UNIVERSIDADE TÉCNICA DE LISBOA

INSTITUTO SUPERIOR TÉCNICO

Algorithms for Quality of Service in a WiFi Network

Daniel Martins Sebastião

(*Licenciado*)

Dissertation submitted for obtaining the degree of

Master in Electrical and Computer Engineering

Supervisor: Doctor Luís Manuel de Jesus Sousa Correia

Jury

President: Doctor Luís Manuel de Jesus Sousa Correia

Members: Doctor Atílio Manuel da Silva Gameiro

Doctor Rui Manuel Rodrigues Rocha

April 2008

To the ones I love

Acknowledgements

This thesis is the final and visible result of more than a year of hard work. It was only possible to complete this work as desired, with the support of some people to whom I would like to thank. These are the people that kept me smiling even when I had no reasons to. Thank you all: without you, I would have not been able to complete this work.

Starting by Prof. Luís M. Correia, I would like to thank him as the supervisor of this work, and, above all, the friend, that gave a great and decisive contribution for the development of the work. He was always available when I needed to talk to him, when I needed some input or advice.

To *Instituto de Telecomunicações*, I would like to thank for the support that it always provided me, starting by the working space and all conditions for finalising this work. Also, all colleagues and friends that I have there, I give special thanks: Carla Oliveira, Gonçalo Carpinteiro, Lúcio Ferreira, Diana Ladeira and Martijn Kuipers.

Next, I would like to thank my parents and brother that have always accompanied me for all my live. They have provided me all the strength, the trust, the education and expertise to complete this task.

To Sara Mendes and Susana Marques, and all my friends that have accompanied me for some years now, and that have always supported me to go one step ahead.

To all of those who are not in this list, but that have helped me to finish this work, even if simply by just keeping me smiling.

Thank you all. Without your help, I would not be writing this.

Daniel Martins Sebastião

Abstract

The goal of this work is to study and develop Quality of Service (QoS) algorithms for Wireless Local Area Networks (WLANs). A standard that aims at providing QoS in WLANs already exists, the IEEE 802.11e one, which was studied in detail. Its performance was analysed, compared to the legacy ones, and new algorithms were developed in order to improve it. All this was done in multi-service, multi-profile scenarios, with a varying number of users, in order to emulate the behaviour of real wireless networks. The OPNET Modeler simulation tool was used.

Network performance was evaluated with and without QoS mechanisms. Performance increases in all scenarios when using QoS mechanisms, *e.g.*, up to 7 times less delay depending on the scenario and on the number of users.

New algorithms in the QoS mechanisms were tested, *i.e.*, changing system parameters to see if a better performance can be achieved. It is seen that default parameters offer a good performance for all profiles even varying the number of users. But it is concluded that it is possible to tune up the network, according to its location and typical usage profile, hence, improving network performance and users QoS.

Keywords

WLANs, IEEE 802.11e, QoS, Algorithms, Simulation

Resumo

O objectivo deste trabalho é o de estudar e de desenvolver algoritmos de Qualidade de Serviço (QoS) para redes locais sem fios (WLANs). Já existe uma norma com este objectivo, nomeadamente a norma IEEE 802.11e e que foi estudada em detalhe. O seu desempenho foi analisado e comparado com as normas anteriores, desenvolvendo-se novos algoritmos para o tentar melhorar. Tudo isto em ambientes multi-serviço, com variações nos perfis de utilização e no número de utilizadores, de forma a tentar emular, uma verdadeira rede sem fios. Foi utilizada a ferramenta de simulação OPNET Modeler.

O desempenho da rede com e sem os mecanismos de QoS foi estudado. O desempenho melhora em todos os cenários, observando-se, por exemplo, melhorias até 7 vezes no atraso, dependendo do cenário e do número de utilizadores.

Novos algoritmos de QoS foram testados, modificando os parâmetros de sistema numa tentativa de melhorar o desempenho. Verifica-se que os parâmetros por omissão são um bom compromisso para todos os perfis, mesmo variando o número de utilizadores. Mas conclui-se que é possível configurar a rede à medida da sua localização e perfil típico de utilização, melhorando desta forma o seu desempenho e a QoS dos utilizadores.

Palavras-chave

WLANs, IEEE 802.11e, QoS, Algoritmos, Simulação

Table of Contents

List of Figures

List of Tables

List of Abbreviations

3GPP	3rd Generation Partnership Project
AC	Access Category
ACK	Acknowledgment
AIFS	Arbitration IFS
AIFSN	AIFS Number
AP	Access Point
APSD	Automatic Power Save Delivery
BAR	Block ACK Request
BS	Base Scenario
BSS	Basic Service Set
CBR	Constant Bit Rate
CDF	Cumulative Density Function
CF	Contention Free
CFB	Contention Free Burst
CTS	Clear To Send
CW	Contention Window
DCF	Distributed Coordination Function
DIFS	Distributed Coordination IFS
DL	Downlink
DLS	Direct-Link Setup
DS	Distribution System
EDCA	Enhanced DCF Channel Access
ERP	Extended Rate Physical
ESS	Extended Service Set
FTP	File Transfer Protocol
FSM	Finite State Machines
HC	Hybrid Coordinator
HCF	Hybrid Coordination Function
HCCA	HCF Controlled Channel Access

