IEEE 802.11b MAC layer's influence on VoIP quality parameters

Measurements and Analysis

JUAN CARLOS MARTÍN SEVERIANO

KTH Microelectronics
and Information Technology

Master of Science Thesis
Stockholm, Sweden 2004

IMIT/LCN 2004-09

IEEE 802.11b MAC layer's influence on VoIP quality parameters: Measurements and Analysis

Juan Carlos Martín Severiano

x01_jcm@e.kth.se

October 2004

Institute of Microelectronics and Information Technology (IMIT)

Royal Institute of Technology (KTH), Stockholm, Sweden

In partial fulfillment of the requirements for the degree of Master of Science

Thesis Advisor: Ian Marsh

Thesis Examiner: Gerald Q. Maguire Jr.

Abstract

Real-time voice measurements were performed to assess whether there are significant problems with 802.11b wireless networks regarding real-time voice communication. We present an analysis of how the 802.11b MAC protocol and diverse environmental conditions affect the quality of real-time voice in terms of loss, delay, and jitter. We also reveal practical issues of wireless monitoring with using passive sniffers in this type of analysis. The results obtained by our measurements show that in the majority of the cases the quality was acceptable, but under some circumstances the requirements for voice communication were not met.

Sammanfattning

Realtidsröstmätningar gjordes för att testa om det finns problem med 802.11b trådlösa nätverk beträffande realtidsröstkommunikation. En analys presenteras av hur 802.11b MACs protokoll och olika tillstånd i omgivningen påverkar kvaliteten på realtidsrösten i form av förluster, fördröjningar och jitter. Även praktiska angelägenheter om trådlös övervakning med passiva sniffers visas. De erhållna resultaten visar att i en majoritet av fallen var kvaliteten acceptabel, men under vissa förhållanden blev inte kraven för röstkommunikation uppfyllda.

Acknowledgements

First and foremost I must thank Ian Marsh, my advisor, for the opportunity he gave me to do this master thesis project and for his continuous and inestimable support, guidance, lessons taught, patience, and also friendship. Special thanks to Victor Yuri Nunes, my thesis project colleague, whose help, motivation, and friendliness made possible the development of this joint project.

I am also very greatful to Professor Gerald Q. Maguire Jr. for his valuable help, feedback, and suggestions and to the people from SICS and LCN for the facilities, equipment, and kind environment provided.

Special thanks go to my all friends. Without them the course not only of this thesis, but also of my Erasmus years would not have been as delightful, amusing, and memorable as it has indeed been. Among them, Samer, Emilio, and Tobias deserve my deepest gratitude for being always there with their friendly smile and support.

And last but not least, I am eternally grateful to my dear family, especially my parents. Thanks to them I had the opportunity to receive one of the best educations one can think of. But above all, mamma and daddy, thanks for your love.

A vosotros dos, Victoria y Juan, va dedicado todo el esfuerzo que he empleado en este proyecto.

Contents

Chapter 1

Introduction

It seems that we need no longer be tied to a cable to be connected to the world. WLANs[1] are rapidly gaining in popularity. The mobility, ease of deployment, low cost, and high capacity offered by the wireless infrastructure offers a highly attractive alternative to existing wired networks.

Voice is already being touted as an important application for these networks, since wireless LANs provide the mobility that wired LANs lack. However, before widely deploying VoIP-capable systems over wireless networks it is essential to know whether such wireless networks are well-suited for voice communication or not. In this project we provide a quantitative answer to this question.

The specific WLANs[2] studied in this thesis are IEEE 802.11b wireless LANs [10]. They provide transmission rates up to 11 Mbps in the 2.4 GHz frequency band. This frequency range was chosen for two reasons: it is license-free (thus, users do not have to pay a fee to use it) and it is also available worldwide. However, even though this is a major advantage there is a clear trade-off, as 802.11b networks have to share this frequency with other technologies, such as microwave ovens, cordless phones, and Bluetooth devices. Since these other devices are very common they can potentially interfere with the WLANs radio transmissions. Another drawback of 802.11b is that it did not specify any explicit QoS support for real-time multimedia traffic. An extension to the standard that will support QoS, the IEEE 802.11e, is currently under development [11].

Nevertheless, these drawbacks have not stopped the wide-spread deployment of this technology in business, institutional, and home wireless networks.

1.1 Motivation and goals

VoIP over wireless networks has been defined by many as the "killer application" in the near future given the recent popularity of wireless local area networks. Thus, a lot of interest and research has occurred for this technology which offers

[1] Wireless Local Area Network - this and other acronyms can be found in the Appendix

[2] The terms WLANs, wireless networks, wireless LANs, and IEEE 802.11b networks will be interchangeably used in the context of this thesis; **all** refer to a IEEE 802.11b local area network.

many (primarily economical) advantages over GSM/UMTS mobile telephony networks; although VoIP over WLANs should be seen as a complement rather than a replacement to the cellular infrastructure. However, WLAN technology is not yet mature and there are still many issues to address before it can be considered as a serious alternative to GSM/UMTS. Therefore, it is highly motivating to examine these promising wireless technologies.

One of the main goals of this thesis is to cover questions that we consider have not been sufficiently examined by recent research, namely performance analysis based on experimental experience and measurements in actual networks. The most current research has focused on simulations and theoretical studies of 802.11 wireless network performance.

The concrete goals of this thesis are to study whether 802.11b networks can properly accommodate voice calls, despite the lack of explicit QoS support. Also, to exam the question of whether fine-tuning of the 802.11b parameters can improve the overall quality of a voice call. Additionally, we had the objective to identify the major impairments that can lead to poor voice quality, either from the environmental conditions (distance or background noise) or from the design of the 802.11b protocol. A last goal is the creation of a data repository containing the data gathered in our measurements.

1.2 Approach

In order to analyse how the inherent design characteristics of the 802.11b protocol affect voice quality parameters we have designed measurement scenarios with different network configurations, load conditions, and environmental situations. The data gathered comprises both the 802.11 MAC performance and the voice quality as seen from the application layer. The link layer performance was measured through passive sniffers that captured the 802.11 frames generated when conducting a VoIP call, whilst the voice quality was evaluated in terms of loss, delay, and jitter.

We investigated the effect of these environments and parameters by trying to isolate their effect as much as possible. With this approach in mind, the parameters that we have studied are:

- Distance, with line of sight between the wireless nodes
- Influence of different bitrates
- Presence of obstacles between the nodes
- Competing traffic
- Differences between ad hoc and managed mode
- The use of the RTS/CTS mechanism to solve the hidden node problem

1.3 Sister project

This investigation has been done in coordination with Victor Nunes, who has conducted his master's thesis project in parallel with this one, but focused on the QoS parameters at the application layer [12]. In contrast, our project is focused on the performance of 802.11b at the Medium Access Control (MAC) layer and how this performance affects the main parameters that determine the quality of voice traffic, namely: loss, jitter, and delay.

Chapter 2

Background

In this section we will give a detailed description of the IEEE 802.11b standard and the VoIP metrics that contribute to voice quality, as well as the relation between these metrics/parameters and the MAC protocol. A description of research related to this thesis is included at the end of this section.

2.1 The IEEE 802.11 standard

2.1.1 Overview

2.1.1.1 History and 802.11 extensions

The IEEE 802.11 standard belongs to the 802 family, a series of standards developed by the IEEE to define specifications for local and metropolitan area networking, mainly at the data-link and physical layers of the OSI reference model. In 1997 the IEEE released the first version of the 802.11 standard, whose purpose was to provide wireless connectivity between different devices in a local area, with a maximum transmission rate of 2 Mbps. Two years later a revision appeared. It included two new extensions which used new modulation schemes to provide rates up to 11 Mbps at the 2.4 GHz frequency band (802.11b) and 54 Mbps at the 5 GHz band (802.11a). Further extensions are being released, addressing aspects such security, higher transmission rates, and quality of service (QoS). The following list summarises the current extensions to the 802.11 standard (some of which are still in draft state) and their main features:

802.11a High speed WLAN standard at the 5 GHz band - supports 54 Mbps.

802.11b WLAN standard for 2.4 GHz band - supports 11 Mbps.

802.11e Defines new schemes for Quality of Service.

802.11f Defines inter-access point communication.

802.11g Additional modulation technique for the 2.4 GHz band, to achieve a rate of 54 Mbps.

802.11h Spectrum management of the 5 GHz band for use in Europe and in the Asia Pacific region.

802.11i Addresses security weaknesses in the original 802.11 standard.

2.1.1.2 802.11b layers

The standard defines a set of medium access control (MAC) and physical layers (PHY) specifications for wireless connectivity. The following figure shows the relation between these layers and the OSI reference model:

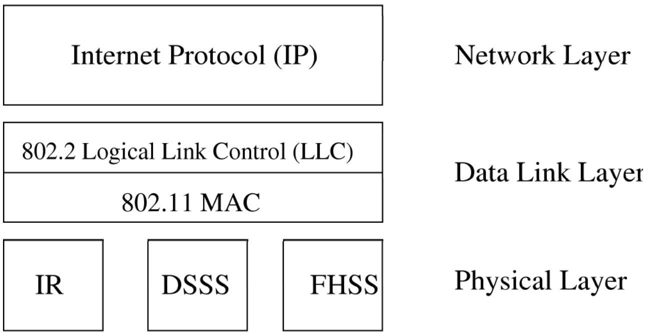


Figure 2.1: 802.11 and OSI layers

As shown in the picture, the MAC is a sublayer of the data-link layer, which offers its service to the logical link sublayer, a common interface for all the IEEE 802 standards. This common interface permits an heterogeneous interconnection of different types of devices by abstracting their underlying media technology.

In order to distinguish between data units of different layers, we will follow the convention of using the name 'packets' to the data units at the IP and higher layers, whilst using the word 'frames' for the link layer and 802.11b data units.

- Physical layer

The physical layer (PHY) specifies low-level communication parameters such as the radio or optical technology, frequencies, channel bandwidth, modulation schemes, and transmission rates. The three boxes shown at the bottom of figure 2.1 represent the three different radio tecnologies that the original 802.11 standard defined for wireless connectivity: IR stands for InfraRed, FHSS for Frequency Hopping Spread Spectrum, and DSSS for Direct Sequence Spread Spectrum.

Whilst all these technologies can support transmission rates of 1 and 2 Mbps, few vendors sell IR compliant products. Also, today frequence hopping products are few in comparison with the number that use direct sequence spread spectrum. An explanation for the dominance of DSSS products is the development of the 802.11b extension that enhanced the basic DSSS mode with additional transmission rates of 5.5 and 11 Mbps.

For spread spectrum techniques the standard defines the S-band ISM (2.4-2.5 GHz) as the frequency range to use. This is because the regulatory authorities permit the unlicensed use of ISM (Industrial, Scientific, and Medical) frequency bands provided that the emitting power of the devices is low and that spread spectrum techniques were used to avoid interference with the primary users of this band.

For direct sequence mode, the standard divides the band into 14 different channels whose mapping to frequencies is shown in table 2.1:

Channel (1-6)	1	2	3	4	5	6	7
Frequency (GHz)	2.412	2.417	2.422	2.427	2.432	2.437	2.442

Channel (7-14)	8	9	10	11	12	13	14
Frequency (GHz)	2.447	2.452	2.457	2.462	2.467	2.472	2.477

Table 2.1: IEEE 802.11b channels

However, this channel distribution causes adjacent channel interference, since the bandwidth used by the 802.11b stations is around 30 MHz and there is only 5 MHz of separation between two adjacent channels. Thus, only channels separated by more than 30 MHz, such as channels 1 and 6 for example, are spread enough to be used in close locations without interference.

This poses a significant challenge to deploy a WLAN with an adecuate coverage while keeping the values of adjacent channel interference and co-channel interference low [27]. Figure 2.2 shows how the frequency band used by each channel overlaps the frequency band of the neighbouring channels.

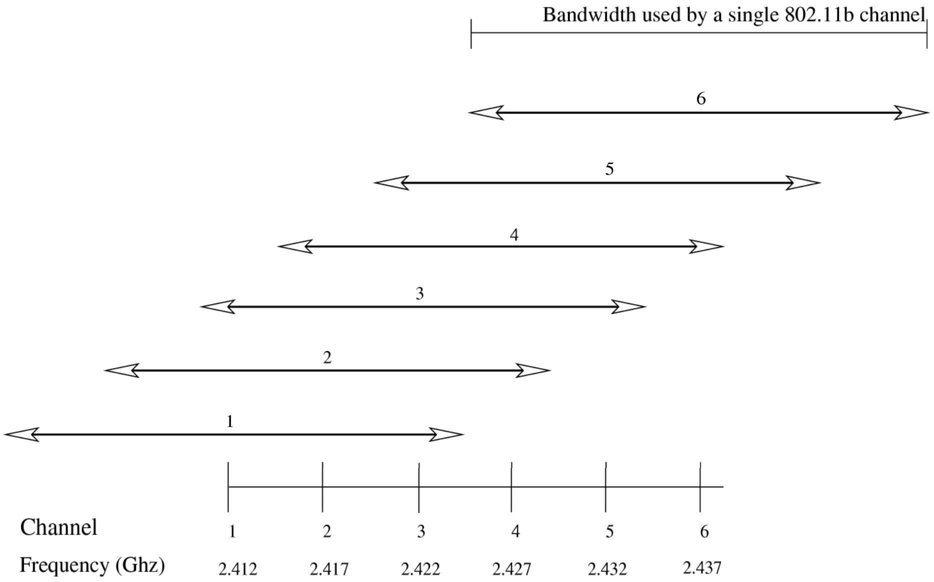


Figure 2.2: Adjacent channel interference

Different modulation schemes are used to achieve the standard bitrates of 1, 2, 5.5 and 11 Mbps. The standard defines the use of BPSK for 1 Mbps, QPSK for 2 Mbps, CCK BPSK for 5.5 Mbps and CCK QPSK for 11 Mbps. However, the denser encodings used to transmit at the highest rates however make them more prone to interference and noise and therefore they have less range than the lower rates. Therefore, a trade-off exists between data throughput and distance and we will see an illustration of this in our experiments.

In order to accomplish interoperability between devices that use different transmission techniques, the PHY layer prepends a physical header to data frames that is always transmitted at 1 Mbps. Further information about the physical layer can be found at [10], section 12.

- MAC layer

The main functions of the medium access control (MAC) layer are to coordinate the stations to gain access to the medium and to define the mapping of physical layer signals to/from link frames. We describe the MAC layer in detail in section 2.1.3.

2.1.2 Architecture

Before describing the two different architectures defined in the the 802.11 standard we describe the four components that a 802.11 network may consist of: An access point, stations, the wireless medium, and a distribution system.

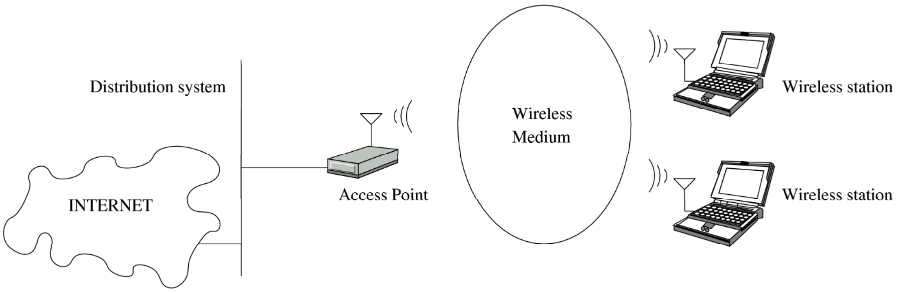


Figure 2.3: Wireless network components

- Station (STA): An electronic device capable of communicating wirelessly with other stations in range.
- Access point (AP): A special wireless station whose main purpose is to provide the wireless stations access to another network such as the Internet.
- Distribution system (DS): In order to provide larger coverage areas, several access points may be used. Then, a distribution system is the logical component of a 802.11 network that allows the different access points to track the location of the wireless stations and to enlarge the coverage area beyond the direct range of a station.
- Wireless medium: While in wired networks a cable is the physical medium used to carry frames from the sender station to the receiver, in wireless networks the physical medium is the air.

