {
  status = send_packet (packet, rate);
  if (status == SUCCESS) {
    g_success++;
    retry = 0;
    if ((g_success == g_success_threshold
         || g_timer == g_timeout)
        && !is_max_rate (rate)) {
      rate = increment_rate (rate);
      g_timer = 0;
      g_success = 0;
      recovery = TRUE;
    } else {
      g_timer++;
      g_recovery = FALSE;
    }
    break;
  } else {
    g_timer++;
    retry++;
    g_success = 0;
    if (g_recovery) {
      g_timer = 0;
      if (retry == 1) {
        report_recovery_failure ();
        if (!is_min_rate (rate)) {
          rate = decrement_rate (rate);
        }
      }
    } else {
      if (retry == 2
          || retry == 4
          || retry == 6
          || retry == 8
```

```
            || retry == 10) {
          report_failure ();
          if (!is_min_rate ()) {
            rate = decrement_rate (rate);
          }
        }
      }
      if (retry >= 2) {
        g_timer = 0;
      }
    }
  }
  return rate;
}
```

D

AMRR Pseudo Code

All variables used in the AMRR Pseudo Code have already been defined for the AARF Pseudo Code. The following functions are used:

- *is_success*: returns TRUE if less than 10% of the packet transmissions failed during the previous period, FALSE otherwise.
- *is_failure* returns TRUE if more than 33% of the packet transmissions failed during the previous period, FALSE otherwise.
- *is_enough*: returns whether or not enough packets were transmitted during the previous period to get significative statistics. By default, *is_enough* returns TRUE if the transmission of 10 distinct packets was attempted during the previous period, FALSE otherwise.

```
void update_timer (void) {
  int need_change = FALSE;
  if (is_success () && is_enough ()) {
    g_success++;
    if (g_success >= g_success_threshold
        && !is_max_rate ()) {
```

```
      g_recovery = TRUE;
      g_success = 0;
      increase_rate ();
      need_change = true;
    } else {
      g_recovery = FALSE;
    }
  } else if (is_failure ()) {
    g_success = 0;
    g_recovery = FALSE;
    if (!is_min_rate ()) {
      if (m_recovery) {
        m_success_threshold *= 2;
        m_success_threshold = min (m_success_threshold,
        m_max_success_threshold);
      } else {
        m_success_threshold = m_min_success_threshold;
      }
      decrease_rate ();
      need_change = true;
    }
  }
  if (is_enough () || need_change) {
    reset_cnt ();
  }
}
```

E

List of Acronyms

AC	Access Category
ACK	ACKnowledgment
ADSL	Asymmetrical Digital Subscriber Loop
AIFS	Arbitration Inter-Frame Spacing
AP	Access Point
AARF	Adaptive Auto Rate Fallback
AMRR	Adaptive Multi Rate Retry
ARF	Auto Rate Fallback
ASRC	Adaptive Source Rate Control
AWGN	Additive White Gaussian Noise
BE	Best Effort
BEB	Binary Exponential Backoff
BER	Bit Error Rate
BK	Background
BPSK	Binary Phase Shift Keying
BS	Base Station

BSA	Basic Service Area
BSS	Basic Service Set
CA	Collision avoidance
CBR	Constant Bit Rate
CCK	Complementary Code Keying
CD	Collision Detection
CDMA	Code Division Multiple Access
CFP	Contention Free Period
CIR	Channel Impulse Response
CL	Controlled Load
CL	Convergence Layer
CLARA	Closed-Loop Adaptive Rate Allocation
CP	Contention Period
CRC	Cyclic Redundancy Check
CS	Carrier Sense
CSMA	Carrier Sense Multiple Access
CTS	Clear To Send
CW	Contention Window
DAW	Distance AWare
DBPSK	Differential Binary Phase Shift Keying
DE-BPSK	Differentially Encoded BPSK
DE-QPSK	Differentially Encoded QPSK
DCF	Distributed Coordination Function
DFE	Decision Feedback Equalizer
DFWMAC	Distributed Foundation Wireless Medium Access Control
DIFS	DCF IFS
DLC	Data Link Control
DQPSK	Differential Quadrature Phase Shift Keying
DS	Direct Sequence
DS	Distribution System

DSSS	Direct Sequence Spread Spectrum
DSR	Dynamic Source Routing
DMT	Discrete Multi Tone
EC	Error Control
EE	Excellent Effort
EDCA	Enhanced Distributed Coordination Access
ERP	Extended Rate PHY layer
ESS	Extended Service System
FAR	Full Auto Rate
FCH	Frame Control Channel
FCS	Frame Check Sequence
FDD	Frequency Division Duplex
FEC	Forward Error Correction
FFT	Fast Fourier Transform
FGS	Fine Grain Scalability
FHSS	Frequency Hopping Spread Spectrum
FIFOs	First In First Out queue
FSK	Frequency Shift Keying
GFSK	Gaussian FSK
GI	Guard Interval
GOF	Group of Frames
GRTS	Group RTS
HAL	Hardware Abstraction Layer
HC	Hybrid Coordinator
HCF	Hybrid Coordination Function
HCCA	HCF Controlled Channel Access
HiperLAN	High Performance European Radio LAN
H-MORSA	Hybrid MORSA
HRC	Hybrid Rate Control
IBSS	Independent BSS

ID	IDentification
IEEE	Institute of Electrical and Electronics Engineers
IETF	Internet Engineering Task Force
IFFT	Inverse FFT
IFS	Inter Frame Spacing
IP	Internet Protocol
IR	Infra-Red
ISI	Inter-Symbol Interference
ISO	International Organization for Standardization
ISP	Internet Service Provider
LA	Link Adaptation
LAN	Local Area Network
LE	Linear Equalizer
LOS	Line Of Sight
LRC	Long Retry Count
LT	Loss Tolerant
MAC	Medium Access Control
MAD	Medium Access Diversity
MACA	Multiple Access with Collision Avoidance
MACAW	MACA for Wireless networks
MBWA	Mobile Broadband Wireless Access
MC-CDMA	Multi Carrier CDMA
MGF	Moment Generating Function
MH	Mobile Host
MIMO	Multiple Input Multiple Output
MiSer	Minimum-energy Transmission Strategy
MLSE	Maximum Likelihood Sequence Estimator
MONET	MObile ad hoc NETwork
MORSA	Media Oriented Rate Selection Algorithm
MPDU	MAC Protocol Data Unit

MT	Mobile Terminal
MVCS	Modified Virtual Carrier Sensing
NACK	Negative ACKnowledgement
NAV	Network Allocation Vector
NC	Network Control
NS	Network Simulator
OAR	Opportunistic Auto Rate
OFDM	Orthogonal Frequency Division Multiplexing
OSI	Open System Interconnection
PAC	PAcket Concatenation
PAN	Personal Area Network
PCF	Point Coordination Function
PDA	Personal Digital Assistant
PDU	Packet Data Unit
PEP	Pairwise Error Probability
PER	Packet Error Rate
PHY	PHYsical layer
PIFS	PCF Interframe Space
PIN	Personal Identification Number
PLCP	Physical Layer Convergence Procedure
PMD	Physical Medium Dependent
PN	Pseudo random Noise code
PPDU	PHY Protocol Data Unit
PPM	Pulse Position Modulation
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QSTAs	QoS-enhanced Stations
QPSK	Quadrature Phase Shift Keying
RBAR	Receiver Based Auto Rate
RF	Radio Frequency

RLC	Radio Link Control
RSH	Reservation Subheader
RSS	Received Signal Strength
RTS	Request To Send
RTT	Round Trip Time
SDR	Software Defined Radio
SIFS	Short IFS
SNR	Signal to Noise Ratio
SNIR	Signal to Noise and Interference Ratio
SRC	Short Retry Count
SS	Spread Spectrum
SSIA	Strength Indication of the Acknowledge
SYNC	Synchronization
TBTT	Target Beacon Transition Time
TC	Traffic Category
TCP	Transport Control Protocol
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
TH	Time Hopping
TPC	Transmit Power Control
TXOP	Transmission Opportunity time
UDP	User Datagram Protocol
UL	UpLink
UP	User Priorities
VoD	Video on Demand
WDS	Wireless Distribution System
WG	Working Group
WLAN	Wireless Local Area Network
WT	Wireless Terminal
XOR	eXclusive OR

F

PRÉSENTATION DES TRAVAUX DE THÈSE EN FRANÇAIS

F.1 Introduction

Cette thèse a pour objectif l'étude des mécanismes d'interaction inter-couche et leur application pour des algorithmes adaptatifs pour contrôler les paramètres de transmission. A cette fin, nous nous sommes intéressés à l'analyse en utilisant des modèles analytiques des performances des couches MAC et PHY du standard IEEE 802.11. Nous proposons ensuite des mécanismes qui utilisent des interactions entre les différentes couches protocolaires pour améliorer les performances des terminaux 802.11. En particulier, nous proposons de nouveaux mécanismes de sélection du débit de transmission physique. Enfin, nous nous intéressés à l'amélioration des performances de la transmission vidéo dans les réseaux sans fil IEEE 802.11 en utilisant des techniques qui prennent en compte les informations des couches applicatives, MAC et PHY.