HEC	HTTP + Email Centric
HEM	HTTP + Email Maximum
HVSC	HTTP + Video Streaming Centric
HVSM	HTTP + Video Streaming Maximum
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineers
IFS	Interframe Space
IP	Internet Protocol
IR	Infrared
LAN	Local Area Network
LLC	Logical Link Control
MAC	Medium Access Control
MAN	Metropolitan Area Network
MMS	Multimedia Messaging Service
nQAP	Non-QoS Access Point
nQSTA	Non-QoS Station
NRT	Non Real-Time
NRTC	Non Real-Time Centric
NRTM	Non Real-Time Maximum
OFDM	Orthogonal Frequency Division Multiplexing
OSI	Open Systems Interconnection
PC	Point Coordinator
PCF	Point Coordination Function
PDF	Probability Density Function
PHY	Physical Layer
PIFS	Point Coordination Function IFS
PLCP	Physical Layer Convergence Procedure
PMD	Physical Medium Dependent
QAP	QoS Access Point
QBSS	QoS Basic Service Set
QoS	Quality of Service
QSTA	QoS Station
RF	Radiofrequency
RRM	Radio Resource Management

RTC	Real-Time Centric
RTM	Real-Time Maximum
RTP	Real-Time Transport Protocol
RTS	Request To Send
SI	Service Interval
SIFS	Short IFS
SMS	Short Messaging Service
STA	Station
TC	Traffic Classes
TSPEC	Traffic Specification
TXOP	Transmission Opportunity
UL	Uplink
UP	User Priority
VBR	Variable Bit Rate
VoIP	Voice over IP
WLAN	Wireless Local Area Network
WWW	World Wide Web

List of Symbols

CW_{min}	Contention Window minimum value
CW_{max}	Contention Window maximum value
N	Number of simulations
ε_N	Relative error at simulation N
μ_N	Cumulative average of N simulations
$\overline{\mu}$	Global average

Chapter 1

Introduction

This chapter gives a short overview of the work, presenting the motivation, current state of the art, and novelty of the work. At the end of the chapter, the work structure is provided.

Communications are essential to Man, as there has always been a need to communicate among people. The provision of effective, reliable and always available communication facilities is, nowadays, essential. A significant effort to improve communications systems is always required, as users always demand higher throughput, more quality, or the possibility of communicating anywhere, anytime.

Mobile communications appeared to respond to this need of having the ability to communicate wherever and whenever needed. The big explosion in mobile cellular communications systems started with the so-called second generation mobile communications systems, the digital generation. The appearance of these new digital systems led the availability of new services, like Short Messaging Service (SMS), or the ability of using email in mobile phones. As a consequence an explosion on the number of mobile communication users worldwide occurred, with penetration rates of above 100 % in many countries by now [ANAC07].

This “mobile explosion” led to the development of new wireless standards, to enable the deployment of new technologies and services. The development of Wireless Local Area Networks (WLANs) followed this trend, trying to enable network access in locations not covered by wired networks. WLANs have a large set of benefits over the common and already existing wired networks, such as: mobility, easy deployment, expandability, and cost. The IEEE 802.11 standard, [IEEE03], is the most common WLAN standard worldwide, with its multiple amendments.

The basic 802.11 standard, defined all the basic principles required to the deployment of these kinds of WLANs. It allows transmission rates of up to 2 Mbps using 2.4 GHz radiofrequency (RF) or infrared (IR). Following the deployment of this standard, several amendments were introduced, trying to compensate for all the drawbacks it had, compared to wired networks: less secure, less reliable, or less throughput. These amendments included the definition of the new physical layer specifications to enable higher throughputs (802.11a/b/g/n), to increase security (802.11i), or to enable its usage in countries with different spectrum regulation (802.11d/h/j), Table 1.1.

Nowadays, multimedia applications are the centre of the Internet. Voice over Internet Protocol (VoIP) and video-conference services, like Skype [SKYP07], are becoming widespread due to their low cost and Video streaming usage is very high these days: it is now possible to watch live television, using streaming services, and online video streaming sites, like YouTube [YOUT07], enjoy enormous success, with millions of videos freely available.

Table 1.1. 802.11 standard and amendments basic description (adapted from [IEEE08]).

Standard	Specification
802.11	The original standard: 1 and 2 Mbps, 2.4 GHz RF and IR standard (1997)
802.11a	New physical layer (PHY), 54 Mbps, 5 GHz (1999)
802.11b	New PHY, 11 Mbps, 2.4 GHz (1999)
802.11c	Bridge operation procedures (2001)
802.11d	International roaming extensions (2001)
802.11e	New Media Access Control (MAC) layer, to enable Quality of Service (QoS) Support (2005)
802.11f	Inter-Access Point (AP) protocol (2003, withdrawn 2006)
802.11g	New PHY, 54 Mbps, 2.4 GHz, compatible with 802.11b (2003)
802.11h	Spectrum managed 802.11a (for Europe) (2004)
802.11i	Enhanced security (2004)
802.11j	Extensions for Japan (2004)
802.11k	Radio resource measurement enhancements (under development)
802.11l	(reserved)
802.11m	Maintenance of the basic standard
802.11n	Enhancements for higher throughput (under development)
802.11o	(reserved)
802.11p	Wireless access for the vehicular environment (under development)
802.11q	(reserved)
802.11r	Fast roaming (under development)
802.11s	Mesh networking (under development)
802.11t	Recommended practice for evaluation of 802.11 wireless performance (under development)
802.11u	Interworking with external networks (under development)
802.11v	Wireless network management (under development)
802.11w	Protected management frames (under development)
802.11x	(reserved)
802.11y	3650-3700 MHz operation in the U.S.A. (under development)
802.11z	Extensions to Direct-Link Setup (DLS) (under development)