2.1.2.1 Infrastructure networks

This is the most commonly used mode of the two modes that the IEEE 802.11b standard defines to build wireless LANs. An infrastructure network relies on an access point to provide connectivity to the wireless stations.

The stations must be in range of the AP, however it is not required that the stations are in range of each other. When this happens, the stations that are not in range of each other are usually referred as *hidden* nodes (see section 2.1.3.9).

It is possible to extend the size of a wireless LAN by interconnecting several APs by a distribution system, thus permitting the wireless stations to roam between adjacent cells.

2.1.2.2 Adhoc (independent) networks

The standard provides a mechanism to create small, usually short-lived networks, with only end stations. A common usage of this is when a connection between two or more stations is desired (file sharing for instance) and no AP is available. However, it is also possible that one of the stations (if equipped with a LAN interface) acts as a bridge between the WLAN and the wired world, providing external connectivity to the other stations.

2.1.3 IEEE 802.11 Medium Access Control (MAC)

The main functions of the medium access control (MAC) layer are to coordinate the stations wishing to gain access to the medium and to define the mapping of physical layer signals to/from link frames. It also manages the bitrate selection and supports different operational modes such as the RTS/CTS handshake. The following sections describe the relevant 802.11b MAC features for the measurements we conducted.

2.1.3.1 MAC access modes

- Distributed Coordination function (DCF)

This is the access mode defined in the 802.11 protocol to provide contention-based access to the medium, through the CSMA/CA protocol (described in section 2.1.3.4).

This contention-based access to the medium results in random delays between each frame transmission, which may be problematic for real-time traffic if the delay increases beyond the limits defined in section 2.2.2.

We have used DCF as the access mode in our measurements, as it is by far the most widely used access mode in 802.11 wireless networks.

- Point Coordination Function (PCF)

The PCF mode is an optional access mode which enables polled transmission of data frames. In this mode, the AP polls the wireless stations granting them access to the medium for a short period of time. Then, the AP moves to the next station in the poll list and thus all the stations obtain a slot of time to transmit data.

Although this access mode seems suitable for real-time communication, it simply is not supported by many 802.11 devices. Further information about PCF mode can be found in [10, section 9.3].

2.1.3.2 Positive acknowledgement

The 802.11 standard defines a positive acknowledgment schema in order to provide some reliability for wireless transmissions. All unicast data frames must be acknowledged.

When a station has properly received a data frame it sends back an ACK frame to the sender so that the sender knows of the successful delivery. If for some reason the ACK frame does not arrive at the sender it will assume that the packet was not delivered and the sender will retransmit it.

The drawback of the ACK mechanism is the overhead that it adds to the communication resulting in a loss of throughput and also additional delay. However, ACK frames are necessary since link conditions are highly variable in wireless networks.

2.1.3.3 Timing - Interframe spacing

The 802.11 standard defines four interframe spacings to prioritise the transmission of certain frames. They are used to ensure that atomic operations such as the frame-acknowledgement pair or the RTS/CTS handshake are not interrupted; they are also used to provide preference to contention-free traffic over contention-based traffic when both exist.

When the medium is busy all the frames have to wait until it becomes idle, in order to be transmitted. Then, the frames with the highest priority gain access to the channel as they are assigned a shorter interframe space. Figure 2.4 shows the different interframe spacings.

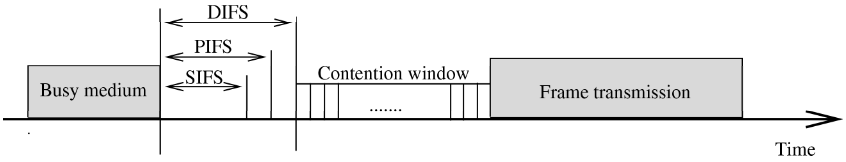


Figure 2.4: Interframe spacing and contention window

- SIFS (Short InterFrame Space)

The SIFS is used for high priority frames such as acknowledgements or CTS frames, which must be sent inmediately after the corresponding frame. It has a duration of 10 μs in DSSS.

- PIFS (PCF InterFrame Space)

The PIFS is used to give higher priority to PCF contention-free traffic over DCF contention-based traffic. It has a duration of 30 μs in DSSS.

- DIFS (DCF InterFrame Space)

The DIFS is the minimum time that the medium must be idle before attempting a contention-based transmission. It has a duration of 50 μs in DSSS.

- EIFS (Extended InterFrame Space)

The EIFS (not shown) is a variable-length interval that is used when a frame is received with errors.

2.1.3.4 Contention using DCF - backoff mechanism

The DCF mode is based on the carrier sense multiple access with collision avoidance (CSMA/CA) protocol. Stations must sense the medium to determine whether it is busy or idle before they attempt to transmit a frame.

In order to resolve contention between several stations waiting for the channel being idle, CSMA/CA defines an *exponential backoff* algorithm. The backoff mechanism works as follows:

1. Every time a station attempts to transmit, it waits for the medium to be idle during a DIFS (or a EIFS, if the transmission was not successful) period. Then, it follows an interval called the *contention window* which is divided into slots of 20 μs. The stations choose a random number of slots and then they wait for these slots to elapse, thus accessing the medium is done by the station that selected the lowest number of slots. The timing of the backoff mechanism is shown in the figure 2.4.

2. During the occupation of the channel by the winning station the other stations suspend their backoff procedure until the medium is idle again. However, the stations resuming the contention do not choose a random number of slots again. Instead, they wait for the slots that remained in their previous contention. In this way, the stations that lost the contention have a higher probability of gaining access to the medium than the one that just transmitted.

3. The random number of slots is chosen from the interval [0, CW], CW being the size of the contention window. This mechanism is called exponential backoff because after a failed transmission a station must double the size of the contention window. This is designed to reduce the collision probability in a heavily loaded network.

4. Collisions occur when two stations select the same number of slots. A collision is only detected by the lack of the corresponding ACK, since the wireless stations cannot listen to the medium whilst they are transmitting.

The contention window size is a power of 2 minus 1, starting at 31. It doubles after each failed transmission until the 5th retransmission, where it is limited to 1023 slots for further retransmission attempts. The contention window reverts to the minimum size after a successful transmission of a frame.

2.1.3.5 Retry counters

The retry counters set a limit on the maximum number of retransmissions allowed per frame before it is discarded by the MAC layer. Some cards define different counters depending on the packet size.

2.1.3.6 Error detection and recovery

When a station does not receive the corresponding ACK it assumes that the frame was lost and it tries to transmit it again. Thus detection and recovery at the MAC layer occurs at the source.

However, after reaching the maximum number of retransmissions the MAC layer discards the frame. It is therefore responsibility of higher layers to perform simply detection or to provide a recovery mechanism in the case where reliability (TCP, for instance) is required.

2.1.3.7 Carrier sensing functions

Carrier sensing is a method invoked by the MAC layer to ascertain whether the medium is busy. It can be either physical or virtual. Physical carrier sensing is performed by the physical layer and it reports the state of the medium to the MAC layer. Virtual carrier sensing is made through the Network Allocation Vector (NAV), which is the expected time that the atomic transmission of a frame will maintain the channel busy. The NAV is calculated from the duration field of the 802.11 header of the existing frame in the air.

2.1.3.8 Bitrate selection

Bitrate selection has a critical role in wireless systems as it directly affects the frame error rate for a given signal-to-noise ratio. The 802.11 standard intentionally left the bitrate selection mechanism unspecified. Thus, vendors have freedom to implement their own mechanisms [29]. However, some drivers permit the user to select specific bitrates.

2.1.3.9 Hidden node problem. RTS/CTS mechanism

In a wireless network transmission ranges have fuzzy boundaries, unlike in a wired network where all the stations can reach all others. It may happen that two stations are in range with a third but not in range with each other, because of some obstacle or simply because they are too far apart.

Figure 2.5 reflects this situation: The pairs of nodes A - B and B - C are in range. However, A cannot hear what C transmits, nor can C hear what A transmits. Thus, A and C may simultaneously start a transmission causing a collision at B because they cannot sense the medium as busy.

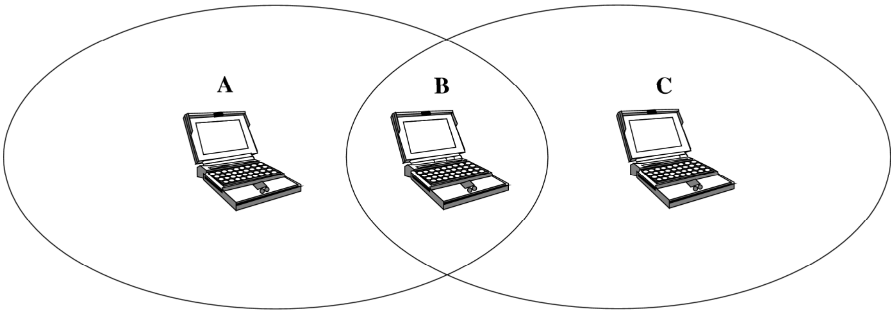


Figure 2.5: Hidden node problem

In order to alleviate this problem the IEEE standard provides a mechanism to reserve the medium, the Request To Send (RTS) and Clear To Send (CTS) messages. The procedure works as follows: A sends a RTS frame to B, which responds with a CTS. Although C cannot hear the RTS it will receive the CTS, thus knowing that a transmission is about to occur. The CTS frame has a duration field that informs all the stations in range about the time that the medium will be busy. Then, after receiving the CTS A transmits the data frame while C waits for the transmission to finish.

RTS frames are very small (20 bytes) and are less likely to be interrupted than a large data frame (which can be up to approximately 1500 bytes). However, when a small data frame is to be transmitted this mechanism does not offer any significant advantage. Additionally, the RTS/CTS reservation adds some overhead to the communication thus reducing the overall throughput of the network.

2.2 VoIP quality metrics

There are several metrics to measure the quality of a voice call. Some rely on the subjective ranking provided by 'expert' listeners who give a score to the quality of a recorded voice call (MOS), while others perform an automatic measurement of the quality (PESQ) [31]. However, rather than using these quality assessment techniques we have collected data of the network parameters that have a major impact on the voice quality. These parameters are loss, delay, and jitter.

2.2.1 Loss

A loss can be defined, in the scope of multimedia applications, as a packet that never reaches its destination, or a packet that arrives too late and thus cannot be used to play out the multimedia content in time. Most real-time multimedia applications are loss-tolerant, i.e. they can still provide good perceived quality to the user even though some packets are not delivered to the application. Additionally, there are numerous loss concealment techniques that provide some robustness for packet loss.

2.2.2 Delay

Delay is the time that it takes the voice to travel from the sender to the receiver. This parameter is important in voice communication because high one-way delays lessen the interactivity. In other words, with high delays none of the participants knows when the other has started or finished talking, leading to both users either being silent or talking simultaneously. The ITU-T defines, in the G.114 recommendation, a limit of 150 ms for high quality voice communication, while others report 200 ms as unnoticeable end-to-end delay [30].

There are several parts in the system that contribute to the total mouth-to-ear delay. For instance, the speech coder, the operating system, the network, or the jitter buffer. We will analyse the contribution of the wireless MAC protocol to the mouth-to-ear delay.

2.2.3 Jitter

Jitter is defined as the variance of inter-packet arrival times compared to the inter-packet times of the original transmission. A buffer is commonly used to absorb this variation at the cost of some additional delay. Jitter's effect in voice communication is undesirable since it can lead either to additional packet loss or to additional delay.

2.3 MAC layer versus VoIP quality

While 802.11b networks have proved their appropriateness for best effort traffic, their lack of QoS support makes it questionable whether real-time multimedia applications, such as voice communication, can be used with reasonable quality. Thus, since packet loss, delay, and jitter are of such importance in order to obtain a good perceived quality in (successful) voice communications, it is critical to know what causes the deterioration of these parameters in a wireless network, and how these parameters are affected by the 802.11b design.

Whilst in wired networks the deterioration is generally caused by congestion, in wireless networks it can be caused both by degradation of the signal (due to fading or interference) and/or congestion. The 802.11b design provides reliability when the environmental conditions are poor. Such reliability is achieved through mechanisms such as the positive acknowledgement, bitrate selection, and RTS/CTS. It also provides some means to avoid collisions (backoff mechanism).

However, the protocol design at the MAC layer does not provide any explicit QoS support [2] to limit the delay or jitter. Likewise, not even the best protocol design can prevent the frame loss caused by weak signals when a node is too far from the destination.

2.4 Related work

There is a large amount of literature regarding the performance of 802.11b networks. Most of it focuses on simulations and modeling, but there are also results about experiments with real networks. In addition, several enhancements to the standard have been proposed in order to provide improved QoS for real-time multimedia applications. In the next paragraphs we present a brief summary of some of these articles and papers, with their conclusions and results.

2.4.1 Performance analysis of IEEE 802.11b

- In "An Analysis of Short-Term Fairness in Wireless Media Access Protocols" [1], E. Koksal, et al. analyse the short-term fairness of several medium access protocols and their conclusion is that CSMA/CA is not especially fair compared to other mechanisms. However, they state that there is a good trade-off between throughput and fairness. They also affirm that poor short-term fairness can significantly affect real-time audio and video in terms of jitter.

- In "Analysis of 802.11b MAC: A QoS, Fairness and Performance Perspective" [2], S. Sharma highlights IEEE 802.11b issues, namely QoS, fairness, and performance. Regarding QoS, he remarks that the DCF mode provides only best effort service, and no guarantees are given in terms of delay and jitter. In relation to Koksal, he states that the unfairness of CSMA/CA is due to the fact that collisions in wireless LANs are asymmetric. He also analyses some of the causes that reduce performance in 802.11b, like the RTS/CTS/ACK mechanisms or the slower transmission rate of headers (physical layer headers are always transmitted at 1 Mbps). He suggests some improvements that could be applied at different levels (physical, MAC, UDP, and TCP).

- In "Performance Analysis of the IEEE 802.11 Distributed Coordination Function" [4], an analytical study is made about DCF's performance. The author evaluates both the basic and RTS/CTS access mechanisms and concludes that RTS/CTS should be used in almost all practical cases. This is due to its superior capabilities in terms of coping with the hidden node problem and performance in large network scenarios.

- In a paper about fairness "Operational and Fairness Issues with Connection-less Traffic Over IEEE 802.11b" [5] from Theo Pagtzis, et al., the authors present an experimental evaluation of IEEE 802.11b giving analytical results of performance, throughput, and error rates obtained in different scenarios. An interesting conclusion is the unfairness of the protocol, due to the fact that it "sets a lowest common denominator of transmission rate for all the stations associated with the AP of a single cell, which is inherently unfair for any 802.11b station operating at signaling rate above 1 Mbps"

- In "Scheduling time-sensitive traffic on 802.11 wireless LANs" [6], Martin Heusse, et al. show that hosts generating time-sensitive traffic in a 802.11b cell may benefit from low delays even in saturation conditions. This is true if the packet rate is kept under a value that depends on the number of transmitting nodes and the type of traffic they transmit.

These and other papers referred to in the reference list reveal that there are issues regarding QoS in wireless networks that justify a deeper study of such networks. Two of the major contributions of this thesis are the data collected from experiments carried out over actual wireless networks and the analysis of the data. With both components we will see how the issues pointed out in these papers manifest themselves and how they affect loss, delay, and jitter.

2.4.2 Voice over IP in WLANs: Measurements and simulations.

- In "Voice transmission in an IEEE 802.11 WLAN based access network" [7], Köpsel, et al. compare the suitability of the DFC and PCF protocols for audio transmission, via simulations, and they conclude that PCF performs better than DCF at higher transmission rates.