F.1.1 Réseaux Sans Fil

Les réseaux sans-fil sont basés sur l'utilisation de liens radio permettant à deux ou plusieurs ordinateurs d'échanger des informations. Les progrès récents et très rapides de ces technologies ont permit un déploiement rapide de ce type de réseaux à travers le monde et l'arrivée du "tout mobile".

F.1.2 Interactions Inter Couche

Les protocoles réseaux sont en général séparés en plusieurs couches: chaque couche est conçue indépendamment des autres et les interfaces entre ces couches sont spécifiées de façon très rigoureuse. Les interfaces entre les couches sont suffisamment flexibles pour qu'il soit possible de remplacer une nouvelle version d'une couche réseau sans pour autant changer l'intégralité du système. Evidemment, ce qui fait la grande force de cette approche, c'est à dire l'indépendance des couches les unes des autres, est aussi sa plus grande faiblesse puisqu'il devient très difficile de gérer les tâches qui requièrent une interaction forte entre plusieurs couches.

Les réseaux sans fil sont particulièrement sujets à ce genre de problème: le lien de communication physique ne peut plus être considéré comme une entité dont la performance est indépendante des autres couches réseaux. Par exemple, le choix d'un débit de transmission physique (en général effectué par la couche MAC d'après ses connaissances sur l'état du canal de transmission), a une influence importante sur le délai moyen des paquets de données sur ce lien radio. La décision de routage effectuée par les protocoles des couches supérieures se base souvent sur une évaluation du délai de transmission. Cette décision de routage va donc dépendre indirectement des conditions de transmission sur le canal physique: le comportement d'une couche affecte donc le comportement des autres couches protocolaires.

Malgré cette complexité supplémentaire, ce type de phénomène est en fait particulièrement intéressant puisqu'il constitue une opportunité supplémentaire d'améliorer les performances du système. En effet, il est possible de concevoir des mécanismes de contrôle des paramètres de

transmission prenant en compte l'état de l'ensemble des couches de la pile réseau. Cette thèse présente des mécanismes simples de contrôle du débit de transmission physique en utilisant des informations provenant des couches MAC et PHY. On montre que ces mécanismes permettent d'obtenir une très forte augmentation de performance.

Motivations et Contributions

La conception de mécanismes basés sur l'interaction entre différentes couches protocolaires requiert un modèle d'évaluation des performances du système, en particulier des couches MAC et PHY: nos modèles d'évaluation ont été conçus pour prendre en compte des canaux de transmission de type bruit blanc Gaussien, atténuation de Rayleigh, et atténuation sur de multiples chemins.

Modélisation des Couches MAC/PHY du Standard IEEE 802.11

Nous avons conçu un modèle analytique nommé "Distance Aware" (DAW) qui prend en compte la position des terminaux radio par rapport au point d'accès ainsi que les caractéristiques du canal de transmission [5]. Etant donné la position de l'ensemble des terminaux et du point d'accès, DAW est capable de calculer le débit de saturation de chaque terminal en fonction de sa position relative par rapport aux autres terminaux. Ce modèle peut être utilisé pour évaluer la performance et les capacités de réseaux existants mais il est aussi utile au dimensionnement de nouveau réseaux ad hoc.

Mécanismes de Sélection des Modes de Transmission dans le Standard IEEE 802.11

Nous avons présenté certains des paramètres du système réel qu'il est nécessaire de prendre en compte pour concevoir un algorithme de sélection de débit de transmission efficace. En particulier, nous avons présenté trois nouveaux algorithmes: adaptive auto rate fallback (AARF) [6], closed-loop adaptive rate allocation (CLARA) [7], et adaptive multi rate retry (AMRR) [6, 8].

AARF se caractérise par une complexité très faible est une performance proche des performances de systèmes beaucoup plus complexes tels que RBAR. CLARA reprend les aspects les plus intéressants des mécanismes ARF et RBAR et y ajoute des fonctionnalités supplémentaires afin d'améliorer la réponse aux problèmes: d'atténuation sur des chemins multiples, la détection

de canal, et l'échange de signaux de contrôle. AMRR apporte une approche plus pratique au problème puisqu'il a été mis en oeuvre sur des cartes utilisant le microprocesseur AR-5212.

F.1.3 Mécanismes Adaptatifs pour Transmission de Données

Le terme "Contrôle Adaptif" se réfère à un système de communication qui utilise de manière automatique les informations sur l'état du système pour ajuster son comportement. Ce type d'information qui peut être obtenu par l'intermédiaire de signaux transmis par le système, est utilisé pour modifier dynamiquement certains des paramètres de l'état du système. Un tel comportement permet d'augmenter la capacité du système à résister aux dégradations de l'environnement dans lequel il opère. Les paramètres du système modifié par ce mécanisme peuvent être continus ou discrets. Les systèmes de contrôle adaptif sont aujourd'hui très largement déployés dans certains types de réseaux sans fil. Les protocoles de routage pour les réseaux sans fil ad hoc ou encore les systèmes d'antennes directionnelles ou de modulation adaptives sont des exemples de ce type d'applications.

Motivations et Contributions

En général, les réseaux sans fil existants utilisent différents mécanismes de correction d'erreurs en fonction de la couche considérée. Par exemple, les codes correcteurs d'erreur (FEC) peuvent être utilisés à la fois au niveau physique mais aussi au niveau applicatif. Une interaction inter couche est nécessaire pour choisir le meilleur mécanisme de correction d'erreurs en fonction des caractéristiques des applications. Par exemple, si une application peut utiliser les paquets de données avec un certain pourcentage de bits erronés, elle peut désactiver le FEC dans la couche physique pour le remplacer par du FEC dans la couche applicative.

Transmission Multimédia sur réseaux sans fil IEEE 802.11

Dans cette thèse, nous présentons un mécanisme de sélection automatique de mode de transmission appelé MORSA pour des réseaux locaux sans fil 802.11 qui s'adapte à la fois aux conditions du canal et aux caractéristiques du média transmis [11, 12]. L'objectif de cette étude est d'améliorer la transmission de certaines applications multimédia qui sont robustes

aux erreurs de bits. En particulier, si la couche MAC est consciente du fait que l'application tolère un certain pourcentage d'erreurs de transmission, elle peut en profiter pour sélectionner un mode qui supporte un débit de transmission plus élevé. Nous comparons notre mécanisme avec d'autres algorithmes de sélection de débit par le biais du simulateur réseau NS-2. La transmission vidéo sur un modèle de canal pour le protocole IEEE 802.11a a été évaluée avec le mécanisme MORSA et comparée avec le mode standard. Les résultats montrent que le nouveau mécanisme permet d'augmenter jusqu'à 4 Mbps le débit reçu.