These new multimedia services impose new challenges to networks, as they have stringent delay and throughput requirements that must be fulfilled. No one will use a VoIP or video-conferencing service if the delay is too large. For such, one needs to have some kind of QoS guarantees provided by the network. This is somewhat easy in a normal wired network, but it is hard in a wireless one, as the medium is highly variable. In WLANs, this is even more of a problem, as it uses unregulated spectrum, thus, no real QoS guarantees can be provided. A possible approach is gradation and prioritisation of services, giving more priority to real-time or near real-time services.

The 802.11e standard defines a new set of QoS enhancements for the 802.11 WLANs, with modifications to the MAC layer. These enhancements include the definition of new modified Acknowledgment (ACK) policies, a more efficient power management method, or the ability of direct connection among stations even in an infrastructure WLAN (DLS). But, the more important features are related to coordination functions.

A new coordination function was defined, namely, the Hybrid Coordination Function (HCF). Within HCF, two channel access mechanisms are defined: Enhanced Distributed Channel Access (EDCA) and HCF Controlled Channel Access (HCCA), which are in general somehow similar to the ones defined in the legacy standard (DCF and PCF). Significantly new is the definition of traffic classes, for traffic prioritisation. These classes allow traffic from different applications, thus, with different priorities to be separated, and to have different priorities while trying to access the channel. Each of these classes uses different values in a given set of system parameters that are employed, while contending for the channel access.

In order to study the behaviour of WLANs one needs to use simulation, as it would be very difficult to analyse such a network analytically, especially in a multi-user, multi-service scenario, such as occurs in real networks. Furthermore, simulation allows one to study what will be the effects of changing network parameters, applications used, or even the number of users. This allows one to have an idea of how will a given network react to a given variation that might occur in real networks.

The goal of this work is to study and analyse QoS algorithms in WLAN networks, specifically looking at the 802.11e standard. The first objective is to see how network performance changes when using the QoS mechanisms, compared to the legacy standard. Also, new algorithms were analysed, trying to see if better performance can be achieved by tuning up system parameters. All this is done in order to emulate the behaviour of a real wireless network.

The OPNET Modeler [OPNE07] commercial simulation engine is used in this work. It includes an extensive collection of libraries of both vendor specific and standard models, which allows one to simulate and study a wide range of communication networks, including WLANs. These models include all APs, Servers, and Stations (STAs) that are needed in this work. The IEEE 802.11e standard is also partially implemented, including all EDCA characteristics and all the different ACK policies. HCCA is currently not implemented in Modeler, and it is not studied in this work.

A reference scenario was defined, based on the AROMA Project specification, [Ljun06], including 6 different services, ranging from real-time to non-real time ones: VoIP, Video Conference, Video Streaming, HTTP, FTP and Email. The "business profile" proposed in AROMA is taken as a reference, with 60 % real-time users and 40 % non real-time ones, with a total of 23 users. Several scenarios were then defined, based on the reference scenario, varying the weights of the available services: ranging from 80 % to 10 % real-time users. Also, the number of users was increased, from the 23 users used as reference to 69 (200 % increase).

The first group of simulations was designed to analyse the impact of using QoS algorithms in this wide range of scenarios, varying the number of users and the application profiles. 10 different application profiles were simulated, with 7 different number of users, both with and without Qos mechanisms, summing up to 1 120 simulations (10 simulations per scenario).

In the second group of simulations, the objective was to study the impact of system parameters variation in global results, with the aim of improving performance by tuning up these parameters. In this group of simulations, only the three most representative scenarios defined earlier were analysed, again with a varying number of users. All four system parameters were varied multiple times, which makes a total of 1 440 simulations (10 simulations per scenario).

The novelty of this thesis is that it provides a complete and comprehensive analysis of QoS mechanisms in WLANs. This analysis is made for multi-service, multi-user scenarios, in order to emulate real networks, which enables one to have a complete know-how about QoS mechanisms performance in WLANs. Not only the impact of using QoS mechanisms was studied, but also, new algorithms were looked into, changing system parameters values, in order to tune up network performance, at least under some conditions.

The results obtained in this work were used as a contribution to the Deliverables of the European IST-AROMA Project, which deals with Radio Resource Management strategies in All IP Heterogeneous Mobile Radio Environment, [AROM07].