- In the paper "Capacity estimation of VoIP channels on Wireless Networks" [8], Patel et al. study the capacity of WLANs for VoIP channels and shows an upper bound on the number of successful VoIP calls that a typical 802.11b cell can handle. In this paper, the authors present measurements in an actual WLAN and compare the results with ns2 simulations.

- In the paper "Internet Telephony Over WLANs" [9] Dimitrou et al. address the issues that can make the deployment of multimedia communications difficult in WLANs. They give an overview of VoIP and wireless LANs, and also provide results of several experiments done in different scenarios. One of these experiments is the measurement of the influence that mobility and interference have on jitter and loss.

- In "Measuring the Impact of Slow User Motion on Packet Loss and Delay over IEEE 802.11b Wireless Links", [24] Christian Hoene et al. show the effect of motion on the performance of wireless links through a series of experiments with moving nodes. They conclude that other factors like modulation type, quality of power supply, environmental setup, and number of retransmissions may have more impact on 802.11b performance than the motion itself.

- "Measuring Traffic on the Wireless Medium: Experience and Pitfalls" [17]. In this paper, J. Yeo et al. identify losses in packet capture points (sniffers) as one of the major problems in wireless monitoring and measurements. We have observed such losses empirically and based on their findings we have used several sniffers to reduce the overall frame capture losses.

Chapter 3

Measurements - Methodology

3.1 Equipment

In order to know which equipment is needed to perform the experiments, we first have to know exactly what we want to do. This was described previously in section 1.2. Here we describe specific equipment that we selected (or that there was available) to carry out the goals of this thesis.

3.1.1 Hardware

- 6 DELL Latitude D500 Laptops with a Intel PRO/Wireless 2100 module.
- 6 PCMCIA Lucent/Agere/Dell Truemobile 1150 wireless cards.
- 1 D-Link DWL 650 wireless card.
- 1 D-Link DI-614+ access point.

3.1.2 Software

- Fedora Core 2 linux (operating system) [49].
- Sphone, a custom VoIP tool developed at the Laboratory of Communication Networks (LCN) and the Swedish Institute of Computer Science (SICS), whose functionality has been extended during the course of this thesis. It provides real-time information about the major voice quality parameters at the application layer, i.e. percentage of RTP packet losses, jitter, and round-trip delay. It also creates a log file for off-line analysis.
- Ethereal 0.10.6 is a network protocol analyser [45]. We have used this tool to capture all the traffic exchanged by the wireless nodes, both at the network layer and at the data-link (MAC) layer, which includes all the 802.11 frames not seen when a wireless device runs in non-monitor mode (see section 3.2)
- Matlab 6, a well-known software for analysis and statistical calculation [46].

- Wireless tools v26, an open source set of tools to configure the wireless devices [47].
- Nttcp 1.47, a traffic generator that we used to send the UDP/TCP flows in the competing traffic experiments.
- Drivers: Hostap 0.1.2 (for the D-Link DWL card and the Centrino interfaces) and Orinoco 0.13d for the Lucent Orinoco cards.

3.2 Frame capture - Monitor mode

In order to carry out an analysis of the characteristics of the wireless medium and the MAC protocol we need to examine the link layer headers and 802.11b frames that are hidden to the network layer. For this purpose some wireless devices support an operational mode called 'monitor', which enables the capturing of MAC headers and frames. However, when these devices operate in this mode they cannot be used as regular network devices, since the monitor mode disables frame transmission. Thus, they can only be used as 'sniffers', devices that passively observe and capture the traffic sent by the nearby stations. This also includes the stations that do not belong to the network under study, but are close enough to be received. The major advantage of this wireless monitoring is that it can be put into operation without interfering with the wireless network under study. The obvious drawback is that at least one additional wireless device is necessary to monitor the wireless medium.

3.2.1 Monitor modes - 'Prism' (physical) header

There are two monitor modes permitted by the drivers we used. One permits the capture of the 802.11 MAC headers. The second permits the capture not only of the 802.11 headers but also of a header called "Prism" that includes some physical layer information. We used the Prism headers to record the bitrate of each frame. This requires a WLAN card implemented using the Prism chipset.

3.2.2 Monitor mode capturing problems

An additional drawback of sniffers in monitor mode, as reported in [17], is that there is no guarantee that they can capture all the frames in the air, no matter how close they are to the sniffed stations and how free from interference the environment is. In normal mode operation (managed or ad hoc), the ACK mechanism enforces a retransmission when the destination node fails to receive a frame, since the sender detects the lack of a corresponding ACK. However, a sniffer in monitor mode does not acknowledge received frames (this is why we call it passive sniffer) and if it fails to capture a frame it misses the only opportunity that existed.

The practical result of this problem is that the usefulness of the sniffer is dramatically reduced if it cannot capture nearly all the frames that exist in the air. According to [17], the frame loss rate is relatively high and variable, they propose the use of multiple sniffers to reduce the overall amount of losses based on the belief that it is very unlikely that a frame is lost by several sniffers simultaneously. By merging the capture traces of all the sniffers we obtain a single capture file in which the losses that occurred at one sniffer (which would appear as 'holes' in the capture traces) are covered by the other sniffers that hopefully did not miss the same frame.

We ran a simple experiment to prove that it is indeed very unlikely that several sniffers miss a frame simultaneously and thus the aggregated trace effectively fills the holes left by each sniffer trace. The idea is the following: One wireless station sends voice packets to a sink, and two sniffers capture the traffic. Then we compare the traces given by both sniffers and we estimate the probability of a simultaneous loss.

As the default number of retransmissions was set to three, this creates a small problem to analyse the probability of joint losses. If both sniffers report zero transmissions for a certain packet then there is no way to know whether there were actually one, two, three or even four transmissions, and each of these cases leads to a very different result of how independent the frame losses are from one sniffer to the other. A simple solution to this problem is to set the maximum number of transmissions per packet to only one. However, this generated an additional difficulty since the orinoco driver of the wireless cards that we initially had available did not allow us to change the number of transmissions. We got another card, a D-Link DWL 650, that used a different driver (hostAP) and that allowed us to set the number of transmissions to one.

The results obtained from a 22 minutes voice session (long enough to provide some statistical validity) with 65535 voice packets sent, are the following:

% single loss, sniffer 1	% single loss, sniffer 2	% simultaneous loss
0.4502	0.8164	0.0366

Table 3.1: Sniffer's loss

In order to know when a frame was lost we looked at the gaps in the RTP sequence numbers.

These percentages show that, in the conditions of this experiment, the probability of missing a frame simultaneously by two sniffers is very small (approximately only one out of 2700 MAC frames). The probability of missing a frame using only one sniffer is not high, as it leads to an error of less than 1% in the measurements[1].

3.3 Test bed

The basic configuration of our experiments utilises one node to send a flow of VoIP packets and another node to act as a receiver of those packets. In order to observe and capture the traffic existing in the air, we decided to use two monitoring devices (sniffers) placed close to the sender station for the reason mentioned previously in section 3.2.2. We did not plan to study the traffic sent back from the receiver (mainly ACK frames). However, we included another sniffer close to the receiver to check that this node was sending ACK frames

[1] Not only are the missed frames a problem, but also the corrupted ones that ethereal captures (due to collissions), that provide incorrect information. See section 4.4.2.

back, and also to capture the traffic that might have interfered with the signal at the receiver. Figure 3.1 shows this basic set-up.

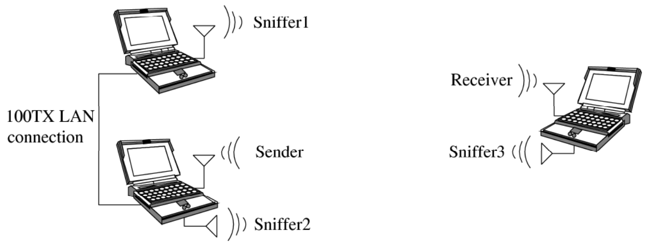


Figure 3.1: Test bed

3.3.1 Node 1 - Sender

Node 1 has two wireless devices. One PCMCIA Lucent card used to send voice traffic and one built-in Centrino wireless device used to capture in monitor mode the traffic in the air (sniffer 2), mainly the outgoing traffic from the sender. Another laptop connected to the sender through a crossed 100TX Ethernet cable was used as sniffer 1, which captured the 802.11 traffic using the Centrino wireless device.

3.3.2 Node 2 - Receiver

The node 2 has also two wireless devices: a Lucent card for the receiver functionality and a Centrino wireless device (sniffer 3).

3.3.3 Description of the voice sessions

During the measurements, a set of fifteen VoIP calls of 80 seconds per specific experiment were generated[2]. However, in some experiments we had to reduce the number of calls due to time or battery restrictions. The audio calls were encoded using a G.711 μ-law PCM CODEC with a sample rate of 8 Khz and 8 bits per sample generating traffic of 64 Kbps. A payload of 160 bytes is generated every 20 ms and then encapsulated into a Real-time Transport Protocol (RTP) data packet. The calls were unidirectional and neither silence suppression nor signaling were used.

3.3.4 Configuration of the wireless devices

We selected channel number 13 for our experiments because it is the least likely to suffer interference from neighbouring WLANs as the most populated channels are 1, 6, and 11. Additional configuration parameters were selected as follows:

- No WEP encryption
- Transmission bitrate: automatic (except for the experiments regarding bitrate)

[2] Eighty seconds was used as it represents the length of a typical business call.

- Maximum number of 802.11 frame transmissions per IP packet: 4 (this implies 3 retransmissions)
- RTS/CTS: off (except for the experiments regarding RTS/CTS)
- Mode: ad-hoc or managed depending on the particular experiment

3.4 Computation of loss, jitter, delay, and MAC transmissions

3.4.1 Application parameters

3.4.1.1 Loss

Every RTP packet has a sequence number. Thus, a loss is detected when there was a gap in the sequence number list. It can be computed either from the Sphone's log files or from the Ethereal's captures recorded at the non-monitor receiver.

3.4.1.2 Round-Trip Time (RTT)

The RTCP protocol (implemented by Sphone) provides round-trip times with a periodicity of one value per second. They were recorded in the Sphone's log files.

3.4.1.3 Jitter

Jitter was calculated from the RTP packet timestamps taken by Ethereal in the receiver, according to the following formula:

$$J_i = T_i - T_{i-1}$$

T_i and T_{i-1}being the timestamps of two consecutive RTP packets.

3.4.2 MAC parameters

3.4.2.1 Number of (re)transmissions

With the device in monitor mode Ethereal captures all the 802.11 frames generated by the MAC layer. Thus, the number of transmissions per RTP packet is the number of all the frames that have the same RTP sequence number.

3.4.2.2 One-way delay and retransmission delay

We calculated both the one-way delay and the delay added by each of the frame transmissions with the timestamps taken by Ethereal from both the application layer and the MAC layer. Figure 3.2 shows the time when the timestamps are taken and the station that records them (t2 means that the timestamp was taken by the sender and t2' means that the timestamp was taken by the receiver).

At the sender, the timestamp that Ethereal takes of a packet from the application layer is the time when the kernel first saw the packet (t1). Then,

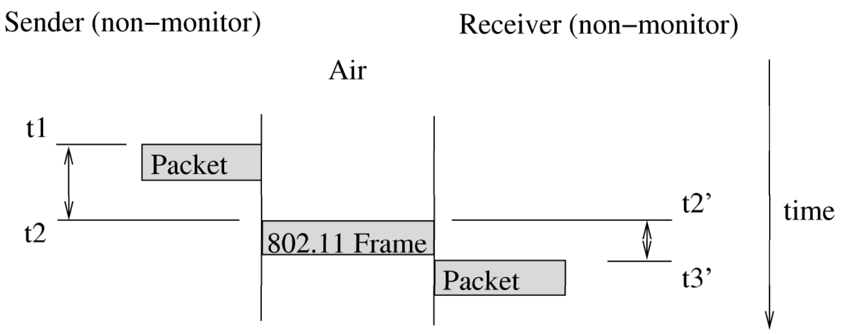


Figure 3.2: Timestamps taken by Ethereal

the packet is sent to the MAC layer to be delivered to the destination. The next timestamp for that packet is the time when the first 802.11 frame transmission was captured in the air (t2 and t2'). As t1 and t2 were taken in the same laptop (the sender) there was no need for time synchronisation, since both instances of Ethereal were running with the same clock reference. The same applies for t2' and t3'.

Thus, $t2 - t1$ is the time that the frame has to wait at the sender before it is finally transmitted over the air. On the other hand, t3'-t2' is the time it takes the 802.11 frame to be delivered from the air to the application layer at the receiver. Then, the network one-way delay, that is the time it takes a packet to go from the application in the sender to the application in the receiver, is:

$$\text{One-way delay} = t2 - t1 + t3' - t2'$$

We observed that $t3' - t2'$ was negligible ($< 50\ \mu$s) compared to $t2 - t1$ ($>$ 1 ms). The reason is that the sender finds contention to put the frame in the air while the receiver simply delivers the frame from the air to the application. Therefore in our calculations we only used t1 and t2 to calculate the one-way delay.

In the case where there are retransmissions, the timestamps taken are the times when the retransmissions were seen in the air. The one-way delay for a RTP packet that needed several retransmissions is calculated by the same formula, but in this case t2 is the timestamp of the last retransmission. From now on we will use t2 as the timestamp of the last transmission and $t2_n$ will be the timestamp of the n-th transmission.

Thus, the one-way delay and the delay of each of the transmissions are calculated as shown in the following table:

One-way delay	$t2 - t1$
1rst transmission delay	$t2_1 - t1$
2nd transmission delay	$t2_2 - t2_1$
3rd transmission delay	$t2_3 - t2_2$
4th transmission delay	$t2_4 - t2_3$

It is worth noting that the results of one-way delay obtained through the

MAC analysis always provide an upper bound to the real delay seen at the destination. This is because the source station may miss some of the ACK frames and thus it will retransmit an unnecessary frame. In this case, the extra frame retransmission is still counted as additional delay in our calculations, which increases the delay obtained through this method. In this thesis we have not studied the percentage of transmissions sensed incorrectly as unsuccessful but it would be possible to investigate this using the traces we collected.

This method permits a fine-grained calculation of the one-way delay. It provides more values, as much as one per 802.11b frame, than the RTT calculation described in section 3.4.1.2. However, we will see in the next chapter that this way of calculating the one-way delay is not always going to be feasible.

Chapter 4

Results - Analysis

In this chapter we will show the behaviour of the MAC layer in different environmental situations and its impact on loss, jitter, and delay. The results shown in this section have been obtained from experiments performed during the course of this thesis.

The common structure for all the experiments is the following: first, we introduce the set-up and the purpose of each experiment. Second, we present the results of the MAC layer obtained in our measurements, followed by a discussion of the results. Finally, loss, delay, and jitter are also presented and discussed.

We will start by showing a single call from the MAC layer point of view, and a brief insight into its repercussions on the voice quality parameters. Next, we will show the results of all the experiments according to the structure mentioned above. Such results usually show the average values of a whole set of VoIP calls made under the same conditions.

The parameters studied in this thesis were bitrate, distance, ad hoc vs. infrastructure mode, TCP/UDP competing traffic, and the usage of the RTS/CTS mechanism.

4.1 Anatomy of a single call

In this section we present a typical call with respect to the number of MAC frame transmissions. In figure 4.1, the x-axis represents the sequence number of each RTP packet, while the y-axis represents the number of frame transmissions for each packet. Additionally, the circles show the occasions when a packet loss occurred. The vertical position of the circles has no meaning.

Figure 4.1 shows that most of the frames were delivered at the first attempt, and a few needed two, three, or four transmissions. Three packets were lost after being transmitted four times.