F.1.4 Organisation de la Thèse

La première partie de cette thèse s'intéresse à la conception et à l'analyse des performances de la transmission de données dans les réseaux sans fil IEEE 802.11. Nous présentons dans le chapitre 2 un résumé du contenu de la norme IEEE 802.11 pour les couches MAC et PHY ainsi que la nature des modulations utilisées par la couche physique. Dans le chapitre 3, nous analysons les performances de la transmission de données avec des modèles de canal de type AWGN, atténuation de Rayleigh, et atténuation sur des chemins multiples.

Dans la deuxième partie de cette thèse, nous présentons les différents modèles des couches 802.11 utilisés aujourd'hui, puis, nous décrivons et évaluons un nouveau modèle analytique qui prend en compte à la fois les caractéristiques de la couche physique et de la couche MAC du standard IEEE 802.11.

La troisième partie de cette thèse étudie les problèmes liés aux mécanismes de sélection du mode de transmission physique. Ce type de mécanisme nécessite une collaboration étroite entre les couches MAC et PHY. Dans le chapitre 5, nous présentons la liste des paramètres clés qui doivent être pris en compte lors de la conception d'un mécanisme de sélection de mode de transmission physique. Enfin, nous décrivons les trois algorithmes de sélection de mode de transmission physique qui ont été développés dans le cadre de cette thèse: AARF, CLARA et AMRR. Ce chapitre présente les résultats de simulation effectuées avec ns-2 et Matlab ainsi que les résultats expérimentaux qui montrent le gain en performances de ces mécanismes.

Enfin, la dernière et quatrième partie de cette thèse présente un mécanisme d'adaptation

qui exploite de manière efficace les informations provenant des couches applicatives, MAC et PHY pour choisir dynamiquement le mode de transmission physique. Nous présentons une évaluation des performances de ce mécanisme en analysant la qualité vidéo obtenue à l'aide d'un encodeur vidéo à granularité fine.

F.2 Chapitre 2: Réseaux Locaux Sans Fil IEEE 802.11

IEEE 802.11 est le standard de réseaux locaux sans-fil le plus déployé de nos jours. Ce standard se concentre sur les deux couches inférieures du modèle ISO, à savoir la couche physique et la couche des liaisons données. Il utilise le protocole d'accès au médium CSMA/CA (carrier sense multiple access with collision avoidance) pour résoudre la contention entre les différents émetteurs.

En ce qui concerne la couche physique, ce standard propose trois interfaces de communication dans sa première version qui a été élaborée en 1997: l'infrarouge (IR), l'étalement de spectre avec sauts de fréquence (FHSS) et l'étalement de spectre par séquence directe (DSSS). Deux ans plus tard, l'IEEE a proposé les standards 802.11b et 802.11a qui permettent de dépasser la barre des 10Mbps sur les bandes de fréquence 2.4GHz et 5.2GHz. Le standard 802.11b conserve l'interface DSSS et intègre un schéma de codage plus efficace appelé CCK (complementary code keying). Le standard 802.11a utilise une technique de transmission qui est incompatible avec 802.11b, c'est une modulation par multi-porteuse appelée OFDM (Orthogonal Frequency Division Multiplexing). Début 2000, l'IEEE a formé un groupe de travail dont le but était de définir le prochain standard haut débit sur la bande 2.4GHz, il s'agit du standard 802.11g. Ce standard utilise pour la couche PHY toutes les modulations disponibles proposées dans les normes 802.11a/b. Pour cette raison, nous omettons l'étude du 802.11g dans le cadre de cette dissertation. Pour élaborer et déployer les mécanismes adaptatifs et inter couche dans les réseaux IEEE 802.11, une connaissance approfondie des fonctionnalités des couches PHY et MAC est nécessaire. Ainsi, dans ce chapitre nous présentons les paramètres importants des couches MAC et PHY dans les standards 802.11a/b/e. Dans la première partie de ce chapitre, nous présentons

le modèle référence IEEE 802.11 ainsi que le protocole CSMA/CA avec le mode de base et le mode optionnel RTS/CTS dans les sections 2.3 et 2.4. Une introduction au standard 802.11e qui propose des mécanismes de support de qualité de services dans les réseaux sans fil est présentée dans la section 2.4.1. Nous présentons les différentes interfaces de communication, à savoir l'infrarouge (IR), l'étalement de spectre avec sauts de fréquence (FHSS), l'étalement de spectre par séquence directe (DSSS), et OFDM dans la section 2.5.

Après présentation d'un modèle de transmission pour la couche physique IEEE 802.11 dans la section 2.6, nous présentons les fonctionnalités des modules de couche de PHY pour les standards 802.11a/b dans la section 2.7 et la section 2.8. Nous présentons également le format des paquets IEEE 802.11 dans la section 2.9. La section 2.10 conclut ce chapitre.

F.3 Chapitre 3: Analyse de Performance des Modes de Transmission pour IEEE 802.11

A la difference des réseaux filaires qui peuvent offrir une large bande passante, celle des réseaux sans fils est limitée à cause de la caractéristique de son canal physique (i.e., l'air) qui est à l'origine de l'erreur des paquets transmises. Il est donc important d'évaluer les performances des équipements sans fils en considérant les caractéristiques de la transmission, les paramètres du canal sans fil et la structure des équipements. Dans ce chapitre, nous envisageons tous ces paramètres cités et on présentons une évaluation complète de tous les modes de transmission pour les équipements 802.11a/b.

En général, les performances de la transmission des données dans les réseaux sans fil sont bien capturées par l'observation du taux d'erreur qui est en fonction du rapport signal bruit au récepteur. Des modèles différents sont déjà proposés dans la littérature pour calculer le rapport signal bruit dans les réseaux sans fils. Ces modèles dépendent de la distance entre la source et le récepteur, du taux de perte des paquets, et du gain du canal.

Celui-ci qui est une variable dynamique, est modélisé par des fonctions de probabilité. Il y a plusieurs types de probabilité de distribution diverse. Dans ce chapitre, on décrit les trois dis-

tributions les plus utilisables pour les fonctions de probabilité, i.e., AWGN, Rayleigh, et Rician. Nous envisageons aussi le cas des récepteurs a multi-liaison. Pour ce genre des récepteurs, le signal est détecté et décodé en utilisant les différents signaux reçus. Comme nous l'avons présenté dans le Chapitre 2, chaque mode physique de transmission nécessite de spécifier la modulation convenable, le codage binaire convolutionnel, et le débit de transmission de données.

De plus, selon la spécification du standard IEEE 802.11, chaque équipement doit utiliser des techniques de transmission (modulation convenable pour canal sans fil) parmi OFDM, DSSS, FH, et IR. Dans notre évaluation, on considère tous les paramètres déjà cites. Pour le codage binaire convolutionnel, on suppose que le récepteur utilise un décodeur Viterbi qui est déjà recommande dans la spécification du standard.

Ce chapitre est organisé comme suit. Tout d'abord, nous présentons la structure des récepteurs IEEE 802.11a/b dans la section 3.3. Une introduction sur la modélisation du canal sans fil et ses paramètres importants est présentée dans la section 3.4. Dans la même section, nous discutons AWGN et les canaux de Fading.