The thesis is structured as follows. In Chapter 2, an overview of WLANs is presented, together with a description and classification of applications and services. Also, the main features of the

IEEE 802.11e standard are described, together with the state of the art concerning QoS mechanisms in WLANs. Next, in Chapter 3, one analyses QoS parameters and algorithms, describing their expected influence in network performance. In addition, the OPNET Modeler simulation engine is presented, together with the simulator implementation. After that, in Chapter 4, one presents how the reference scenario was defined, together with its main characteristics needed to implement it in Modeler. All simulation scenarios are then described, based on this reference scenario, and results from the two groups of simulations are shown. Chapter 5 presents the main conclusions, and some suggestions for future work. Finally, annexes present some useful information, like the detailed description of all applications and scenarios needed to define them by using Modeler, or the definition of all performance metrics used in this work.

Chapter 2

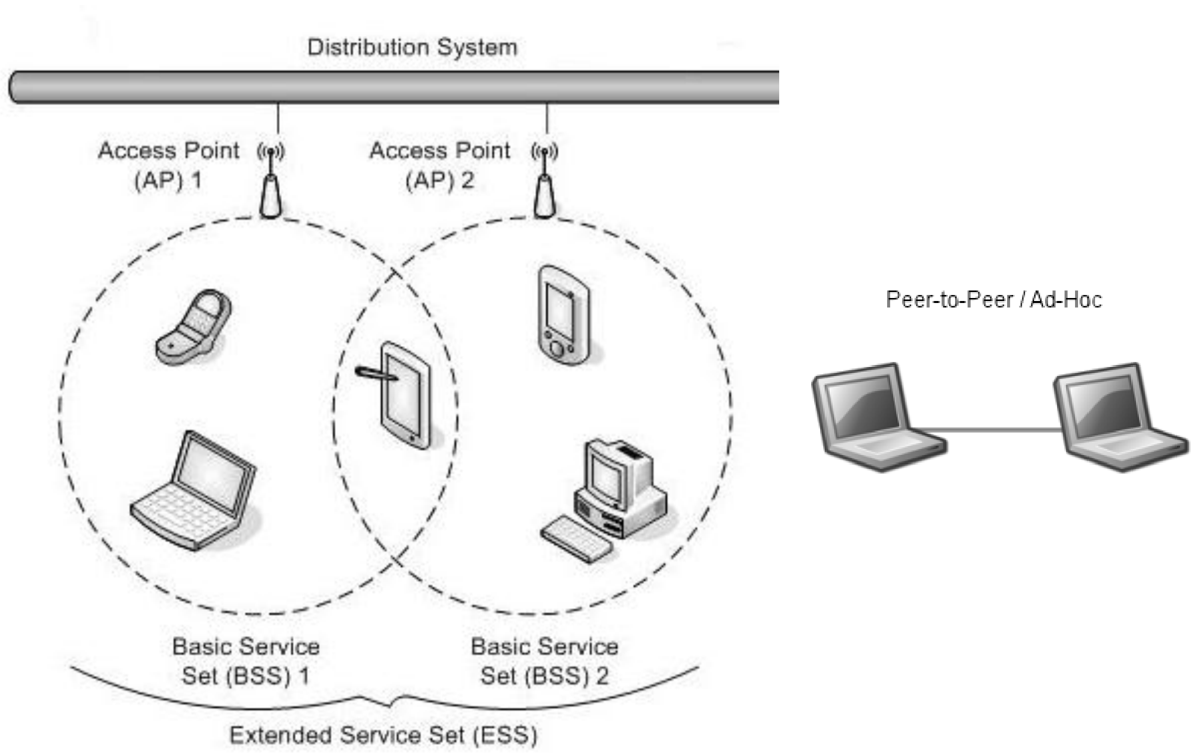
Wireless LANs

This chapter provides an overview of Wireless LANs in general, and more particularly of IEEE 802.11 networks. Fundamental aspects are addressed here, together with a brief description of services and applications. Then, the QoS part of WLANs, that is to say, the 802.11e standard, is subject to a more detailed analysis followed by the state of the art on this subject.

2.1. Wireless Local Area Networks

Wireless Local Area Networks enable the establishment of a wireless communication among computers or other similar connection devices. From a user's stand point, they work exactly as a normal wired LAN, but allowing mobility while still being connected to the network.

Two main WLAN architecture types exist, the so called ad-hoc and infrastructure networks [STAL05], Figure 2.1.



a) Infrastructure b) Ad-hoc

Figure 2.1. Types of WLANs (adapted from [WIKI06]).

The simplest networks are the ad-hoc ones, where wireless stations communicate directly with each other. Usually, this kind of networks is created just on a temporary basis, to meet some temporary communication needs, as for example an exchange of files between two computers. The other type is the infrastructure network, where all stations are connected to the network through a central point, the AP, a collection of stations connected to an AP being the Basic Service Set (BSS). No direct communication is allowed among stations in the BSS, rather, they communicate to the AP, and the AP then forwards the data to the destination one. More than one AP can be present in a given network, each with its own coverage area, the cell. APs are connected to other APs using the Distribution System (DS); usually, the definition of the DS is beyond the scope of WLANs, and some kind of wired LAN, *e.g.*, Ethernet, is used. A collection of BSSs interconnected by the DS is known as the Extended Service Set (ESS).

The IEEE (Institute of Electrical and Electronics Engineering) 802.11 standards family is currently the most widely used one for WLANs [IEEE08], being developed by Working Group 11 of the IEEE LAN/MAN Standards Committee. An original standard for a complete WLAN specification was initially designed, and after that, several other standards have been defined under the 802.11 family for, *e.g.*, different PHY specifications, enhancing security, service enhancements and extensions, or corrections to previous specifications.