Even though delay and jitter are not shown they are *qualitatively* included in the plot, with the following relation: the greater the number of transmissions for a specific packet, the longer time the packet took to arrive at the destination. Similarly, jitter is caused by the varying number of transmissions between two consecutive RTP packets. Delay and jitter also have to include the effect of (1) waiting for the medium to be idle and (2) contention.

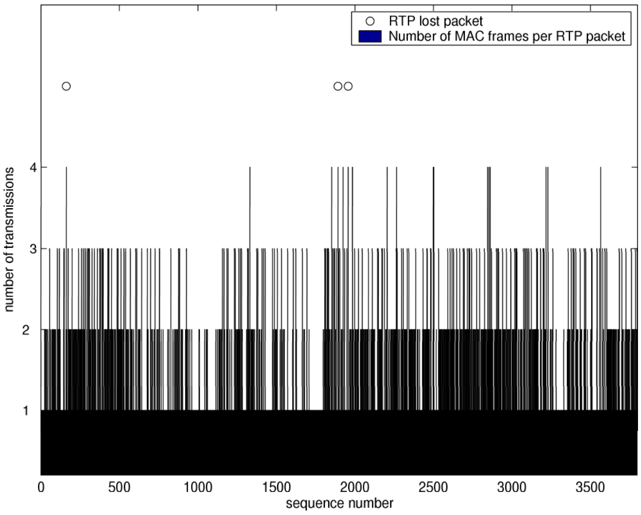


Figure 4.1: MAC frame transmissions per RTP sequence number

We will consider if the additional delay and jitter are significant when we look at the details of the retransmissions.

4.2 Transmission rate experiments

4.2.1 Set-up

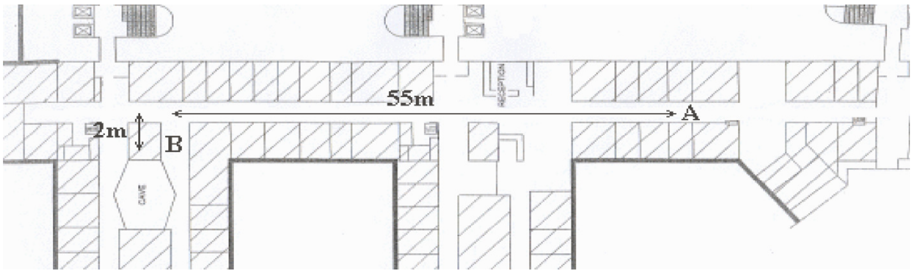


Figure 4.2: Location of nodes for the bitrate experiment

In this experiment we evaluated the effect that different bitrates have on the voice quality parameters. For this purpose we chose a placement where the sender and receiver nodes were not in line of sight, but where it was possible to establish an ad hoc network communication between the two nodes at 11 Mbps.

We looked for a pair of positions resulted in the link quality being sufficiently poor to cause a significant number of retransmissions at 11 Mbps. The purpose of this experiment was to verify whether reducing the bitrate increases the probability of successfully delivering a frame.

We placed the sender and the receiver stations at the points marked as A and B in figure 4.2. Office walls were the obstacle that the radio waves had to overcome.

Considering that the location is an office it is possible that some spurious traffic could interfere, but the experiments were made late at night to minimise this effect as much as possible. We recorded nevertheless some beacon frames from nearby access points. However, these frames had a negligible impact on our measurements.

We made four different experiments with the bitrates defined in the 802.11b standard, namely 11 Mbps, 5.5 Mbps, 2 Mbps, and 1 Mbps.

4.2.2 Transmissions histogram

The histograms of the number of frame transmissions per RTP packet, shown in figure 4.3, reflect the probability of transmitting a data frame successfully increasing when decreasing the bitrate. At 11 Mbps around one third of the packets had to be retransmitted, whilst at 5, 2, and 1 Mbps the retransmissions decreased to less than 5%, with the last three bitrates showing similar performance.

One can conclude that 5.5 Mbps was the best option at that specific location of the nodes, since 2 and 1 Mbps rates did not offer significally lower frame error rates and they would incur higher channel occupation.

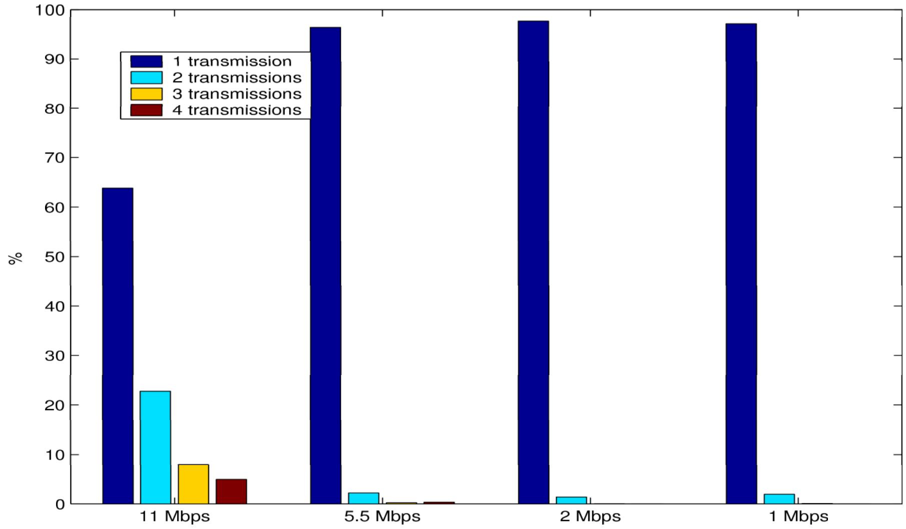


Figure 4.3: MAC frame transmissions histograms at different bitrates

4.2.3 Loss

One of the major foci of this thesis is the relationship between MAC layer performance and application layer performance. We will make a comparison between the MAC transmission histograms and the loss detected at the application layer to verify whether this relationship is significant.

The number of occasions that the RTP packets led to four 802.11b frame transmissions provides an upper bound to the packet loss, as a packet loss can only happen when the packet is discarded after four failed transmissions, although not all the occasions with four transmissions led to a loss. Thus, the percentage of packet loss (indicated by crosses in figure 4.4) must be lower than the percentage of four frame transmissions[1] (indicated by circles). However, in the same figure we can see that the percentages of packet loss for 5.5, 2, and 1 Mbps are above the limits imposed by the percentage of four transmissions.

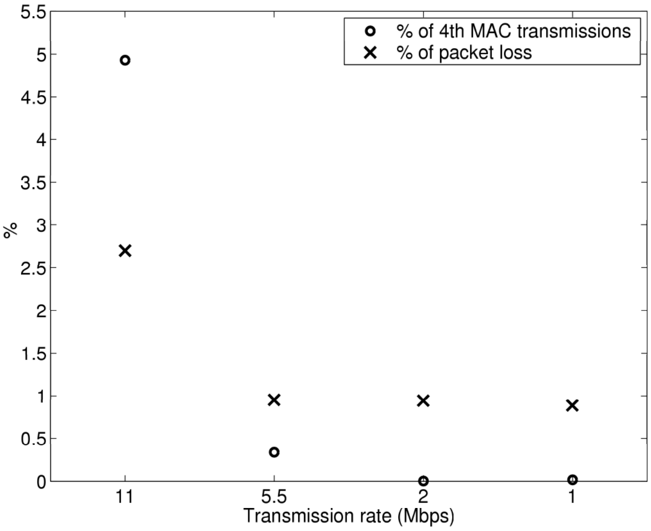


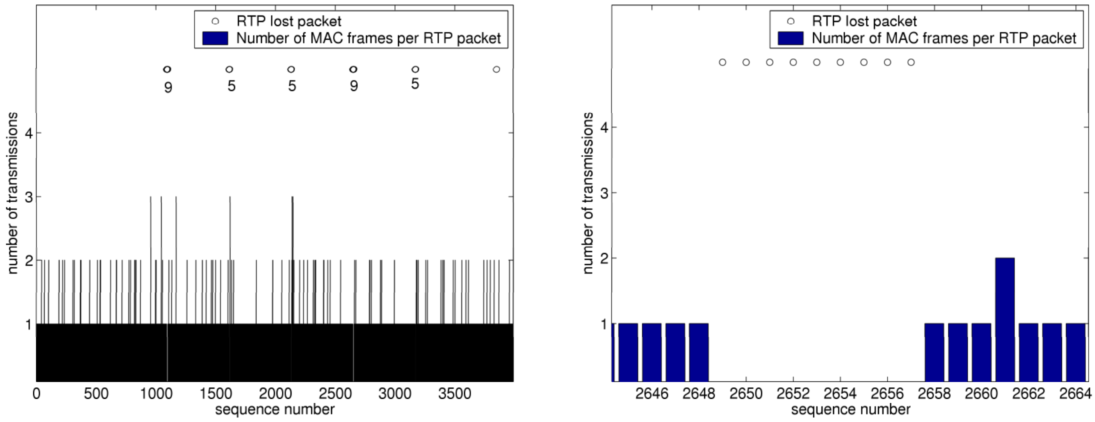
Figure 4.4: Packet loss against transmission rate

We looked for an explanation for this anomaly and we found it by looking at the captured MAC traces. At certain instants we observed that the sender stopped sending frames until it received a response to a *probe request frame* [2], meanwhile discarding every RTP packet that the application layer sent. The plots in figure 4.5 represent the performance of a call made at 5.5 Mbps (subfigure a) where this anomaly occurred and a close-up of one of these packet loss bursts (subfigure b). The numbers under the circles indicate the burst length. Obviously, the frames that were **not sent** appeared as packet losses in the receiver.

A Ethereal snapshot (figure 4.6) illustrates this behaviour. According to the ethereal capture, after the reception of the ACK corresponding to packet number 2648 the sender station starts an active scanning of the medium. Then, after several probe requests, the sender seems to recover the connection and starts to send voice packets again.

[1] One might argue that it would be better to compare the 4th frame transmissions that did not receive an ACK from the receiver with the actual packet loss, but this would increase the complexity of our analysis. Moreover, the sniffers were to capture the outgoing traffic, but not the incoming (i.e. ACK incoming frames). Another reason to prefer the upper bound approach is that we can never be sure that an ACK sniffed in the vicinity of the sender station was properly received by the sender, since the actual sensitivity of the wireless devices used for transmission and sniffing may vary.

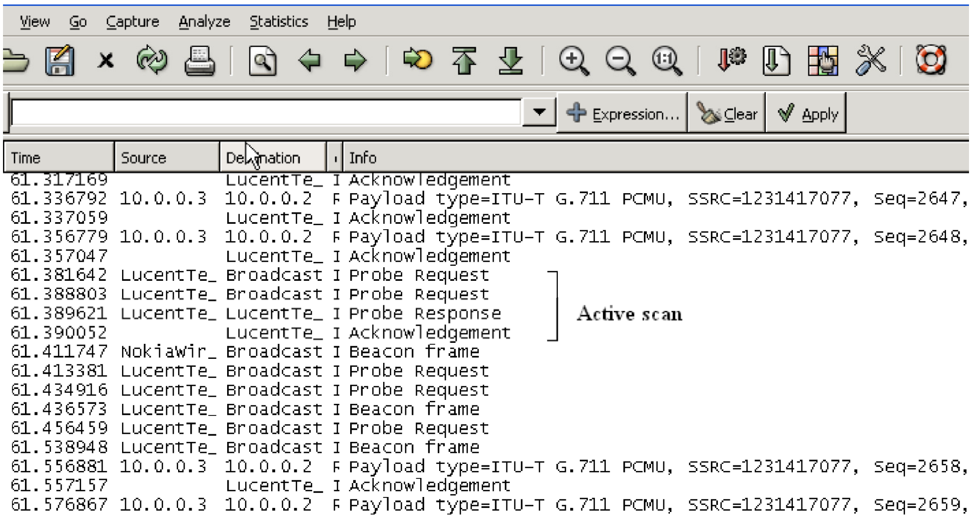
[2] A probe request frame is sent when a wireless station performs an active scanning of the medium, in order to look for available networks to associate with.



(a)

(b)

Figure 4.5: MAC frame transmissions per RTP sequence number



```
61.317169          LucentTe_ I Acknowledgement
61.336792 10.0.0.3 10.0.0.2  F Payload type=ITU-T G.711 PCMU, SSRC=1231417077, Seq=2647,
61.337059          LucentTe_ I Acknowledgement
61.356779 10.0.0.3 10.0.0.2  F Payload type=ITU-T G.711 PCMU, SSRC=1231417077, Seq=2648,
61.357047          LucentTe_ I Acknowledgement
61.381642 LucentTe_ Broadcast I Probe Request
61.388803 LucentTe_ Broadcast I Probe Request
61.389621 LucentTe_ LucentTe_ I Probe Response
61.390052          LucentTe_ I Acknowledgement
61.411747 NokiaWir_ Broadcast I Beacon frame
61.413381 LucentTe_ Broadcast I Probe Request
61.434916 LucentTe_ Broadcast I Probe Request
61.436573 LucentTe_ Broadcast I Beacon frame
61.456459 LucentTe_ Broadcast I Probe Request
61.538948 LucentTe_ Broadcast I Beacon frame
61.556881 10.0.0.3 10.0.0.2  F Payload type=ITU-T G.711 PCMU, SSRC=1231417077, Seq=2658,
61.557157          LucentTe_ I Acknowledgement
61.576867 10.0.0.3 10.0.0.2  F Payload type=ITU-T G.711 PCMU, SSRC=1231417077, Seq=2659,
```

Active scan

Figure 4.6: Ethereal snapshot

It is not clear why the sender loses the connection and tries to associate again with the receiving station. This temporary disconnection occurs, most of the times, periodically - as shown by the distance between packet losses represented by the circles in figure 4.5(a).

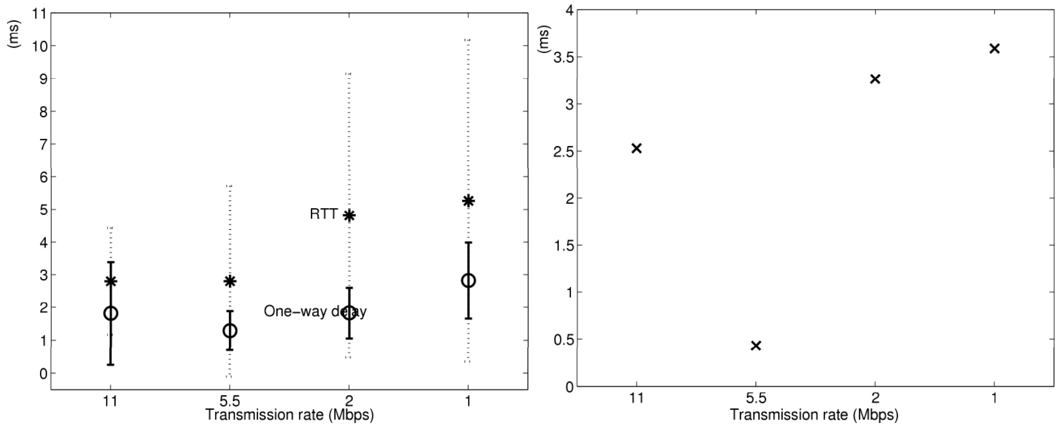
This behaviour causes a particular type of loss that we did not considered before, as we contemplated a packet loss as a consecuence of four failed frame transmissions.

It is worth noting that these periodic bursts were detected only in this set of experiments, with a Lucent Orinoco card. We used another wireless card, of the same brand, in the following experiments and we did not observe this behaviour, which may indicate some problem with that specific card. It could also be that in this particular scenario the conditions that trigger an active scanning were met.

4.2.4 Delay

The histograms of MAC frame transmissions cannot provide a quantitative estimate of the delay, as it happens with loss. However, we had additional information that allowed us to measure the delay added by the MAC (re)transmissions, i.e. the timestamps of the frames received at the sniffer (described in section 3.4.2.2).