Ensuite, nous présentons une évaluation de performance détaillée des modes de transmission 802.11b respectivement dans la section 3.5 et 3.6. Cette analyse est faite sur un canal AWGN et des canaux de Rayleigh Fading en considérant la structure du récepteur multipath. On évalue aussi la performance des décodeurs Vertabi dans 802.11a. Finalement, nous concluons le chapitre dans la section 3.7.

F.4 Chapitre 4: Modélisation des Couches MAC et PHY du Standard IEEE 802.11

Différents modèles analytiques ont été proposés et un grand nombre de simulations ont été menées pour évaluer la couche MAC du standard IEEE 802.11 pour les réseaux locaux sans fil. Ces études calculent le débit de saturation en considérant la couche MAC uniquement et donc le protocole CSMA/CA.

Dans la majorité des cas, la plus grande imperfection de ces modèles est que les caractéristiques de la couche physique ne sont pas prises en compte. Les modèles existants pour le standard IEEE 802.11 supposent que toutes les stations utilisent les mêmes paramètres de la couche physique et ont des caractéristiques du canal de transmission identiques. Par exemple, pour envoyer les paquets au point d'accès, les stations utilisent la même puissance, le même codage, la même modulation, etc. Ainsi quand plus d'une station transmettent des paquets au même instant, tous les paquets émis sont perdus. Le modèle de Bianchi est le modèle le plus utilisé et entre dans cette catégorie. Il établit une expression pour calculer le débit de saturation en fonction de la probabilité de transmission et de la probabilité de collision de chaque paquet.

Mais ce modèle est peu réaliste car en pratique, une station peut être très proche du point d'accès alors que les autres stations sont loin du point d'accès. Ce comportement, connu sous le nom d'"Effet de Capture " (Capture Effect), est observé sur les points d'accès en présence de signaux multiples: seul le signal radio le plus fort est reçu. Le phénomène d'effet de capture peut être analysé en considérant les positions des stations par rapport au point d'accès.

Nous présentons dans ce chapitre un modèle analytique pour estimer les performances de la couche MAC du standard 802.11 qui prend en compte les positions des stations (et donc les caractéristiques de la couche physique). Notre modèle se base sur le modèle de Bianchi qui estime les performances de 802.11 en utilisant une chaîne de Markov à temps discret, mais qui suppose implicitement que toutes les stations se trouvent à la même distance par rapport au point d'accès.

Dans notre modèle, étant donnée la position d'une station, nous calculons son débit de saturation en tenant compte de la position des autres stations (c'est-à-dire, l'interférence engendrée par les autres stations). Notre modèle permet aussi d'estimer le débit total de saturation du réseau sans fil. En effet, nous pouvons calculer pour une topologie donnée le débit de toutes les stations sans fil en utilisant le protocole CSMA/CA du standard IEEE 802.11 avec un protocole de couche PHY (par exemple, 802.11a/b/g). Dans ce chapitre, nous considérons seulement le scénario d'une communication " montante " (uplink), c'est-à-dire lorsque la station mobile émet dans la direction du point d'accès. Le scénario avec un trafic bidirectionnel est une extension

facile de ce modèle.

Nous résolvons notre modèle numériquement et nous montrons que le débit de saturation d'une station est dépendant de sa propre position et de la position des autres stations par rapport au point d'accès. Nous validons nos résultats par simulation avec une version du simulateur ns-2 qui émule non seulement les caractéristiques de la couche MAC mais aussi celles de la couche PHY et MAC des réseaux sans fil IEEE 802.11. Cette évaluation est effectuée pour de standard IEEE 802.11b avec des débits de 1 et 2 Mbps. Ce modèle peut facilement être utilisé pour les autres modes de transmission en considérant leur probabilité d'erreur de transmission de paquet comme expliqué dans les sections 3.5 et 3.6.

Nos résultats montrent qu'une station obtient un débit élevé lorsqu'elle est proche du point d'accès et que son débit diminue quand elle s'éloigne de lui. Il existe une distance à partir de laquelle le débit d'une station baisse rapidement. Nous pensons que notre modèle est un bon outil pour dimensionner les réseaux 802.11 et étudier leur performance.

Dans ce chapitre, nous présentons tout d'abord les travaux relatifs à la modélisation de la couche MAC et PHY des réseaux IEEE 802.11 dans la section 4.3. Dans la section 4.4, nous présentons notre modèle. Les résultats numériques et des simulations sont présentés dans la section 4.6. La section 4.7 conclut ce chapitre avec des pointeurs pour de futurs travaux.

F.5 Chapitre 5: Sélection des Modes de Transmission dans le Standard IEEE 802.11

Les trois couches physiques IEEE 802.11a/b/g disponibles aujourd'hui offrent toutes des services multi-modes. Afin d'obtenir des performances élevées avec des conditions de transmission variables, les systèmes qui utilisent ces couches physiques doivent continuellement changer le mode de transmission utilise pour maximiser le débit disponible au niveau applicatif.

Bien que les algorithmes de sélection de mode constituent un composant important de la performance de ces réseaux locaux sans fil, aucun algorithme n'a été proposé dans le standard IEEE 802.11a/b/g. Les problèmes fondamentaux liés à l'implémentation de tels algorithmes

n'ont jamais été pris en considération dans des articles publiés.

Nous présentons donc dans ce chapitre le paramètre de ces systèmes 802.11 qui a le plus grand impact sur la conception d'algorithmes de contrôle de mode de transmission. Nous avons identifié deux types de systèmes: les systèmes à faible latence et ceux à forte latence de communication entre les couches physiques et à couche MAC. Puis, nous proposons deux nouveaux algorithmes de contrôle de mode de transmission dont les performances approchent l'optimum représenté par RBAR et qui sont, contrairement à RBAR, réalisables aujourd'hui sans modifications incompatibles du standard 802.11. AARF (*adaptive ARF*) et AMRR (*adaptive multi rate retry*), conçus respectivement pour les systèmes dits à faible latence et à forte latence, ont été implémentés et simulés à l'aide de NS.

Nous avons aussi évalué une implémentation de AMRR dans un driver Linux pour cartes 802.11 qui utilisent le chipset AR5212. Les résultats d'expérimentaux confirment les résultats de simulation et montrent une amélioration importante du débit obtenu au niveau applicatif. Finalement, nous avons présenté CLARA (closed-loop adaptive rate allocation) qui prend en compte les aspects les plus intéressants de ARF et RBAR et ajoute des fonctionnalités supplémentaires afin d'améliorer la réponse aux problèmes d'atténuation muti-path, la détection de canal et l'échange de signaux de contrôle.

Ce chapitre est organisé comme suit. Tout d'abord, nous présentons les paramètres principaux qui doivent être pris en compte pour des mécanismes d'adaptation de débits dans la Section 5.3. Dans la section 5.4, nous présentons un aperçu des mécanismes existants. Nous présentons et évaluons nos algorithmes AAFR, CLARA, et AMRR dans les Section 5.5, 5.6, et 5.7. Finalement, nous concluons le chapitre dans la section 5.8.

F.6 Chapitre 6: Mécanisme de Sélection de Mode de Transmission en Fonction des Besoins de l'Application

Les nouvelles applications multimédias introduisent des problèmes de transmission intéressants dans les réseaux ad hoc (MANETs). En général, Il y a beaucoup de facteurs qui peuvent changer

la topologie de MANETs tels que la mobilité des stations ou les changements de niveau de puissance. Par exemple, le mécanisme de contrôle de puissance dans la couche physique peut affecter toutes les autres stations dans MANETs en modifiant les niveaux d'interférences des autres stations et la connectivité du réseau qui effectue aussi le routage. Par conséquent, le mécanisme de contrôle de puissance n'est pas limité à la couche physique et peut affecter l'opération des couches plus hautes. Il peut aussi être une occasion de proposer des mécanismes intercouches dans les réseaux sans fil. En effet une approche intercouche devient nécessaire pour optimiser de la transmission des données selon les caractéristiques des données et aux conditions de canal.