In Figure 2.2, the relation between the 802.11 standard protocol layers and the OSI (Open System Interconnection) reference model is shown. The scope of the 802.11 family of standards is lower layers, including the physical layer and the data link one.

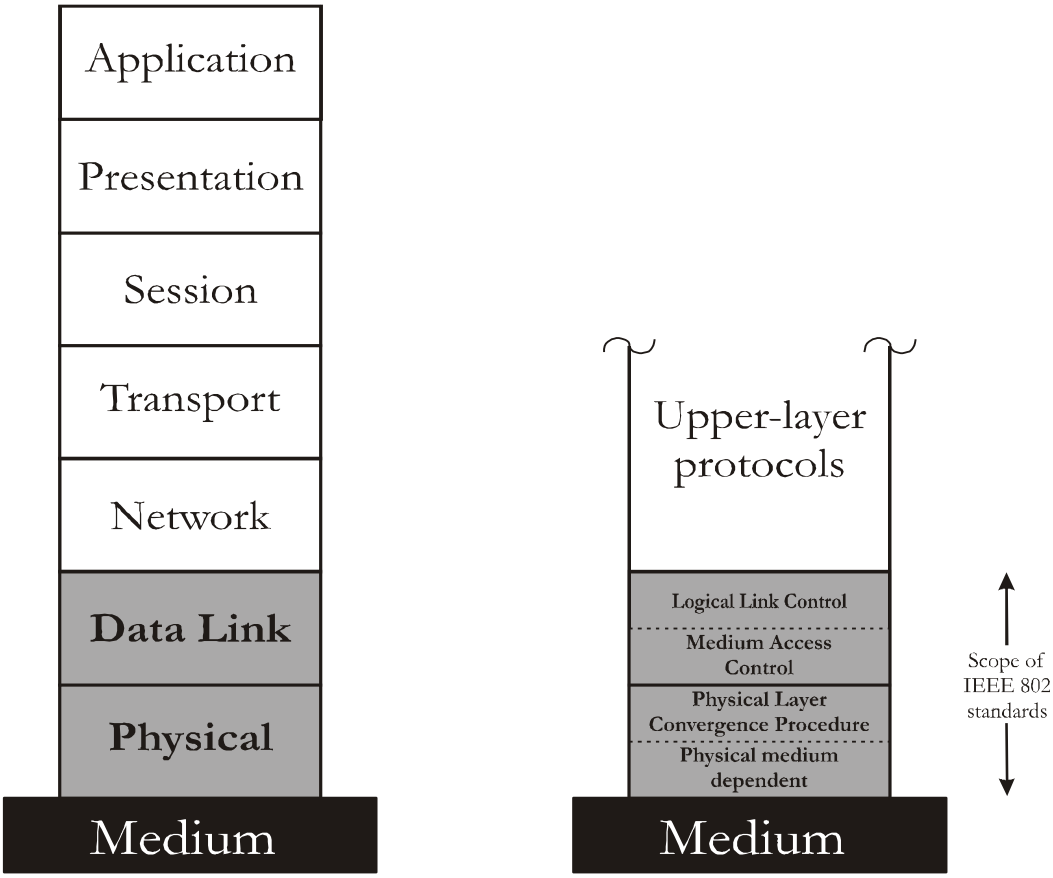
The PHY is further divided into the Physical Layer Convergence Procedure (PLCP) and the Physical Medium Dependent (PMD) sub-layers. This separation is done because the choice of the transmission medium and topology is very important in LAN design.

The 802.11 PHY has the main task of providing the upper layers with mechanisms of wireless transmission, being also responsible for assessing the state of the channel and reporting it to the MAC layer. As long as the interface between the MAC and the PHY is maintained, different PHYs can be used, which enables 802.11 WLANs to achieve higher data rates, by simply using enhanced specifications within the PHY.

There are several PHY specifications defined by the 802.11 standards body:

- 802.11, infrared, up to 2 Mbps;
- 802.11, 2.4 GHz direct sequencing, up to 2 Mbps;

- 802.11, 2.4 GHz frequency hopping, up to 2 Mbps;
- 802.11b, 2.4 GHz direct sequencing, up to 11 Mbps;
- 802.11g, 2.4 GHz extended rate physical (ERP) layer, up to 54 Mbps;
- 802.11a, 5 GHz Orthogonal Frequency Division Multiplexing (OFDM), up to 54 Mbps.



a) OSI Model b) IEEE 802.11

Figure 2.2. IEEE 802 protocol layers compared to the OSI model (adapted from [STAL05]).

Each PHY has multiple data rates defined, which allows the network to act in response to channel variations. At a given instant, the data rate can be lowered, this way allowing to obtain a larger coverage area and better resilience against errors, by using simpler modulations and different coding rates.

The layer above PHY is the data link layer, which is responsible for providing services to users, such as, providing assembly and disassembly of frames and error control. In 802.11, this layer is also further divided into two sub-layers, namely the Logical Link Control (LLC) and the MAC ones, since the traditional data link layer from the OSI model is not able to manage the access to a shared medium [STAL05].