Figure 4.7(a) shows the contribution of the MAC layer to the mouth-to-ear delay for each of the bitrates. We will consider this delay is the network one-way delay, as explained in 3.4.2.2. We also show the network round-trip times (RTT) for comparison. The oblique lines show the evolution of the mean delay (μ), while the vertical lines show the standard deviation (σ), covering the range $[\mu - \sigma, \mu + \sigma]$.



(a) Delay (one-way and RTT)

(b) Jitter

Figure 4.7: Delay and jitter at different transmission bitrates

The difference between the standard deviation of both delay and RTT plots may be explained by the considerable difference in the sampling frequency of both parameters. While the RTT values are taken at a rate of one value per second, the one-way delay values are computed for every RTP packet, i.e. 50 delay values per second, thus providing more accurate results. As expected the one-way delay is roughly half of the RTT.

It can be inferred from figure 4.7 that 5.5 Mbps had the best performance, with respect to delay. This result is consistent with the MAC frame results shown in figure 4.3.

At 11 Mbps each frame take less time to transmit than at 5.5 Mbps. However, since at 11 Mbps many of the frames had to be retransmitted, the average delay ends up being higher than at 5.5 Mbps.

We have also calculated the delay added by each of the transmissions for this particular experiment. We present in figure 4.8 the cumulative[3] delay that

[3] Cumulative: the delay shown for the n-th transmission includes the delay of all the

every subsequent transmission adds to the whole mouth-to-ear delay, in the conditions of this experiment (i.e. no background traffic) for each bitrate. The y-axis represents time (delay) in milliseconds and the x-axis shows the n-th transmission for each RTP packet.

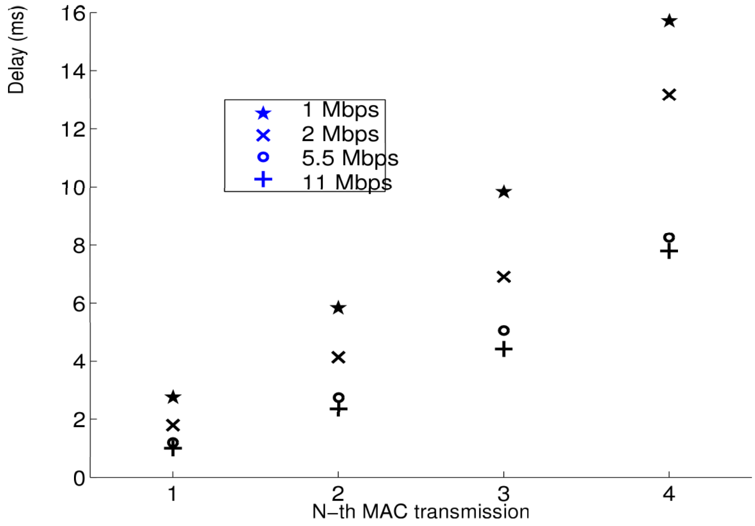


Figure 4.8: Cumulative delay of the n-th transmission

From this plot a few conclusions can be drawn. First, the tendency is as expected: The lower the bitrate, the higher the delay. Additionally, the more transmissions a packet needs to be delivered, the higher the delay. Second, a single transmission at 5.5 or 2 Mbps causes less delay than two transmissions at 11 Mbps, which would make it preferable to select 5.5 Mbps if we substantially decrease the frame error rate, as happens in this scenario.

Given the low delay obtained at 1 Mbps, we could even select it as the bitrate for voice communication as it offers superior robustness against fading and low signal to noise ratio, whilst still keeping the delay figure low. Moreover, since a voice packet only occupies the wireless channel at most 50 times per second and only for a single short frame, it would not cause a significative decrease in the overall throughput of the network.

Nevertheless, since for this particular pair of locations the frame error rates of 5.5 Mbps and 1 Mbps were very similar, 5.5 Mbps remains the best option.

4.2.5 Jitter

Figure 4.7(b) shows the jitter for each of the 802.11b transmission rates, which is quantitatively quite low for all the bitrates. The difference between the jitter found at 11 Mbps and at 5 Mbps can be explained, again, by examining the histograms. At 11 Mbps there is a large variance in the number of transmissions needed for a packet, this is the main cause of jitter. However, at 5.5 Mbps the sender is able to transmit the packets to the receiver more than 95% of the time using a single transmission. This, together with the lack of competition for

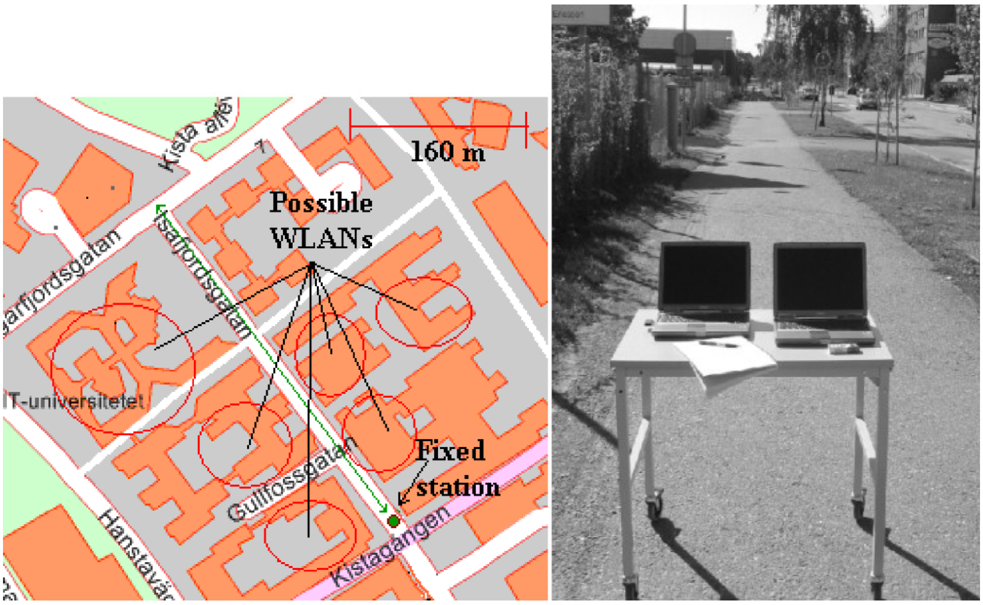
previous frame transmissions.

the channel, results in the separation between consecutive packets being rather constant and hence the jitter is low.

The values of jitter for 2 and 1 Mbps show the jitter increasing as the bitrate decreases. Since the magnitude of jitter is directly related to delay (see section 2.2.3), hence jitter is also higher at the lowest bitrates.

4.3 Distance

4.3.1 Set-up



(a)

(b)

Figure 4.9: Location and positioning of the stations along Isafjordsgatan, Kista

We performed this experiment in an (almost) open space environment to see the effect of distance on voice quality while trying to avoid any interference.

Several distances were selected between 80 m and 400 m. We decided to use intervals of 16 m, but soon we realised that we would not have enough battery supply to cover all the points, thus we chose only eight different locations between 80 and 400 m. We used, as in the previous experiment, a sender and a receiver communicating in ad hoc mode.

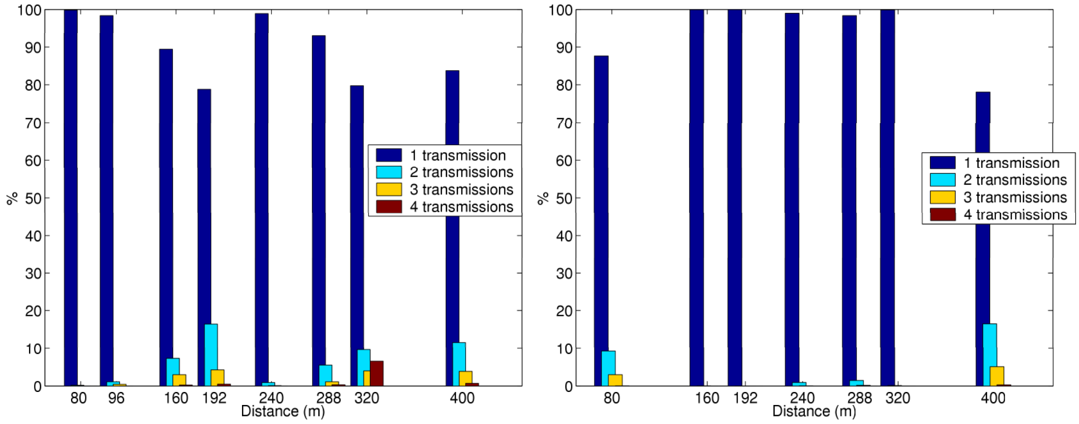
The transmission rate was set to *automatic*; in this configuration, the driver dynamically selects the transmission rate that best suits the actual link conditions. We looked for information about how the bitrate selection is performed, but we did not find anything relevant, neither in the Orinoco official website nor in the linux Orinoco driver website.

The location chosen was Isafjordsgatan, in Kista. This is a long straight street shown as in the map of figure 4.9.

The stations were placed on two carts and kept in line of sight throughout the measurements. One station, the sender, was fixed at a given position while the receiver was moved from one location to the next.

4.3.2 Transmissions histogram

Figure 4.10(a) shows the histograms of MAC transmissions versus distance. We can see the general tendency of the number of successful first transmissions decreasing with the distance. Thus, the number of retransmissions increases with the distance. However, this trend is not constant or consistent. There was something in the middle, around 192 m, that caused a disturbance that breaks such a trend. The same pattern is observed around the 320 m distance.



(a) First experiment

(b) Second experiment

Figure 4.10: Histogram of MAC transmissions against distance

The reason for the bad link quality at certain points could be some access point transmitting in the vicinity of the receiver in a frequency channel close to the one we used. It may be also possible that the significant increase in the link quality at 240 m was due to a change in the transmission rate, but we did not record the transmission rate used.

We looked at the 802.11 traffic captured by the sniffer placed close to the receiver, at 192 m, and we observed some beacon frames sent by an unknown AP. This confirmed the existence of some wireless network in the vicinity of the receiver, which probably interfered with our voice traffic at the distance of 160-192 m. Hence, the frame loss rate was higher in this region. Some spurious traffic was also found at 320 m.

When we made this experiment we did not have an appropriate wireless device to capture the information about transmission rates, which might provide information to explain the frame loss rate. Thus, we decided to repeat the experiments with an appropriate device to capture the bitrate. The new results are displayed in the figure 4.10(b).

After analysing the results we observed a substantial difference between the histograms[4], which made it difficult to extrapolate the results of bitrate obtained

[4] The experiment at 96 m could not be performed the second time due to battery restrictions.

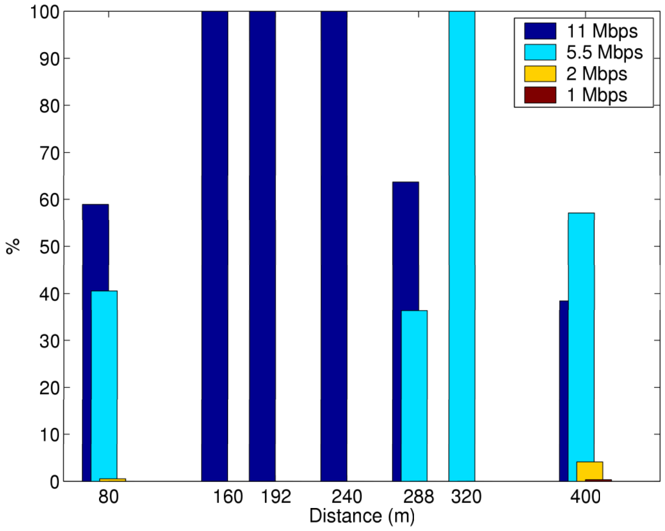


Figure 4.11: Bitrate histogram against distance

in the second experiment to the first one. Additionally, the transmission rate selection was not as straightforward as we expected, as shown in figure 4.11.

We can observe that at 80 and 288 meters the sender chose 5.5 Mbps as the bitrate for 40% of the frames, whilst at 160, 192, and 240 m the transmission was done completely at 11Mbps.

The existence of noise coming from different locations could be an explanation for such bitrate selection, as the bitrate selection mechanism is usually sensitive to the signal to noise ratio or the packet error rate. Thus, by observing the plot, we could infer that around 80 meters there was a source of noise that interfered with the communication. That noise did not affect the following experiments at 160, 192, and 240 m, as the sender transmitted all the frames at 11Mbps.

These results make it difficult to conclude what effect distance has on voice quality or, more precisely, on frame loss rate, as there were noise sources where we expected to have little interference. In the case of measurements at 80 m it is clear that noise is the cause of the higher frame loss rate (and not distance), but we could not affirm whether the change to 5.5 Mbps at 288 m was due to the effect of distance or to some other noise source. At 288 m we did not see any frames from nearby APs, but as stated in [28], the power needed to interfere with the reception of a frame is lower than the power needed to successfully transmit a frame. In other words, there could have been some interference source near the 288 m point which was not detected by capturing the 802.11 frames.

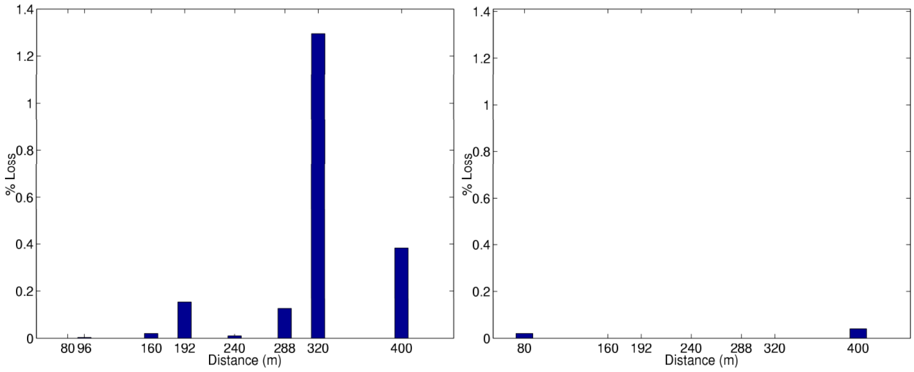
The change from all the frames being transmitted at 5.5 Mbps (at 320 m) to the mixture of bitrates found at 400 m, that includes a 40% of 11 Mbps, is also strange. Again, we could not determine whether it was the effect of distance or the existence of interference what made the sender chose those bitrates. It is likely that both effects overlapped, thus making it difficult to separate the effects of both distance and noise.

Table 4.1 shows the expected range in open space, given in the datasheet of the Orinoco cards. This range seems fairly conservative according to the bitrate values that we obtained (shown in figure 4.11).

Bitrate (Mbps)	11	5.5	2	1
Range in open space (meters)	160	270	400	550

Table 4.1: Expected range of the different bitrates in open space

4.3.3 Loss



(a) First experiment

(b) Second experiment

Figure 4.12: Probability of loss versus distance

As in the previous experiments, packet loss is related to the number of instances of four transmissions ocurred per RTP packet. While the percent of loss obtained is low for most of the codecs, some of the losses occur in bursts, which could have a severe impact on voice quality. A study about the loss burstiness for the distance experiments can be found in the sister project [12].