Dans ce chapitre, nous présentons un mécanisme de sélection automatique de mode de transmission appelé MORSA pour des réseaux locaux sans fil 802.11 qui s'adapte à la fois aux conditions du canal et aux caractéristiques du média transmis. L'objectif de cette étude est d'améliorer la transmission de certaines applications multimédia qui sont robustes aux erreurs de bits. En particulier, si la couche MAC est consciente que l'application tolère un certain pourcentage d'erreurs de transmission binaire, elle peut sélectionner un mode qui supporte un débit de transmission plus élevé. Pour cela, l'application émettrice peut spécifier une demande de qualité de service (débit de transmission, tolérance d'erreurs de transmission binaire, etc) afin que le récepteur sélectionne le meilleur mode de transmission tout en prenant en compte les conditions dynamiques du canal. Nous présentons les modifications nécessaires dans les en-têtes de paquets de contrôle et de données pour implanter ce mécanisme dans un réseau sans fil 802.11.

Nous avons comparé notre mécanisme avec d'autres algorithmes de sélection de débit par le biais du simulateur réseau NS-2. Les résultats montrent que le nouveau mécanisme permet d'augmenter jusqu'à 4 Mb/s le débit reçu et d'accroître de 20 mètres la portée de la transmission. Nous évaluons aussi, la qualité d'une vidéo transmis sur un canal 802.11a en utilisant MORSA et la comparons à la qualité obtenue quand nous ne tenons pas compte des caractéristiques de l'application (en utilisant l'approche standard). Nos résultats prouvent que MORSA peut atteindre une qualité visuelle comparable que celle obtenue avec le mécanisme

standard tout en utilisant seulement très bas (5%) FEC au niveau d'application au lieu de la couche physique FEC (50% ou 25%).

Dans ce chapitre, nous présentons tout d'abord les travaux relatifs a l'interaction intercouche pour les réseaux sans fil IEEE 802.11 dans la section 6.3. Dans la section 6.4, nous présentons notre mécanisme MORSA. Les résultats de simulation NS sont présentés et évalués dans la section 6.5. Nous évaluons la qualité d'une transmission vidéo sur un canal sans fil 802.11a dans la section 6.6. La section 6.7 conclut ce chapitre.

F.7 Chapitre 7: Conclusion

F.7.1 Résumé des Contributions

Nous avons examiné dans cet ouvrage la conception de plusieurs mécanismes d'interaction intercouche et de communication adaptative pour les réseaux sans fil IEEE 802.11. Les idées présentées dans cet ouvrage peuvent améliorer de façon significative la performance des réseaux sans fil IEEE 802.11. Dans cette section nous résumons les contributions de ce travail de recherche.

Dans le chapître 2, nous avons présenté une synthèse de la couche MAC 802.11 CSMA/CA, de la couche MAC 802.11e, ainsi que plusieurs techniques de transmission définies dans 802.11 telles que IR, DSSS, et OFDM.

Dans le chapître 3, nous avons présenté une évaluation détaillée des modes de transmission pour 802.11a/b, avec l'objectif de décrire les effets que peuvent avoir, sur les performances des appareils 802.11, les caractéristiques du canal radio (AWGN, fading, et multipath), la modulation, la structure de réception, le codage convolutif binaire, le décodage de Viterbi, et le débit physique.

Dans le chapître 4 nous présentons DAW, un modèle qui tient compte de la distance, et qui permet de modéliser la performance des cartes 802.11 en utilisant à la fois les caractéristiques des couches MAC et PHY. Ce modèle repose sur celui de Bianchi pour la couche MAC CSMA/CA de 802.11. DAW utilise le niveau d'interférence et de bruit pour calculer la probabilité d'une

perte de paquet, et reflète ainsi l'effet de capture. Nous analysons la performance de réseaux sans fil dans des scénarios incluant des topologies fixes ou aléatoires, grâce au modèle DAW. Nous validons notre modèle par le biais de simulateur NS-2, et montrons que DAW permet d'obtenir des résultats plus réalistes que les modèles ne reposant que sur la couche 802.11 MAC.

Le chapître 5 concerne les mécanismes de sélection du débit physique. Nous y décrivons le vaste état de l'art sur cette question, en présentant pour chaque algorithme l'idée principale, ainsi que ses avantages et inconvénients. Nous identifions également les paramètres principaux utilisés par ces mécanismes. Les contributions principales de ces algorithmes sont:

1. AARF et AMRR, conçus pour des systèmes de communication à latence faible et élevée respectivement, présentent tout deux des performances proches de l'optimal, si l'on compare les résultats expérimentaux à ceux de la simulation.

2. AMRR est implanté en utilisant un driver Linux pour le chipset Atheros AR5212. Cette implémentation démontre qu'AMRR peut être aisément mis en œuvre avec la technologie actuelle, et qu'il améliore les performances.

3. CLARA est un mécanisme de sélection intelligent qui peut choisir le meilleur mode de transmission à tout moment, en se basant sur les attributs mesurés et estimés du canal radio.

4. CLARA pourrait être implémenté au niveau physique, il est indépendant de la couche MAC et pourrait donc être utilisé également pour des stations mobiles 802.11e.

Dans le chapître 6, nous présentons MORSA, un mécanisme de sélection de débit qui prend en compte les caractéristiques de l'application. Nous y abordons les problèmes relatifs à la mise en œuvre l'implémentation d'une station 802.11e. Nous démontrons que la performance des réseaux mobiles ad hoc peut être améliorée de manière significative en utilisant ce simple mécanisme inter-couche. Nous évaluons également la qualité d'une application vidéo en utilisant le décodeur WAVIX lorsqu'il emploie le mécanisme MORSA, et démontrons que l'application peut ainsi obtenir de meilleurs résultats en termes de débit, de latence et de gigue.

F.7.2 Directions Futures

À l'heure actuelle, les mécanismes d'interaction inter-couche dans les réseaux sans fil font l'objet d'une activité importante de la part des équipes de recherche en réseaux. Des efforts de standardisation sont en cours auprès de BRAN HiperLAN2, 3GPP (High Speed Downlink Packet Access), et IEEE, tels que décrits en section 6.3.

La plupart des travaux de recherche présentés dans cette thèse se concentrent sur ces mécanismes et sur des sujets apparentés. Nous décrivons à présent les directions possibles futures autour des domaines de recherche abordés dans cette thèse.

Dans un premier temps, la modélisation inter-couche pourrait proposer un modèle plus précis; cela pourrait consister à intégrer l'évanouissement, ainsi que les canaux radio, dans le modèle DAW, ou encore l'évaluation du modèle avec d'autres distributions aléatoires de topologie. Nous pourrions également évaluer le délai des paquets et modéliser plus finement les liens montant et descendant. Ce modèle serait utile pour différents problèmes qui se posent également dans les réseaux ad hoc. Par exemple notre modèle permet d'évaluer la capacité, la connectivité, le contrôle d'énergie, et les coûts des réseaux ad hoc, puisqu'il tient compte à la fois les couches MAC de 802.11 et PHY. Des modèles basés sur DAW pourraient être utilisés, avec quelques améliorations, pour la mise en place des réseaux WLAN, par exemple pour déterminer l'emplacement optimal des points d'accès, afin d'obtenir la meilleure couverture et le meilleur débit. Nous pouvons également envisager d'évaluer la performance de diverses applications (e.g. TCP) sur les réseaux 802.11 avec le modèle DAW.

En ce qui concerne les mécanismes adaptatifs de sélection de débit physique, de nouvelles études sont nécessaires pour évaluer les mécanismes proposés sur des connections "multi-hop". De manière générale, les mécanismes proposés jusqu'ici sont évalués pour une seule connexion. Cependant dans les topologies réelles on a la plupart du temps affaire à plusieurs connexions et plusieurs "hops". À notre connaissance, aucun travail de recherche à ce jour ne porte sur ces problèmes.