The MAC layer is responsible, among other things, for the medium access control itself. As any wireless network, 802.11 WLANs are subject to considerable unreliability, due to noise, interference, and many other propagation effects. In order to fight this problem, a frame exchange protocol has been defined by the standard. When a station receives a data frame from another one, it must return an ACK to the source station. If the source station does not receive the ACK within a short period of time, then, it must retransmit the data frame. In order to enhance reliability, an optional four frame exchange scheme may be used, which involves sending a Request To Send (RTS) frame by the source station to the destination one, which must respond with a Clear To Send (CTS) frame. After this, the normal data exchange scheme is used, terminating with the ACK. The RTS/CTS scheme prevents all other stations from transmitting during the data exchange.

Regarding MAC in WLANs, two coordination functions may be used: a distributed access protocol, where the decision to transmit is distributed over all stations, and a central access one, where a centralised coordinator regulates the transmission in the medium.

The 802.11 standard defines a mandatory distributed access protocol, where the decision to transmit is distributed over all stations by using a carrier-sense mechanism, as it is used in Ethernet. But, wireless stations cannot sense if the channel is idle while transmitting, thus, a different access mechanism is needed. 802.11 based WLANs use Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA).

In the CSMA/CA mechanism, a given station must sense the channel when it wants to transmit, and if the channel is busy, it must wait until it becomes idle; after this, it must wait a random amount of time before it attempts to transmit again. This random amount of time, called backoff time, is used to prevent that multiple stations start to transmit at the same time when the channel becomes idle. When the station starts to transmit, it indicates how long it will take, in order to prevent other stations from transmitting at the same time. If the communication is successful, then, the receiving station must acknowledge it.

The CSMA/CA mechanism is implemented through the Distributed Coordination Function (DCF), in which the amount of waiting time is called Interframe Space (IFS).

The backoff time is a random number, between 0 and a value known as Contention Window (CW). This value varies from a minimum one, CW_{min}, being doubled after each unsuccessful transmission, until it reaches its maximum, CW_{max}. This is known as binary exponential backoff, and it is used to provide a means of handling heavy loads. Repeated failed attempts will result in longer and longer backoff times, which will help smoothing the load. Both CW_{min} and CW_{max} have fixed default values, depending on the PHY layer used.

There are three different types of IFS, which are used in different situations, Figure 2.3:

- SIFS (Short IFS): It is the shortest one, being used for all immediate responses, *e.g.*, for acknowledgements;

- PIFS (Point Coordination Function IFS): It is used when there is a centralised controller of the medium;

- DIFS (Distributed Coordination Function IFS): It is the longest one, used by the stations contending for the channel.

The relationships among IFS specifications are defined as time gaps on the medium, and the associated attributes are fixed per PHY [IEEE03]. The PIFS and DIFS timings are derived from the SIFS one, by adding a given number of timeslots. Both the SIFS and the timeslot values are fixed per PHY, being calculated as the sum of various timings, *e.g.*, MAC processing delay or the propagation time.

The whole process of transmission using DCF is shown in Figure 2.3.

There is another medium access technique defined in the standard, which is not mandatory and not very much used, called Point Coordination Function (PCF). PCF is implemented on top of DCF, and it consists of polling the stations by a centralised Point Coordinator (PC). The PC makes use of PIFS, shorter than the DIFS used by stations, capturing the control of the medium, locking the asynchronous traffic while issuing polls.

When PCF is implemented, time must be divided between PCF and DCF, or else PCF would lock all the asynchronous traffic. These timings are not defined by the standard.

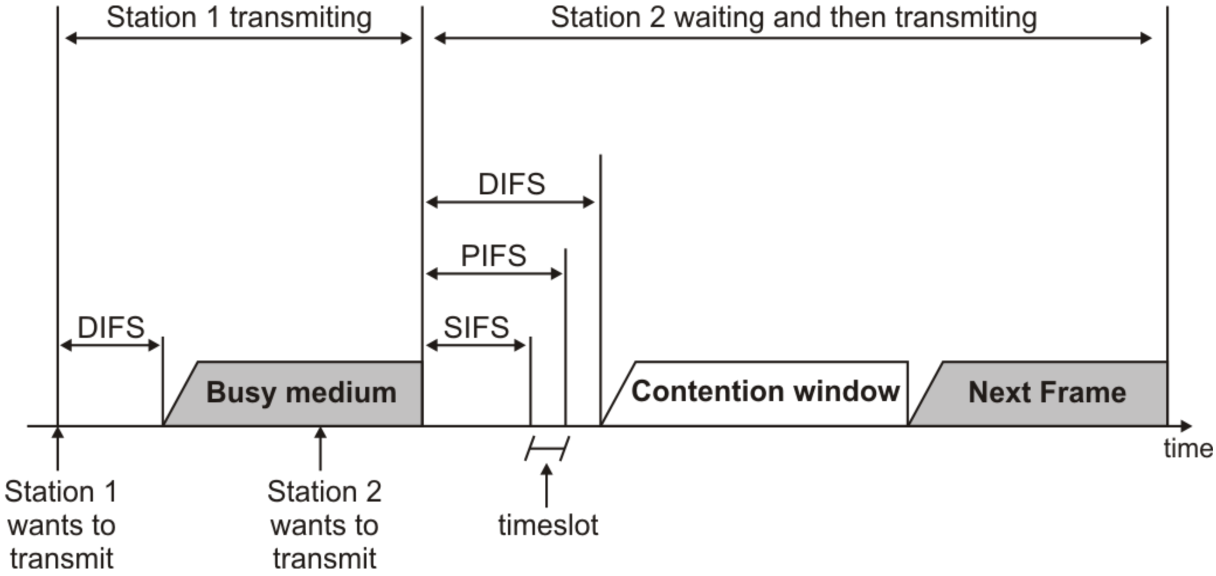


Figure 2.3. DCF coordination function (adapted from [STAL05]).