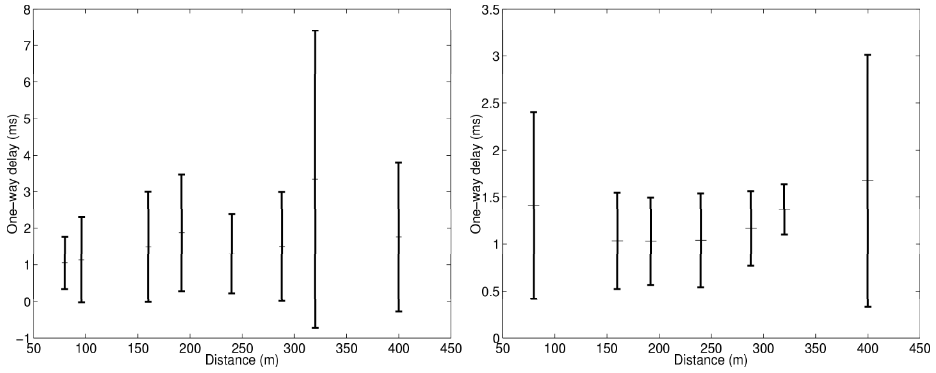
The signal-to-noise ratio (SNR), when sufficiently low, is the major factor that causes frame loss. A study of the signal and noise values would provide relevant information about the noise sources that we found in our experiments. However, such study was out of the scope of this thesis.

4.3.4 Delay and jitter

From figure 4.13 we can observe the plots of delay versus distance. As in the bitrate experiments, delay is mostly related to the number of retransmissions and the selected bitrate. This was described in section 4.2.4.

It is significant that even under the worst conditions of link quality delay does not pose a significant problem for voice quality, as the delay caused by poor SNR is practically never higher than 8 ms.

Jitter, not shown in this case, is similar in behaviour to delay, being no higher than 4 ms, regardless of the distance.



(a) First experiment

(b) Second experiment

Figure 4.13: Delay against distance

4.4 Competing traffic in ad hoc mode

4.4.1 Set-up

In this experiment we measured the effect of competing traffic on voice quality in an ad hoc wireless network. For that purpose we placed sender, receiver, and four additional laptops in the same room, all of them sharing the same wireless channel. This placement guaranteed line of sight between all the stations while minimising the effect of low signal strength owing to distance. To load the link we used two different types of traffic: TCP and UDP.

TCP, whose greedy character makes it take as much bandwidth as it can, has most of the time a packet ready for transmission, except at some instants when TCP backs off because of its congestion control mechanism and during the beginning of the TCP transmission. The TCP acknowlegment packets sent back from the destination also contribute to load on the network. UDP, on the other hand, can continously send packets at the maximum available rate.

The flows were sent between pairs of stations according to the following schema (1→2 means unidirectional flow from node 1 to node 2):

- 1 node transmitting TCP/UDP: 1→2.
- 2 nodes transmitting TCP/UDP: 1→2, 2 →1.
- 3 nodes transmitting TCP/UDP: 1→2, 2→1, 3→4.
- 4 nodes transmitting TCP/UDP: 1→2, 2→1, 3→4, 4→3.

The payload size was selected to create an IP packet of 1500 bytes, which is the maximum packet size that the 802.11b MAC layer can transmit without fragmentation. TCP traffic was started some time before the voice calls to avoid its slow start phase.

4.4.2 Effect of collisions on MAC layer captures

Collisions are problematic when performing MAC analysis. Even though the CSMA/CA medium access mechanism is designed to minimise collisions, they

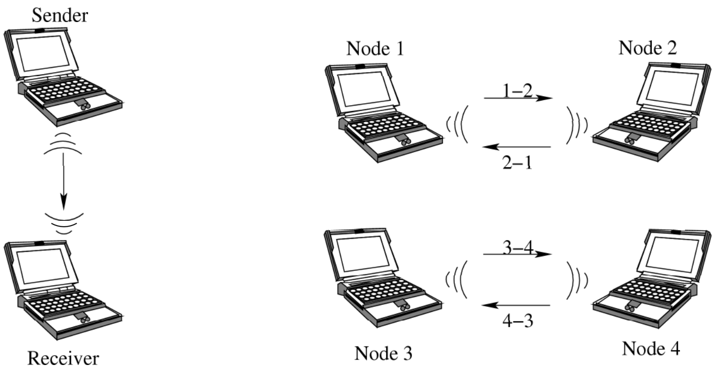


Figure 4.14: Ad hoc mode, competing traffic set-up

can occur and will lead to an incorrect calculation of the number of transmissions and hence the contribution of each MAC frame to the delay.

The main cause is that failed transmissions owing to collisions do not always appear in the traces, because the frame headers are often damaged as a result of a collision. Thus, when a retransmitted frame appears in the traces it is not possible to identify its order since there is no way to ensure whether a retransmitted frame was the first, the second, or the third retransmission, as there could be a previous retransmission that was not captured because of a collision.

The reason for not being able to identify the frame order is that MAC headers do not contain such information, with the exception of the first transmission.

The uncertainity introduced by the collisions handicaps the calculation of the MAC transmissions histogram and, as stated before, the one-way delay. Consequently, this analysis is not feasible when collisions are frequent, as the error introduced increases with the collision frequency.

An additional issue caused us to completely discard the MAC frame analysis for the experiments with competing traffic. Some frames that collided were actually captured and decoded by the sniffers, since the collision occurred *after* the header transmission. These headers can be considered as fortunate and did not suffer any damage, but others were not so lucky and some of the bits of the header were damaged.

We observed that damaged headers contained incorrect information about MAC, IP, or the even the RTP header. Although careful filtering would discard most of these frames as useless, it would not detect headers that were changed exactly in the RTP sequence number. Since our MAC analysis is entirely based on the timestamps associated with RTP sequence numbers, a change in the sequence number would lead to erroneous results in the delay or MAC transmissions figures. This problem would be avoided if the sniffers performed a checksum validation in order to discard corrupted frames.

We were unaware of these problems until we performed the MAC analysis of these particular experiments, as we did not see them reported in the literature, particularly in [17]. Nevertheless, the results obtained in non-monitor mode, such as with the Sphone and Ethereal traces are still useful as collisions do not affect them. We will use such traces, rather than MAC layer information to analyse the effect of high link load on voice quality.

4.4.2.1 Estimation of collision frequency

Despite the problems mentioned above, it is possible to extract useful information from the 802.11 traces when there are collisions. For instance, we can roughly estimate the collision frequency.

The results shown in table 4.2 reveal that without competing traffic, the voice station delivered practically all the frames on the first attempt. However, when one additional station sends TCP traffic the percentage of retransmissions rises up to 11.5 %. Hence, assuming that the majority of the retransmissions were caused by a collision (as there are practically no retransmissions without background traffic), we can determine that 11.5% is a lower bound on the collision frequency in this experiment. The real frequency is probably higher, since our traces do not show all the frames that collided for the reason mentioned above.

	% successful first transmissions	% retransmissions
No background traffic	99.3	0.7
One TCP flow	88.5	11.5

Table 4.2:

It is interesting that the fraction of retransmitted frames, without background traffic, is similar to the fraction of missed frames by the sniffers, shown in table 3.1. This indicates a consistent frame loss rate for the wireless devices that we used, even with the best link conditions. This loss rate is approximately one frame out of 150.

4.4.3 Loss

Figure 4.15 shows the packet loss that the voice calls suffered when facing an increasing number of nodes transmitting TCP or UDP traffic. Table 4.3 shows the aggregate throughput achieved by both TCP and UDP, which is an indication of the network load.

These results show that even four nodes did not cause a severe degradation to the voice stream, from a packet loss point of view. It is surprising that TCP traffic led to higher loss than UDP, considering that TCP should occupy the link slighty less than UDP, owing to the TCP's congestion control mechanism. An explanation could be the extra channel competition that the TCP acknowledgements coming from the sink stations add to the network.

Number of nodes	TCP aggregated throughput	UDP aggregated throughput
1	5.1 Mbps	5.9 Mbps
2	4.9 Mbps	6.1 Mbps
3	4.9 Mbps	6.5 Mbps
4	5.5 Mbps	6.6 Mbps

Table 4.3: TCP/UDP average throughput, ad hoc mode

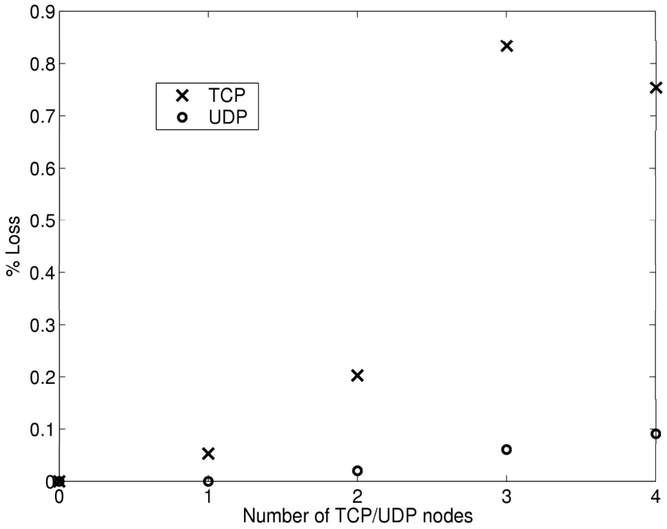


Figure 4.15: Packet loss vs. extra traffic sources, in ad hoc mode

The loss obtained for three TCP flows does not seem to follow the pattern followed by the other values, but we did not detect anything unusual in the traces. Again, the TCP acknowledgements might be the root of this high values.

In this scenario, the percentage of loss is directly related to the ability of the IEEE 802.11b DCF mode to avoid frame collisions. Four consecutive collisions must occur to cause the voice stream to lose a packet. Because of the MAC's DCF backoff mechanism, consecutive collisions become less likely. Therefore, the packet loss pattern is not bursty in nature in this particular scenario when there is good link quality.

4.4.4 Delay and jitter

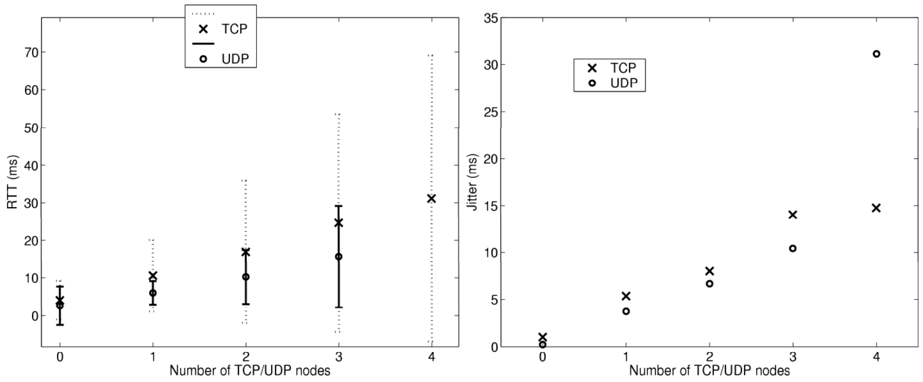
Each transmission attempt leads to different delays depending on the background traffic, because a station that wants to transmit has to wait for the channel to be idle and then contend for it once it becomes free. The average throughput given by table 4.3 is an indicator of how much the link is loaded. However, the load (or more precisely, the competition) seen by the sender is higher with several competing stations than with only one station, even with the same aggregate throughput.

Assuming that the 802.11b MAC provide long-term fairness to all the stations, as stated in [1], every node would access the channel after waiting for $(n-1)/2$ frames from other stations, with n being the number of stations within the same cell[5]. This formula is valid if all the stations most of the time have a frame ready for transmission. Thus, as we increase the number of competing nodes we should see the delay increasing as well.

Figure 4.16 shows the evolution of round trip times (RTT) versus number of competing TCP/UDP flows. The mean and standard deviation are the metrics shown. If we look at the mean values, represented by the 'X' and '$\star$' symbols,

[5] However, the voice stream only has a packet every 20 ms.

we observe that TCP caused higher delay than UDP, which is consistent with the loss results shown above.



(a) (b)

Figure 4.16: RTT and jitter versus number of competing stations, ad hoc mode

We also observe a nearly linear increase in the RTT with the number of nodes, which matches the linearity of the $(n - 1)/2$ formula except for the unusually high RTT obtained when four nodes are transmitting UDP traffic. For this particular experiment we obtained values around 900 ms, which does not seem normal according to the expectation from the other experiments. We checked the values given by 'ping', and the results were similar. However, we were not able to determine what caused these high delay values.

The standard deviation, represented by the vertical lines, also shows an increasing tendency. This can be explained by the MAC backoff mechanism: When the number of competitors increase, then collisions become more likely, which in turn causes an increase in the number of retransmissions. Then, after a collision the backoff contention window doubles its size, and the delay variation increases as well.

Looking at the quantitative values of the plots, we can state that under the conditions the experiment was conducted, the delay introduced was low, except for the anomalous values commented on above. Assuming that the one-way delay is approximately half of the RTT, we can affirm that most of the packets suffered a one-way delay of less than 40 ms.

Regarding jitter, we observe a similar trend to that of the delay (figure 4.16(b)). Again, the difference between the delay of two consecutive packets is the cause of jitter.

4.5 Competing traffic in infrastructure mode

In this experiment we wanted to measure the impact of two nodes communicating via an access point on voice quality. Additionally, we wanted to observe the evolution of the voice quality when facing the channel competition of several nodes transmitting TCP flows. We will show a comparison with the values obtained in ad hoc mode for TCP traffic.

4.5.1 Set-up

As in the previous experiment, we placed all the stations in the same room. In this case, however, all the stations including the voice nodes communicated through an access point, also placed in the room. Figure 4.17 shows this set-up.

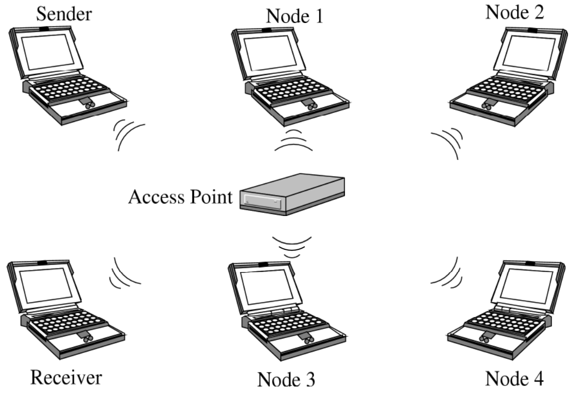


Figure 4.17: Infrastructure mode, competing traffic

4.5.2 Loss

Figure 4.18 shows the loss percentages of both managed and ad hoc modes. Assuming that the communication between the sender and the AP behaves similarly to an ad hoc communication between two laptops, the access point should lose roughly the same percentage of packets as two laptops in ad hoc mode. Then, the receiver would see the loss due to the first hop (sender to AP) plus the additional loss between the AP and the receiver.

In the case of one and two competing nodes the behaviour is as expected. In both cases the loss percentages in managed mode are higher than in ad hoc mode.

However, we observed that for the cases of three and four stations, the loss is smaller than in ad hoc mode. As this result is somewhat surprising, additional experiments would be necessary to confirm whether wireless communication via an AP leads to higher loss than communicating in ad hoc mode or not.

According to the quantitive values obtained we can state that, in both ad hoc and managed modes, voice communication does not severely degrade even with four nodes loading the network with TCP traffic, from the loss point of view. Loss of 1% is tolerated by most of today's codecs.

4.5.3 Delay and jitter

Competing traffic has a substantial impact on delay when an AP is used. Since there is a central point that receives and sends all the traffic, the network becomes saturated earlier than in ad hoc mode.

In managed mode, every packet has to be transmitted twice, first to the AP and second to the receiver. Such a redundancy has a major impact on the overall throughput, as shown in table 4.4.