D'autres problèmes liés aux mécanismes de sélection concernent l'allocation des fréquences, et la gestion de l'énergie. Comme nous l'avons évoqué dans le chapître 5, la définition d'un

mécanisme permettant de gérer de manière conjointe l'utilisation des fréquences, l'utilisation de l'énergie et le choix du débit va devenir un sujet très populaire, en particulier dans les réseaux ad hoc.

L'évaluation de mécanismes adaptatifs sur des structures de récepteurs MIMO ou des réseaux de capteurs pourrait également présenter de l'intérêt. Enfin, les problèmes liés à l'implantation de CLARA devraient être approfondis.

Nous pouvons aussi proposer des mécanismes de sélection hybrides en utilisant AARF et CLARA pour différentes caractéristiques du canal comme cela est fait dans HRC [47, 48]. Il semble également possible d'obtenir de meilleures performances en utilisant des algorithmes simples et efficaces comme FAR [45] en conjonction avec AARF ou CLARA pour déterminer le débit utilisé pour la transmission de trames de contrôle (i.e. RTS, CTS, et ACK).

Concernant les communications adaptatives et le mécanisme MORSA, il faudrait envisager d'évaluer ces mécanismes avec d'autres types d'application comme la voix sur IP ou avec d'autres codages vidéo. Par exemple, la qualité de flux audio abîmés pourrait être étudiée avec le EModel.

Puisque MORSA encourage les utilisateurs à émettre avec des débits plus importants, cela devrait réduire les effets que l'on constate dans les scénarios où un réseau sans fil utilise plusieurs débits différents, et dans lesquels les utilisateurs ne sont pas traités à la même enseigne. Ce problème peut être étudié avec MORSA. Une implémentation réelle de MORSA sur une plateforme de radio logicielle (SDR) pourrait également être envisagée, par exemple en utilisant des cartes 802.11a/SDR disponibles auprès de Cisco.

BIBLIOGRAPHY

[1] IEEE 802.11 WG, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," 1999. 1, 8, 9, 10, 11, 15, 16, 23, 70, 79, 92, 129

[2] IEEE 802.11 WG part 11a, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, High-speed Physical Layer in the 5 GHz Band," 1999. 2, 8, 12, 14, 17, 19, 20, 22, 126, 130, 145

[3] IEEE 802.11 WG part 11b, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Higher Speed PHY Layer Extension in the 2.4 GHz Band," 1999. 2, 8, 12, 14, 23

[4] IEEE 802.11 WG part 11g, "Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications, Further Higher Speed Physical Layer Extension in the 2.4 GHz Band," 2003. 2, 8, 12, 14, 17

[5] M.H. Manshaei, G. R. Cantieni, C. Barakat, and T. Turletti, "Performance Analysis of the IEEE 802.11 MAC and Physical Layer Protocol," in *IEEE WoWMoM*, June 2005. 3, 163

[6] M. Lacage, M.H. Manshaei, and T. Turletti, "IEEE 802.11 Rate Adaptation: A Practical Approach," in *ACM MSWiM*, October 2004. 3, 88, 163

[7] C. Hoffmann, M.H. Manshaei, and T. Turletti, "CLARA: Closed-Loop Adaptive Rate Allocation for IEEE 802.11 WirelessLANs," in *IEEE WIRELESSCOM*, June 2005. 3, 84, 88, 89, 163

[8] "Madwifi," http://sourceforge.net/projects/madwifi/. 3, 76, 86, 108, 163

[9] Tetsuro Ueda, Shinsuke Tanaka, Siuli Roy, Dola Saha, and Somprakash Bandyopadhyay, "ACR: An Adaptive Communication-Aware Routing through Maximally Zone-Disjoint Shortest Paths in Ad Hoc Wireless Networks with Directional Antenna," *Journal of Wireless Communications and Mobile Computing*, to appear 2004. 4

[10] Mohammed-Slim Alouini, *Adaptive and Diversity Techniques for Wireless Digital Communications over Fading Channels*, Ph.D. thesis, Department of Electrical Engineering, California Institute of Technology, June 1998. 4

[11] M. H. Manshaei, T. Turletti, and M. Krunz, "A media-oriented transmission mode selection in 802.11 wireless LANs," in *IEEE Wireless Communications and Networking Conference (WCNC)*, March 2004. 4, 115, 118, 164

[12] M.H. Manshaei, T. Turletti, and T. Guionnet, "An Evaluation of Media-Oriented Transmission Mode Selection for Multimedia Application in MANETs," *EURASIP Journal on Wireless Communications and Networking, Ad Hoc Networks: Cross-Layer Issues*, First quarter 2005. 4, 115, 164

[13] IEEE 802.11e WG, "Amendment : Medium Access Control (MAC) Quality of Service (QoS) Enhancements," January 2005. 8, 13, 25, 113, 115

[14] D. B. Johnson, D. A. Maltz, and J. Broch, "Ad Hoc Networking, Chapter 5: DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks," 2001, Addison-Wesley. 10, 121, 122

[15] Charles E. Perkins, Elizabeth M. Royer, and Samir R. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," *draft-ietf-manet-aodv-08.txt*, March 2001. 10

[16] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," *Request for Comments-3626*, October 2003. 10

[17] Q. Ni, L. Romdhani, and T. Turletti, "A Survey of QoS Enhancements for IEEE 802.11 Wireless LAN," *Wiley Journal of Wireless Communication and Mobile Computing (JWCMC)*, vol. 4, no. 5, pp. 547–566, 2004. 14, 115

[18] Imad Aad, *Quality of service in wireless local area networks*, Ph.D. thesis, Universite Joseph Fourier de Grenoble, 2002. 14

[19] Stefan Mangold, Sunghyun Choi, Peter May, Ole Klein, Guido Hiertz, and Lothar Stibor, "IEEE 802.11e wireless LAN for Quality of Service," in *Proceedings of European Wireless*, February 2002. 14

[20] Homayoun Hashemi, "The Indoor Radio Propagation Channel," *Proceeding of The IEEE*, vol. 81, no. 7, pp. 943–968, July 1993. 30, 32

[21] John G. Proakis, "Digital Communications," 1995, 4th edition, McGraw Hill. 33, 34, 35, 38, 39, 41, 44

[22] M. K. Simon and M. S. Alouini, "Digital Communication over Fading Channels," 2004, 2nd edition, John Wiley. 33, 34, 36, 38, 39, 44

[23] L.L. Yang and L. Hanzo, "A Recursive Algorithm for the Error Probability Evaluation of M-QAM," *IEEE Communications Letter*, vol. 4, no. 10, October 2000. 35

[24] M. B. Pursley and D. J. Taipale, "Error Probabilities for Spread-Spectrum Packet Radio with Convolutional Codes and Viterbi Decoding," *IEEE Transactions on Communications*, vol. 35, no. 1, pp. 1–12, January 1987. 39

[25] P. Frenger, P. Orten, T. Ottosson, and A. Svensson, "Multi-Rate Convolutional Codes," Tech. Rep., Communication System Group, Chalmers University of Technology, Sweden, April 1998. 39

[26] M. K. Simon, A. M. Hinedi, and W. C. Lindsey, "Digital Communication Techniques, Signal Design and Detection," 1995, Printice Hall. 41

[27] C. Jonietz, W. Gerstacker, and R. Schober, "Receiver concepts for wlan ieee 802.11b," in *International Conference on Source and Channel Coding*, January 2004. 45

[28] J. d. Prado and S. Choi, "Link Adaptation Strategy for IEEE 802.11 WLAN via Received Signal Strength Measurement," in *ICC*, May 2003. 45, 70, 76, 83, 92