2.2. Services and Applications

When a user is connected to the Internet via a WLAN, he/she can use a wide variety of different applications with different requirements, in terms of throughput or delay, just to name a couple. WLANs must be very flexible, in order to support this wide range of applications.

Services must be classified in order to group similar applications, so that QoS guarantees to these applications can be provided. This way, services can be defined by the capacity set that is provided by the network, enabling users to use applications. On the other hand, applications are tasks that allow the connection between two or more terminals, [FeSC02]. But the classification of different services by itself cannot give any guarantees to these services. It is also necessary to mark the different types of traffic present in the network, so that it can be differentiated and the service classification can be used.

Regarding services, there are several proposed classifications, like, for instance, the one from the 3rd Generation Partnership Project (3GPP), [3GPP06a] and [3GPP06b]. As proposed by 3GPP, services can be classified into groups, according to the QoS that they can assure to the final user, Table 2.1. This classification is mainly directed to cellular networks. The characteristics of a WLAN are somehow different, but even so, the 3GPP classification can be used as a starting point to analyse services differentiation.

Table 2.1. Service classes main characteristics (extracted from [3GPP06a] and [3GPP06b]).

Class	Conversational	Streaming	Interactive	Background
Real-time	Yes	Yes	No	No
Symmetric	Yes	No	No	No
Assured throughput	Yes	Yes	No	No
Delay	Minimum and fixed	Minimum and variable	Moderate and variable	Large and variable
Example	Voice	Video-clip	World Wide Web (WWW)	SMS

In the Conversational class, services have near-symmetric two-way traffic, in Downlink (DL) as in Uplink (UL). In this class of services, by imposition of human perception, the end-to-end delay must be small (as an example, it is considered that a conversation only seems fluent when delays are lower than 400 ms, [3GPP06a]). Voice, video telephony, and some games, which, for their own characteristics need very low delays, are the best examples for this category.

In the Streaming class, information is transported in a continuous stream, allowing its processing by the end user (*e.g.*, visualisation) before the reception of the entire file is finished. Traffic is very asymmetric, DL being the most significant one. The usage of buffers allows this class of services to be more delay tolerant, compared to the Conversational one. Examples of some applications for this class are audio streaming and video on demand.

In the Interactive class, the user can ask for different kinds of information from a certain remote server. Services in this class are generally more tolerant to delays, and generate an asymmetric traffic. However, on the other hand, there is no tolerance to errors in the received information; therefore, the error probability must be low to prevent too many retransmissions. As an example of an application in this class, one can mention the WWW browsing.

The common aspect for applications considered in the Background class relies on that the user does not have a limited time to receive the information, so, the system does not need to process the information immediately, which allows delay to be high. Therefore, applications inserted in this class only use the network for the information transmission when its resources are not being used by other applications from other service classes. Despite the delays, the information transmitted should not have errors. An application example of this class is Email.

In the 802.1D standard [IEEE04], another type of traffic differentiation is proposed, some new traffic types being defined. The types defined in the standard represent different kinds of traffic that can be present in a given network. Having a distinction between traffic types allows latency and throughput guarantees to be supported by the network. This traffic classification was proposed for wired LANs, but it is used also by the 802.11e [IEEE05] standard for QoS in WLANs.

Seven traffic types are defined in IEEE 802.1D, as follows:

- Network Control – the most important traffic that must have priority over the rest;
- Voice – very stringent regarding delay, the maximum delay being as low as possible;
- Video – with some limitations regarding the maximum delay, but not as severe as with the Voice traffic type;
- Controlled Load – having important applications subject to admission control and with controlled throughput;
- Excellent Effort – a best effort traffic with higher priority than the lower classes;
- Best Effort Traffic – the normal LAN traffic;
- Background – traffic that is allowed on the network, but that should not interfere with the traffic from any of the other classes.

In some scenarios, *e.g.*, 802.11e networks, [IEEE05], a different number of Traffic Classes (TCs) can be available on a given network, and, sometimes, some of the defined traffic types must be grouped together into a single class. This problem is also addressed in the standard, some possible solutions being proposed, depending on the number of classes available within the network. A mapping between various TCs and the traffic types defined (including a spare type) is also identified in the standard, Table 2.2. According to the standard, this spare type is to be used for scenarios where it may be advantageous to have an additional traffic type, similar to best effort, to support bandwidth sharing management for bursty data applications.

Table 2.2. Mapping between User Priority (UP) and traffic types (extracted from [IEEE04]).