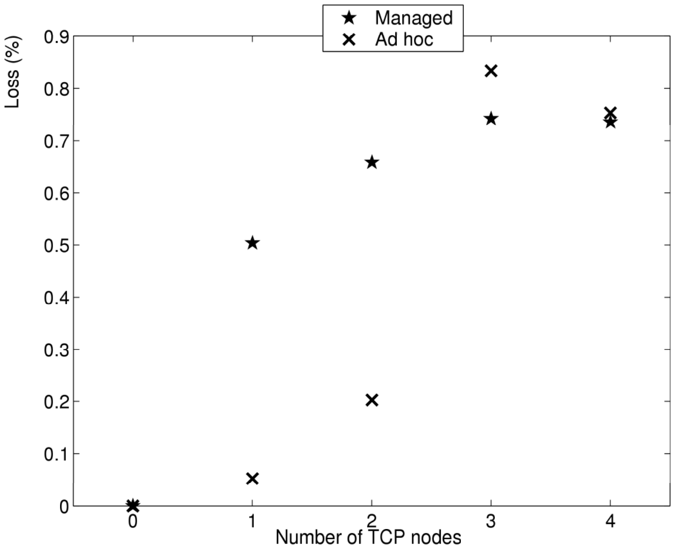


Figure 4.18: Packet loss against number of extra stations, ad hoc and managed mode

Number of nodes	TCP throughput (ad hoc)	TCP throughput (managed)
1	5.1 Mbps	2.4 Mbps
2	4.9 Mbps	2.3 Mbps
3	4.9 Mbps	2.2 Mbps
4	5.5 Mbps	2.4 Mbps

Table 4.4: TCP aggregated throughput, ad hoc and managed mode

In both hops the packet finds competition for the channel, since the AP has exactly the same priority as the wireless stations in the 802.11b DCF mode. However, the second hop is quite different from the first. The voice packets find a queue in the AP with the packets that the AP has to forward to the final stations, i.e. -all- the packets that the AP receives from the network. Therefore, the AP acts as a bottleneck for the voice packets (and for the other traffic as well), and the wait for accessing the channel increases, since the voice packets also have to wait in this queue. This queue would lead to a lower delay provided that the AP had higher priority to gain access to the medium.

The queueing at the AP, when the load in the network is high, has a critical impact on delay, as shown in figure 4.19(a). RTT values of managed and ad hoc modes are shown for comparison.

This experimental scenario is unlikely to be found in a real network. However, the measured values have application at the end point of a voice call. If the destination of the call is in a wireless network with several users making use of peer to peer services, the voice flow will find high uplink and downlink traffic, as in the investigated scenario.

The values obtained show that the end point of a voice communication can suffer delays higher than recommended for good quality, i.e. 150-200 ms, when facing competition by more than two nodes sending and receiving TCP traffic.

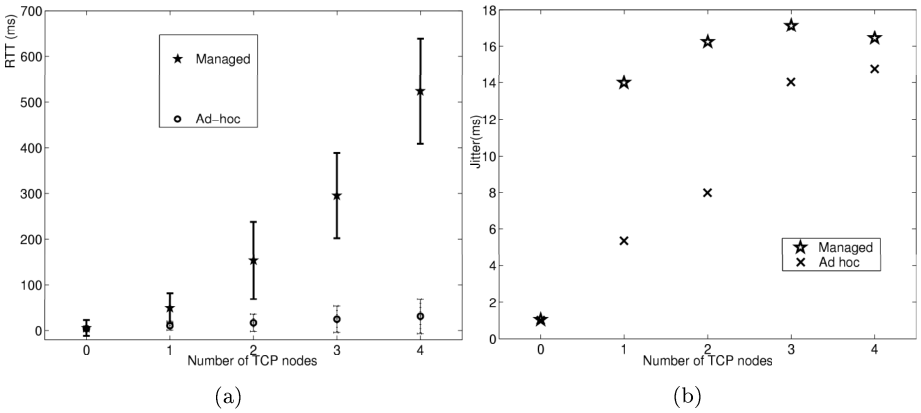


Figure 4.19: RTT against number of extra stations, ad hoc and managed mode

In this case, the downlink or second hop delay is the delay shown for the managed mode minus the delay obtained in ad hoc mode, which represents the first hop delay.

4.6 Hidden node. Effectiveness of the RTS/CTS mechanism

4.6.1 Set-up

In this experiment we wanted to verify whether the RTS/CTS handshake mechanism is effective against the hidden node problem. For that purpose we placed the sender and receiver stations out of the transmission range of two additional stations, so that each pair of stations was hidden from each other. Regular office walls were the obstacles preventing the radio signals from reaching the hidden stations. In the middle we placed an access point, which was in range of all the wireless stations. Figure 4.20 shows this set-up.

The traffic flows were the following: Nodes 1 and 2 transmitted TCP flows to each other, whilst the sender station transmitted voice streams to the receiver station. All of them used the AP to communicate with their respective destinations.

There are several questions that we tried to answer by conducting this experiment:

- Does the RTS/CTS mechanism reduce the performance of VoIP when there are no hidden nodes?
- Do the hidden nodes pose a real problem? If so, how much is the performance degradation?
- Does the RTS/CTS mechanism improve the performance of VoIP when there are hidden nodes, and the sender is the only one that uses RTS/CTS?
- Does the RTS/CTS mechanism improve the performance of VoIP when there are hidden nodes, and all the nodes use RTS/CTS?

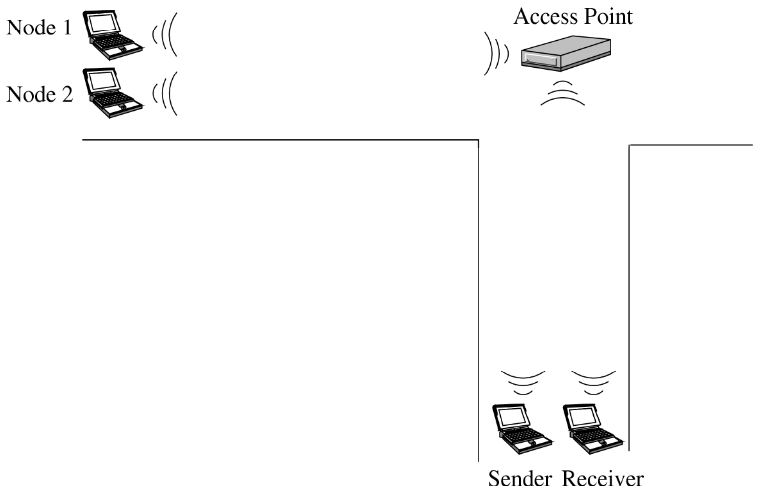


Figure 4.20: RTS/CTS experiment's set-up

In order to answer these questions, we conducted six different experiments with the following characteristics:

A No background traffic, RTS disabled in sender station.

B No background traffic, RTS enabled in sender station.

C Two stations transmitting TCP flows, RTS disabled in all the stations.

D Two stations transmitting TCP flows, RTS enabled in all the stations.

E Two stations transmitting TCP flows, RTS enabled in voice sender station only.

F Two stations transmitting TCP flows, RTS disabled, stations not hidden to each other.

4.6.2 Loss

Figure 4.21 shows the results obtained regarding loss. We will begin by considering the experiments without background traffic, i.e. A and B. The plot shows an increase up to 3% of losses when the RTS mechanism is used. However, after looking at the loss pattern we observed that the losses occurred in equally spaced bursts. This may indicate that this degradation was not due to the RTS mechanism itself, but rather to some odd behaviour of the AP. Note that these losses are not related to the bursts previously described.

Now we will focus on the experiments with background traffic. If we look at what happened after enabling the TCP flows in nodes hidden to the sender and receiver stations, we observe that the hidden nodes caused a 25% loss to the voice stream (C case), whilst the loss percentage was only 0.3% when the stations were not hidden (F). In order to reduce the loss, we enabled the RTS mechanism in all the stations, and the loss percentage effectively reduced to 2% (D).

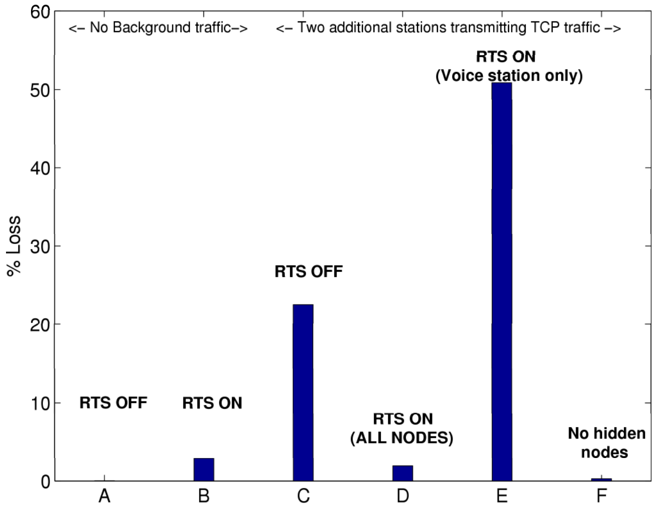


Figure 4.21: Loss, RTS/CTS mechanism

We also wanted to verify whether enabling RTS in the voice sender station only would also help to minimise the loss percentage. However, the loss percentage, instead of reducing, increased up to 50% (E). This shows that the RTS mechanism is effective against the collisions caused by hidden nodes, but only if all the stations have it enabled. In fact, to enable it in only one station causes even worse degradation than leaving it disabled, at least for this particular set-up.

The following table shows how the TCP aggregated throughput varied in the situations described above.

Experiment	A	B	C	D	E	F
Throughput (Mbps)	-	-	1.1	1.1	1.56	2.1

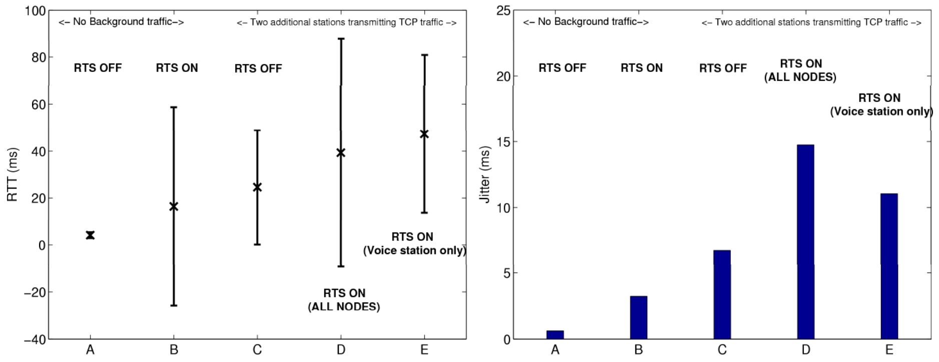
According to these values, there was no decrease in the TCP throughput after enabling RTS in all the nodes. However, the throughput increased when the voice station was the only one that enabled RTS. It is interesting that the use of the RTS mechanism benefited the performance of other stations whilst the station that used it worsened its own performance.

The highest throughput was achieved when all the nodes were in range, as the collision frequency became lower.

4.6.3 Delay and jitter

Regarding delay and jitter, figure 4.22 shows the performance of both. The overhead introduced by the RTS mechanism can be significant, although the delay values obtained when using RTS were still well below the 150 ms limit (approximately 50 ms considering a bad case scenario).

The values obtained in the F scenario were very high, around 1.5 seconds for the RTT, which is the reason why they are not shown. The reason for this high delay could not be determined, but it must be related to the high values obtained in the ad hoc competing traffic experiment.



(a) RTT

(b) Jitter

Figure 4.22: RTT and jitter, RTS/CTS mechanism

4.7 Odd cases. Occasional bursts

After looking at all the single calls to verify that the experiments were conducted properly, we observed that in some calls there were occasional bursts of losses. Such bursts were probably caused by signal fading or loss of connection (or association) between the wireless nodes.

In most of the cases such bursts were not representative. We decided to discard these calls for the computation of the traces, as the bursts would dramatically change the loss percentage without indicating a representative value for the majority of the calls.

For instance, in the bitrate measurements there was a large burst in one of the calls of the 1 Mbps set. That call was the only one that showed a burst of loss, whilst the rest of calls had zero loss. However, to include that call in our computation would have raised that loss percentage to almost a 4%, which does not represent the typical performance of 1 Mbps in that experiment.

Even though we have not incorporated these calls in our results, we must state that such large bursts can occur, regardless of the experiment type. The impact of such bursts is a severe degradation of the voice quality during the burst. However, such bursts occur rather seldom (for instance, in the bitrate measurements we discarded 3 out of 40 calls, i.e. 7.5%). Figure 4.23 shows one call that belongs to the competing traffic experiments, where a large burst of losses happened (approximately 250 packets, or 5 seconds).

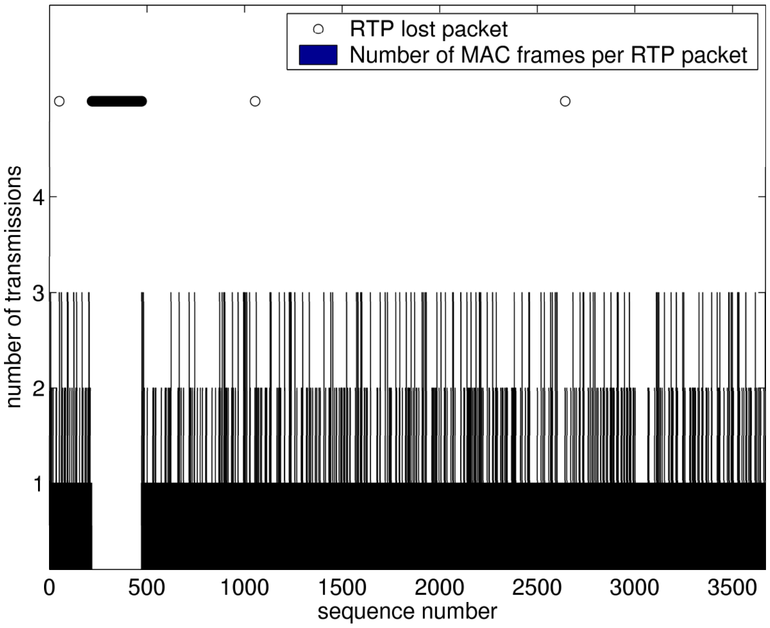


Figure 4.23: Occasional burst of losses

Chapter 5

Conclusions

5.1 General conclusions

In this thesis we have conducted measurements of the performance of the 802.11b protocol in real scenarios and considered its appropriateness for voice communication. We had no previous experience with the required techniques to conduct such experiments and thus we had to face many problems that were, in most of the cases, completely unexpected. Real experiments, compared to simulations, pose significant challenges and require a careful methodical execution, since the smallest slip can invalidate large amounts of measurements. Furthermore, several trial and error attempts must be performed before extensive data is collected. It can also happen that after days or even weeks of hard work one discovers that there was something missing and therefore all the work must be repeated, which is a *very* unpleasant experience.

However, this analysis reveals many issues that are impossible to capture using modeling or simulation and thus the information gathered via an experimental approach is tremendously valuable. It also provides values with a flavour of being 'real', in contrast to a simulation approach. From a masters student point of view the adquired knowledge is priceless, especially when the design of the measurements has to be done from scratch.

Several scenarios have been tested, and a significant amount of data collected. It was difficult to keep the focus on the main goals of the thesis since every subsequent look at the data revealed more issues that were not observed before, which dangerously diversified the potential analyses. Although this is itself a positive point, it can lead to the loss of the 'big picture' and to try to pursue every single discovered detail, as sometimes there is a inclination to not leave anything out.

Moving onto more specific details of the thesis, we can affirm that the chosen methodology was quite successful. The measurements of two different sets of parameters (MAC layer and application layer data) proved to be useful to gain an insight from both the 'top-down' and the 'bottom-up' approaches to the problem. It even led to two different, although very related, master theses.

The use of passive sniffers provided the information necessary to study the MAC protocol behaviour, although it was not well-suited for the specific case of a highly loaded network. The collisions affected not only the network performance, but also the recording of the 802.11 MAC data. Also, the selected tools (Ethereal, Sphone, Nttcp, and device drivers) were appropriate although we had to overcome a significant number of bugs with versions of drivers that were in a very early stage of development, such as the ipw2100 linux driver for the Centrino devices.