[29] R. Schober, W.H. Gerstacker, and L. Lampe, "Performance Analysis and Design of STBC's for Frequency-Selective Fading Channels," *IEEE Transactions on Wireless Communications*, 2003. 45

[30] Giuseppe Bianchi, "Performance Anaylsis of the IEEE 802.11 Distributed Coordination Function," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 18, no. 3, March 2000. 50, 51, 53, 92

[31] A. U. Shankar A. Agrawala A. Kochut, A. Vasan, "Sniffing out the correct Physical Layer Capture model in 802.11b," in *IEEE ICNP*, October 2004. 50, 139

[32] P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, March 2000. 50, 67

[33] H. Wu, Y. Peng, K. Long, and J. Ma, "Performance of Reliable Transport Protocol over IEEE 802.11 Wireless LAN: Analysis and Enhancement," in *IEEE INFOCOM*, 2002, pp. 599–607. 50, 51, 59, 92

[34] C. H. Foh and M. Zukerman, "Performance Analysis of the IEEE 802.11 MAC Protocol," in *EW*, February 2002, pp. 184–190. 51

[35] H. Kim and J. Hou, "Improving protocol capacity with model-based frame scheduling in IEEE 802.11-operated WLANs," in *MobiCom*, September 2003. 51

[36] P. Chatzimisios, V. Vitsas, and A. C. Boucouvalas, "Throughput and Delay analysis of IEEE 802.11 protocol," in *IEEE International Workshop on Networked Appliances (IWNA)*, 2002. 51

[37] P. Chatzimisios, V. Vitsas, and A. C. Boucouvalas, "IEEE 802.11 Packet Delay A Finite Retry Limit Analysis," in *IEEE Global Telecommunications Conference (Globecom)*, 2003. 51

[38] Z. Helkov and B. Spasenovski, "Saturation Throughput-Delay Analysis of IEEE 802.11 DCF in Fading Channel," in *ICC*, May 2003. 53, 54

[39] P. Chatzimisios, V. Vitsas, and A. C. Boucouvalas, "Performance Analysis of IEEE 802.11 DCF in Presence of Transmission Errors," in *ICCC*, June 2004. 53

[40] J. Gosteau, M. Kamoun, S. Simoens, and P. Pellati, "Analytical developments on QoS enhancements provided by IEEE 802.11 EDCA," in *ICC*, June 2004. 54

[41] T. S. Rappaport, "Wireless Communications: Principles and Practice," 1996, Printice Hall. 55

[42] M. H. Manshaei and T. Turletti, "Simulation-Based Performance Analysis of 802.11a WLANs," in *International Symposium on Telecomunication*, August 2003. 60, 97, 117, 118

[43] A. Kamerman and L. Monteban, "WaveLAN II: A high-performance wireless LAN for the unlicensed band," Tech. Rep., Bell Labs Technical Journal, Summer 1997. 70, 76, 77, 98, 108

[44] G. Holland, N. Vaidya, and P. Bahl, "A Rate-Adaptive MAC Protocol for Multi-Hop Wireless Networks," in *ACM MOBICOM*, July 2001. 70, 71, 77, 97, 98, 108, 117, 121

[45] Z. F. Li, A. Das, A. K. Gupta, and S. Nandi, "A Full Auto Rate (FAR) MAC Protocol for Wireless Ad Hoc Networks," *IEE Proceedings Communications*, to appear. 70, 79, 141, 176

[46] Z. Ji, Y. Yang, J. Zhou, M. Takai, and R. Bagrodia, "Exploiting Medium Access Diversity in Rate Adaptive Wireless LANs," in *ACM MOBICOM*, September 2004. 70, 75, 77, 81, 82

[47] I. Haratcherev, K. Langendoen, R. Lagendijk, and H. Sips, "Hybrid Rate Control for IEEE 802.11," in *MobiWAC*, September 2004. 70, 77, 84, 141, 176

[48] I. Haratcherev, J. Taal, K. Langendoen, R. Lagendijk, and H. Sips, "Automatic IEEE Rate Control for Streaming Application," *Journal of Wireless Communication and Mobile Computing (JWCMC)*, vol. 5, no. 4, pp. 421–437, June 2005. 70, 84, 92, 97, 141, 176

[49] D. Qiao, S. Choi, A. Jain, and K.G. Shin, "MiSer: an optimal low-energy transmission strategy for IEEE 802.11a/h," in *ACM MOBICOM*, 2003, pp. 161–175. 70, 76, 85

[50] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly, "Opportunistic Media Access for Multirate Ad Hoc Networks," in *ACM MOBICOM*, September 2002. 70, 75, 77, 80

[51] D. Aguayo, J. C. Bicket, S. Biswas, G. Judd, and R. Morris, "Link-level measurements from an 802.11b mesh network," in *SIGCOMM*, 2004, pp. 121–132. 70, 87

[52] Agere systems, "WaveLAN 802.11b chipset for Standard Form Factors," December 2002, Preliminary Product Brief. 75

[53] M. Lacage, M.H. Manshaei, and T. Turletti, "IEEE 802.11 Rate Adaptation: A Practical Approach," Tech. Rep., INRIA, May 2004. 75

[54] M. Heusse, F. Rousseau, G. Berger-Sabbatel, and A. Duda, "Performance anomaly of 802.11b," in *IEEE INFOCOM*, April 2003. 75, 141

[55] I. Tinnirello and S. Choi, "Temporal Fairness Provisioning in Multi-Rate Contention-Based 802.11e WLANs," in *IEEE WoWMoM*, June 2005, pp. 220–230. 75

[56] V. Kanodia, A. Sabharwal, and E. Knightly, "MOAR: A Multi-channel Opportunistic Auto-rate Media Access Protocol for Ad Hoc Networks," in *IEEE Broadnets*, October 2004. 76

[57] Daji Qiao, Sunghyun Choi, and Kang G. Shin, "Goodput Analysis and Link Adaptation for IEEE 802.11a Wireless LANs," *IEEE Transaction on Mobile Computing*, vol. 1, no. 4, pp. 278–292, October-December 2002. 77, 88, 97, 98

[58] John Bicket, Daniel Aguayo, Sanjit Biswas, and Robert Morris, "Architecture and evaluation of an unplanned 802.11b mesh network," in *Proceedings of the 11th ACM International Conference on Mobile Computing and Networking (MobiCom '05)*, Cologne, Germany, August 2005. 87

[59] John Bicket, "Bit-rate selection in wireless networks," M.S. thesis, Massachusetts Institute of Technology, February 2005. 87

[60] D. Qiao and S. Choi, "Goodput Enhancement of IEEE 802.11a Wireless LAN via Link Adaptation," in *ICC*, June 2001. 88, 97, 130

[61] J. C. Wu, H. Liu, and Y. Lung, "An Adaptive Multirate IEEE 802.11 Wireless LAN," in *International Conference on Information Networking*, 2001, pp. 411–418. 88

[62] Robert M.Metcalfe and David R. Boggs, "Ethernet: Distributed Packet Switching for Local Computer Networks," *ACM Communications*, vol. 19, no. 5, pp. 395–404, July 1976. 89

[63] "Network simulator 2 (ns2)," http://www.isi.edu/nsnam/ns/. 96

[64] "ar5212," http://www.atheros.com/pt/index.html. 110

[65] S. Shakkottai, T. S. Rappaport, and P. C. Karlsson, "Cross-layer design for wireless networks," *IEEE Communications Magazine*, vol. 41, no. 10, pp. 74–80, October 2003. 112, 113, 114

[66] S. Toumpis, *Capacity and cross-layer design of wireless ad hoc networks*, Ph.D. thesis, Department of Electrical Engineering of Stanford University, USA, July 2003. 112, 113

[67] A. Safwat, H. Hassanein, and H. Mouftah, "Cross-Layer designs for energy-efficient wireless ad hoc and sensor networks," in *22nd IEEE International Performance, Computing, and Communications Conference*, April 2003. 112

[68] W. H. Yuen, H. Lee, and T. D. Andersen, "A cross layer networking system for mobile ad hoc networks," in *IEEE PIMRC*, September 2002. 112, 114

[69] G. Holland and N. Vaidya, "Analysis of TCP Performance Over Mobile Ad Hoc networks," *ACM Wireless Networks*, vol. 8, no. 2, March 2002. 112, 113

[70] J. Mitola, "The Software Radio Architecture," *IEEE Communications Magazine*, May 1995. 112

[71] H. Jegou and C. Guillemot, "Source Multiplexed Codes for Error-prone Channels," in *IEEE ICC*, May 2003. 112

[72] T. Guionnet, *Codage robuste par descriptions multiples pour transmission sans fil d'information multimédia*, Ph.D. thesis, University of Rennes, 2003. 112

[73] V. Bhuvaneshwar, M. Krunz, and Alaa Muqattash, "A Cross-Layer Power Aware Protocol for Mobile Ad Hoc Networks," in *IEEE ICC*, June 2004. 113

[74] U.Kozat, I. Koutsopoulos, and L. Tassiulas, "A Framework for Cross-layer Design of Energy-efficient Communication with Qos Provisioning in Multi-hop Wireless Networks," in *IEEE INFOCOM*, March 2004. 113

[75] S. Krishnamachari, M. VanderSchaar, S. Choi, and X. Xu, "Video Streaming over Wireless LANs: A Cross-Layer Approach," in *IEEE Packet Video*, April 2003. 113, 114

[76] M. Conti, G. Maselli, G. Turi, and S. Giordano, "Cross-Layering in Mobile Ad Hoc Network Design," *Computer Magazine*, vol. 37, no. 2, February 2004. 113

[77] Y. Shan and A. Zakhor, "Cross Layer Techniques for Adaptive Video Streaming Over Wireless Networks," in *International Conference on Multimedia and Expo*, August 2002. 114

[78] H. Liu and M. El Zarki, "Adaptive source rate control for real-time wireless video transmission," *ACM Mobile Networks and Applications*, vol. 3, no. 1, 1998. 114

[79] L.A. Larzon, M. Degermark, and S. Pink, "UDP Lite for Real Time Applications," Tech. Rep., HP Laboratories Bristol, April 1999. 116

[80] "The Rice University Monarch Project, Mobile Networking Architectures," http://www.monarch.cs.rice.edu/. 117, 121

[81] "Cisco Aironet 1200 Series Access Point Hardwara Installation Guide," http://www.cisco.com. 117

[82] R. Khalili and K. Salamatian, "A new analytic approach to evaluation of Packet Error Rate in Wireless Networks," Tech. Rep., LIP6-CNRS, October 2004. 126

[83] G. David Forney Jr., "Convolutional codes II. maximum-likelihood decoding," *Information and Control*, vol. 25, no. 3, pp. 407–421, 1974. 127

[84] D.S. Taubman and M.W. Marcellin, "JPEG2000: Fundamentals, Standards and Practice," 2002, Kluwer Academic Publishers. 128

[85] J. Vieron and C. Guillemot, "Low rate FGS video compression based on motion-compensated spatio-temporal wavelet analysis," in *Proc. of the SPIE International Conference on Visual Communication and Image Processing (VCIP)*, July 2003. 128

[86] T. Guionnet and C. Guillemot, "Soft Decoding and Synchronization of arithmetic Codes: Application to Image Transmission Over Noisy Channels," *IEEE Transaction Image Processing*, vol. 12, no. 12, pp. 1599–1609, December 2003. 128

[87] I. Kozintsev, J. Chou, and K. Ramchandran, "Image Transmission Using Arithmetic Coding Based Continuous Error Detection," in *Data Compression Conference*, March 1998, pp. 339–348. 129

[88] "News Releases: Cisco 802.11a Software Defined Radio," 12 September 2005, http://newsroom.cisco.com/dlls/innovators/wireless/index.html. 141

RÉSUMÉ

Cette thèse a pour objectif l'étude des mécanismes d'interaction inter-couche et leurs applications pour des algorithmes adaptatifs pour contrôler les paramètres de transmission. Après une description rapide des protocoles des couches liaison et physiques du standard IEEE 802.11, nous présentons une analyse de performance des modes de transmission spécifiés par les normes 802.11a/b. La deuxième contribution de cette thèse porte sur la modélisation des réseaux 802.11. Nous proposons un modèle analytique qui prend en compte la position des terminaux radio par rapport au point d'accès afin d'évaluer les performances de la couche MAC. Nous proposons ensuite de nouveaux mécanismes de sélection du débit de transmission physique. Nous avons élaboré l'algorithme AARF, basé sur l'algorithme ARF, qui s'intéresse à la classe des terminaux dits à faible latence et offre des services d'adaptation à court et à long terme. Un autre algorithme basé sur les mêmes idées que celles développées pour AARF, mais conçu pour fonctionner avec des systèmes à grande latence, a été implanté et expérimenté sur des terminaux à base du processeur Atheros. Enfin, nous avons élaboré un troisième algorithme de contrôle dynamique du mode de transmission en boucle fermée appelé CLARA qui est compatible avec les standards 802.11a/b/g. La dernière contribution de cette thèse porte sur l'optimisation de la transmission de flots multimédias temps réel sur des réseaux 802.11. Nous présentons MORSA, un mécanisme d'optimisation inter-couche efficace qui permet de sélectionner le mode de transmission à utiliser en fonction de l'état du canal de transmission et des caractéristiques de l'application.

Mots-clés: Réseaux locaux sans fil IEEE 802.11, Réseaux Ad Hoc, Modélisation des couches 802.11 MAC/PHY, Intéraction inter-couches, Sélection du débit de transmission physique, Transmission multimédia dans les réseaux locaux sans fil.

ABSTRACT

The main goal of this thesis is to propose efficient adaptive communication mechanisms using cross layer interactions in IEEE 802.11 WLANs. First, we present a detailed performance evaluation of 802.11a/b PHY layer transmission modes. The second contribution of the thesis concerns 802.11 MAC/PHY layers modelling. An analytical model that accounts for the positions of stations with respect to the access point while evaluating the performance of 802.11 MAC layer, has been proposed. The third contribution of the thesis concerns rate adaptation mechanisms and especially cross layer algorithms between MAC and PHY layers. An adaptive rate selection algorithm, called AARF for low latency systems that improves upon ARF to provide both short-term and long-term adaptation has been proposed. In this field, we also present a new rate adaptation algorithm designed for high latency systems named AMRR that has been implemented and evaluated on an AR5212-based device. We then propose a closed-loop, dynamic rate selection algorithm that can be implemented in all 802.11a/b/g compliant wireless local area networks. This algorithm called CLARA is a culmination of the best attributes of the transmitter-based ARF and the RBAR control mechanisms with additional practical features to facilitate multipath fading channel sensing and feedback control signalling. The last contribution of the thesis is on the optimization of real time multimedia transmission over 802.11 based networks. In particular, we propose a simple and efficient cross layer mechanism, called MORSA, for dynamically selecting the transmission mode considering both the channel conditions and characteristics of the media.

Keywords: IEEE 802.11 Wireless LAN, Ad Hoc Networks, 802.11 MAC/PHY Layer Modelling, Cross Layer Optimization, Physical Rate Selection Mechanisms, Multimedia Transmission in WLANs.