The selected scenarios covered well many of the potential situations that a user of a wireless device can find. Distance, obstacles, and competing traffic are elements always present in wireless communication. We also examined different configurations defined in the 802.11b standard that may help to improve the quality of voice, such as the transmission rate or the use of the RTS/CTS mechanism. In the majority of the experiments the results matched the expected results. An exception is the RTS/CTS experiment, where we observed that the use of this mechanism does **not** increase the performance, but instead it worsens it when applied to only one station, at least for the specific location and network load utilised.

In the case of the distance experiment it is clear that we did not choose the best test location. The effect of background noise originating from surrounding WLANs was mixed with the effect of distance, thus it was not possible to separate them in our analysis. Some conclusions could be drawn nevertheless, such as the trade-off between distance and transmission rate, and also the superior actual ranges compared to the nominal ones given in the devices's datasheet.

In the competing traffic experiments we observed that even four nodes heavily loading the network did not cause any severe impairment to the voice quality. An exception was the high delay observed when we used an access point.

The transmission rate experiment confirmed that the bitrate selection plays a major role in the voice quality, specifically concerning loss. As a side-effect it revealed potential problems such as the loss of connectivity that the triggering of an active scanning may cause; for instance, resulting significant bursts of losses.

Finally, the RTS/CTS experiment ascertained that hidden nodes in infrastructure networks pose a threat to wireless networking and specifically to voice communication, as they can cause loss percentages that codecs or concealment techniques will find difficulty in handling when attempting to provide an acceptable quality. Besides, this experiment showed that the RTS/CTS mechanism can be a solution but only if all the stations in the network use it.

5.2 Future work

With our measurements we tried to cover a number of the most relevant scenarios to VoIP over WLANs. However, we have left some out that are also interesting and which are strong candidates if this work is to be continued. Some of them are:

- Extensive measurements in public WLANs at different hours of the day. This would imply the recording of additional parameters such as the number of active users and the typical traffic that flows the network.

- Measurements dedicated to analysing additional problematic issues in wireless networks such as *exposed nodes*, adjacent channel interference, co-channel interference, or interference range.

- Measurements to analyse the signal absorption of different materials found in typical environments and its impact on voice quality.

- Measurements of the impact of motion. The mobility that a wireless terminal provides would not be that advantageous if motion significantly degrades the voice quality.

- Measurements with different number of retransmissions, in order to observe the trade-off between loss and delay that such tuning implies.

- We observed the effect of different parameters separately. It would be interesting to intentionally overlap the effect of several parameters in order to see whether the results obtained are the sum of the individual effects or not.

It would be also interesting to investigate additional techniques of wireless analysis. We have used sniffers to measure the MAC performance, a technique which has proved to be useful but not exempt from difficulties. In fact, for some scenarios it was not even appropriate. Other potential methods may be to extract information from either the device's firmware or the device driver.

Wireless technology is progressing rapidly and there are new standards available that greatly improve the features of the 802.11b standard, such as the a/g extensions. Additional standards will be available soon improving even further the wireless experience. Measurements of the performance of such technologies would complement the results presented in this thesis.

The number of scenarios covered is quite rich. Thus, the analysis of each of the scenarios has been deep only to a certain extent, but time restrictions have not allowed us to perform a more comprehensive examination of each of the scenarios and the collected data. Therefore further analysis can still be done with the data that we have already collected.

Bibliography

[1] Emre Koksal, Hisham Kassab, and Hari Balakrishnan. "An Analysis of Short-Term Fairness in Wireless Media Access Protocols". Extended version of short paper in Proc. ACM SIGMETRICS, June 2000. http://nms.lcs.mit.edu/papers/fair-sigmet00.ps

[2] Srikant Sharma. "Analysis of 802.11b MAC: A QoS, Fairness, and Performance Perspective". Department of Computer Science Stony Brook University, NY. January 2003. http://www.ecsl.cs.sunysb.edu/tr/wlanrpe.pdf

[3] George Xylomenos and George C. Polyzos. "TCP and UDP performance over a Wireless LAN". Proceedings of the IEEE INFOCOM 1999, pp. 439-446. http://www.mm.aueb.gr/archive/publications/gxpapers/c5.pdf

[4] Giuseppe Bianchi. "Performance Analysis of the IEEE 802.11 Distributed Coordination Function". IEEE Journal on Selected Areas in Communications, March 2000 pp. 535-547.http://www.ece.utexas.edu/~jandrews/ee381k/EE381KTA/802.11_throughput.pdf

[5] Theo Pagtzis, Peter Kirstein, and Steve Hailes. "Operational and Fairness Issues with Connection-less Traffic Over IEEE802.11b". Proceedings of IEEE International Conference on Communications (ICC), Helsinki, Finland, June 2001. http://www.cs.ucl.ac.uk/staff/t.pagtzis/papers/papers/ICC01opIssues802.11b.pdf

[6] Martin Heusse, Paul Starzetz, Franck Rousseau, Gilles Berger-Sabbatel, and Andrezj Duda. "Scheduling time-Sensitive traffic on 802.11 wireless LANs". QoFIS October 2003, Stockholm, Sweden. http://drakkar.imag.fr/IMG/pdf/qofis.pdf

[7] Andreas Köpsel and Adam Wolisz. "Voice transmission in an IEEE 802.11 WLAN based access network". Proceedings of the 4th ACM international workshop on Wireless mobile multimedia, Rome, 2001. http://www.tkn.tu-berlin.de/tkn/publications/papers/wowmom01.pdf

[8] T.J. Patel, V.A. Ogale, S. Back, N. Cui, R. Park. "Capacity estimation of VoIP channels on Wireless Networks". Dept of Electrical and Computer Engineering, The University of Texas at Austin. March 26th 2003. http://www.ece.utexas.edu/~wireless/EE381K11_Spring03/projects/11.3.pdf

[9] Eleftherios Dimitrou and Patrik Sörqvist. “Internet Telephony Over WLANs”. Global IP Sound White Paper, September 2003. http://www.globalipsound.com/solutions/wlan.usta_paper.pdf

[10] IEEE 802.11b IEEE Standard for Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. Institute of Electrical and Electronics Engineers, November 1999.

[11] Status of Project IEEE 802.11e. http://grouper.ieee.org/groups/802/11/Reports/tge_update.htm

[12] Victor Yuri D. Nunes. “Quality of Service (QoS) aspects of VoIP in Wireless LANs” Master Thesis, Royal Institute of Technology (KTH), IMIT. August 2004. http://www.e.kth.se/~nunes/

[13] Raphael Rom and Moshe Sidi. “Multiple Access Protocols: Performance and analysis”. Springer Verlag, New York (1990). http://www.comnet.technion.ac.il/rom/PDF/MAP.pdf

[14] Ian Marsh. “Quality aspects of audio communication”. Licentiate thesis, KTH Stockholm, June 2003. http://www.sics.se/~ianm/Lic/lic_thesis.pdf

[15] Iyad Al Khatib. “Performance Analysis of Wireless LAN Access Points”. Tekn. Lic. thesis, Department of Microelectronics and Information Technology (IMIT) KTH, Stockholm. 27 May 2003.

[16] Hector Velayos and Gunnar Karlsson. "Techniques to Reduce IEEE 802.11b MAC Layer Handover Time"- Department of Microelectronics and Information Technology (IMIT) KTH, Stockholm. April 2003.

[17] Jihwang Yeo, Suman Banerjee, and Ashok Agrawala. “Measuring Traffic on the Wireless Medium: Experience and Pitfalls”. Department of Computer Science University of Maryland. December 2002. http://www.cs.umd.edu/~jyeo/TR.pdf

[18] David Gibson. “Wireless networking with Linux and IEEE 802.11b”. Linux-Kongress 2001 Enschede, The Netherlands. http://ozlabs.org/people/dgibson/papers/wireless_networking.ps.gz

[19] L. F. Sun, G. Wade, B. M. Lines, and E. C. Ifeachor. “Impact of Packet Loss Location on Perceived Speech Quality”. Department of Communication and Electronic Engineering, University of Plymouth. Internet Telephony Workshop, New York, Apr. 2001. http://www.iptel.org/2001/pg/final_program/15.pdf

[20] Iyad Al Khatib, Rassul Ayani, and Gerald Q. Maguire Jr. “Wireless LAN Access Points Uplink and Downlink Delays: Packet Service-Time Comparison”. Proceedings of The 16th Nordic Teletraffic Seminar (NTS-16), August 2002, Espoo, Finland, pp. 253-264. http://www.it.kth.se/~ikhatib/lic/publications/NTS16/nts16-wlan-ap-udc-iyad.pdf

[21] Enrico Pelletta. “Maximum Throughput of IEEE 802.11 Access Points: Test Procedures and Measurements”.

[22] Rob Flickenger. “Building Wireless Community Networks”. O’Reilly & Associates, Inc. January 2002.

[23] Matthew Gast. “802.11 Wireless Networks. The Definitive Guide”. O’Reilly & Associates, Inc. April 2002.

[24] Christian Hoene, André Günther, and Adam Wolisz. "Measuring the Impact of Slow User Motion on Packet Loss and Delay over IEEE 802.11b Wireless Links". Technical University of Berlin. In proc. Of Workshop on Wireless Local Networks (WLN) 2003, Bonn, Germany. October 2003. http://www.tkn.tu-berlin.de/publications/papers/hoene_01.pdf

[25] Yuval Boger. “Fine-tuning Voice over Packet services”. VP Business development, RADCOM Ltd. Accessed June 2004. http://www.protocols.com/pbook/pdf/voip.pdf

[26] Pablo Brenner. “A Technical Tutorial on the IEEE 802.11 Protocol”. Technical Report, 1997. http://www.sss-mag.com/pdf/802_11tut.pdf

[27] Alex Lackpour. “Comparison of Adjacent and Co-Channel Interference for Three and Four Channel Assignment Plans in IEEE 802.11b WLAN Networks”. Oberon Wireless, Inc. April 2003.

[28] Kaixin Xu, Mario Gerla, and Sang Bae. “How effective is the IEEE 802.11 RTS/CTS handshake in ad hoc networks?”. University of California, Los Angeles. Published in IEEE Global Communications Conference (GLOBECOM02), Nov 2002.

[29] Ivaylo Haratcherev, Reginald Lagendijk, Koen Langendoen, and Henk Sips. “Hybrid Rate Control for IEEE 802.11”. ACM International Workshop on Mobility Management and Wireless Access Protocols (MobiWac), Philadelphia, PA, October 2004. http://www.isa.ewi.tudelft.nl/~koen/papers/MobiWac.pdf

[30] J. H. James, Bing Chen, and Laurie Garrison. "Implementing VoIP: A Voice Transmission Performance Progress Report" IEEE Communications Magazine, July 2004, Vol. 42, No. 7, pp. 35-41.

[31] Akira Takahashi, Hideaki Yoshino, and Nobuhiko Kitawaki. “Perceptual QoS Assessment Technologies for VoIP”. IEEE Communications Magazine, vol. 42, no. 7, July 2004.

[32] Jost Weinmiller, Hagen Woesner, Jean-Pierre Ebert, and Adam Wolisz: "Analyzing the RTS/CTS Mechanism in the DFWMAC Media Access Protocol for Wireless LAN’s", IFIP TC6 Workshop Personal Wireless Communications (Wireless Local Access), Prague, April 1995. http://citeseer.ist.psu.edu/135438.html

[33] Giuseppe Anastasi and Eleonora Borgia. “Wi-fi in Ad Hoc Mode: A Measurement Study”. Second IEEE International Conference on Pervasive Computing and Communications (PerCom’04), March 2004. http://www.dsg.cs.tcd.ie/uploads/category410/240.pdf

[34] G. Holland, N. Vaidya, and P. Bahl. "A Rate-Adaptive MAC Protocol For Wireless Networks". Mobicom 2001. http://citeseer.ist.psu.edu/holland00rateadaptive.html

[35] Javier del Prado Pavon and Sunghyun Choi. "Link adaptation strategy for IEEE 802.11 WLAN via received signal strength measurement". ICC 2003 - IEEE International Conference on Communications, vol. 26, no. 1, May 2003 pp. 1108-1113. http://mwnl.snu.ac.kr/~schoi/publication/Conferences/03-ICC-LA.pdf

[36] L. Sun and E. Ifeachor. "Perceived Speech Quality Prediction for Voice over IP-based Networks". Proceedings of IEEE International Conference on Communications (IEEE ICC'02), New York, USA, April 2002, pp.2573-2577.

[37] Akira Takahashi, Hideaki Yoshino, and Nobuhiko Kitawaki. "Perceptual QoS Assessment Technologies for VoIP". IEEE Communications Magazine, vol. 42, no. 7, July 2004. http://dl.comsoc.org/cocoon/comsoc/servlets/GetPublication?id=3566244

[38] Abdelbasset Trad, Qiang Ni, and Hossam Afifi. "Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Networks". Second International Workshop on Multimedia Interactive Protocols and Systems (MIPS 2004), Grenoble, France. November 2004.

[39] A. Servetti and J.C. De Martin. "Adaptive Interactive Speech Transmission Over 802.11 Wireless LAN's". Workshop on DSP in Mobile and Vehicular Systems. Nagoya, Japan, April 2003. http://www.cercom.polito.it/Publication/Pdf/175.pdf

[40] Andrzej Kochut, Arunchandar Vasan, A. Udaya Shankar, and Ashok K. Agrawala. "Sniffing out the correct Physical Layer Capture model in 802.11b". IEEE International Conference on Network Protocols (ICNP), Berlin, Germany. October 2004.http://www.cs.umd.edu/Library/TRs/CS-TR-4583/CS-TR-4583.pdf

[41] http://www.catb.org/~esr/faqs/smart-questions.html. Accessed March 2004.

[42] Intel PRO Wireless 2100 Driver for Linux (ipw2100) driver. http://ipw2100.sourceforge.net/. Accessed June 2004.

[43] Linux Hostap driver. http://hostap.epitest.fi/. Accessed June 2004.

[44] Linux Orinoco driver. http://www.nongnu.org/orinoco/. Accessed May 2004.

[45] Ethereal, a network protocol analyser. http://www.ethereal.com/. Accessed June 2004.

[46] Matlab, the language of technical computing. http://www.mathworks.com/. Accessed June 2004.

[47] Wireless tools. http://www.hpl.hp.com/personal/Jean_Tourrilhes/Linux/Tools.html. Accessed May 2004.

[48] Nttcp network traffic generator. http://www.leo.org/~elmar/nttcp/. Accessed July 2004.

[49] Fedora project. http://fedora.redhat.com/. Accessed February 2004.

Appendix - Acronyms

Acronym	Full term
ACK	Acknowledgement
AP	Access Point
BPSK	Binary Phase Shift Keying
CCK	Complementary Code Keying
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CTS	Clear To Send
DCF	Distributed Coordination Function
DIFS	DCF Interframe Spacing
DSSS	Direct Sequence Spread Spectrum
EIFS	Extended Interframe Spacing
IEEE	Institute of Electrical and Electronics Engineers
FHSS	Frequency Hopping Spread Spectrum
IP	Internet Protocol
IR	Infrared
ISM	Industrial, Scientific, and Medical
KTH	Royal Institute of Technology
GSM	Global System for Mobile Communications
LAN	Local Area Network
LLC	Logical Link Control
MAC	Medium Access Control
MOS	Mean Opinion Score
NAV	Network Allocation Vector
OSI	Open System Interconnection
PCM	Pulse Code Modulation
PESQ	Perceptual Evaluation of Speech Quality
PCF	Point Coordination Function
PIFS	PCF Inteframe Spacing

QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RTCP	RTP Control Protocol
RTP	Real-time Transmission Protocol
RTT	Round-Trip Time
RTS	Request To Send
SICS	Swedish Institute of Computer Science
SIFS	Short Interframe Spacing
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications System
VoIP	Voice Over IP
WEP	Wired Equivalent Privacy
WLAN	Wireless LAN

IMIT/LCN 2004-09

www.kth.se