User Priority/Traffic Classes	Designation	Traffic type
1	BK	Background
2	-	Spare
0 (Default)	BE	Best Effort
3	EE	Excellent Effort
4	CL	Controlled Load
5	VI	Video
6	VO	Voice
7	NC	Network Control

2.3. Quality of Service in WLANs: The 802.11e Standard

The 802.11e standard defines the MAC procedures to support LAN applications with QoS requirements, including the transport of voice, audio, and video, [IEEE05]. These latency sensitive multimedia applications require effective QoS mechanisms to ensure that their priority over common network traffic, as e-mail or web traffic, is maintained.

802.11 networks work well with time-insensitive data applications, thus, providing guarantees for QoS in these networks is a challenging task. Nevertheless, one should understand that providing QoS “guarantees” is limited by the medium itself and its characteristics, this being particularly true for unlicensed spectrum, like in 802.11 WLANs. Some other problems, like the hidden node problem, co-channel overlap, or even the fact that it is a half duplex medium, can also increase the difficulty for QoS provisioning. However, gradations of service can be made, as well as provisions for the prioritisation of traffic.

Some approaches, [Caei04], can be used to have some kind of QoS provisions, even with the legacy 802.11 standard. For instance, PCF can be used by polling the stations with scheduling/admission control, according to their priorities. Also, one could change system parameters, like the IFS value of the stations, with the purpose of changing their priorities. These approaches have some drawbacks, 802.11e being developed with the objective of overcoming them.

In 802.11e, in order to indicate that the AP or the stations support QoS, their names change to

QoS Access Point (QAP) and QoS Stations (QSTA), which together form a QoS Basic Service Set (QBSS). The standard is compatible with the legacy 802.11 one, so that QSTA can connect to non-QoS Access Points (nQAP), without using the QoS enhancements. A non-QoS station (nQSTA) can also be associated to a QBSS, if allowed by the QAP.

The main changes in the 802.11e standard are on the coordination functions, a new coordination function, called Hybrid Coordination Function, being created. The new MAC architecture is presented in Figure 2.4, where one can see that HCF, and the already existing and optional PCF, are provided through the services of DCF.

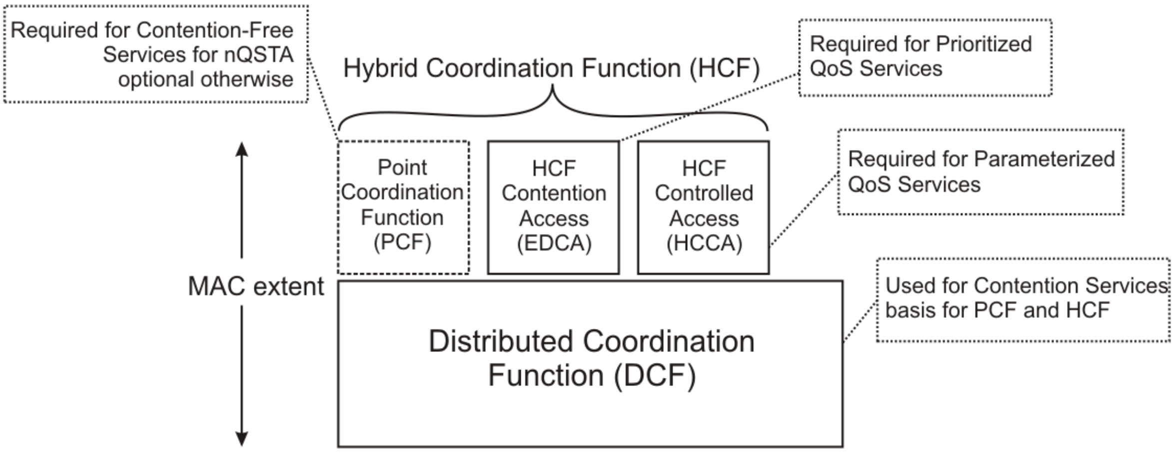


Figure 2.4. MAC layer on 802.11e (extracted from [IEEE05]).

Within HCF, two channel access methods are defined: HCF Controlled Channel Access, for contention free channel access, and Enhanced DCF Channel Access, for contention based channel access.

The EDCA channel access method replaces DCF as the contention based channel access. Within EDCA, the prioritisation of traffic is achieved by separating the one with higher priority from the other with lower priority. The traffic classification proposed in the IEEE 802.1D standard is used, with the eight different user priority values, Table 2.2. The different traffic types are then mapped onto four different Access Categories (ACs), Table 2.3.

This classification of traffic into classes is mandatory, in order for the system to support QoS, but the mechanisms of classifying and marking traffic with the appropriate QoS values are outside the scope of the 802.11e standard; it just defines the mechanism to differentiate traffic according to its priority.

Table 2.3. UP to AC mapping (adapted from [IEEE05]).

Priority	**UP (Same as 802.1D user priority)**	**AC**
Lower	1	0
↓	2	0
	0	1
	3	1
	4	2
	5	2
	6	3
Higher	7	3

For any given QSTA, for each of the ACs, an enhanced variant of the DCF is used, which contends for Transmission Opportunities (TXOPs), *i.e.*, an interval of time, defined by an initial value and a maximum duration. When a QSTA gains access to the medium, it can perform multiple frame/acknowledgement exchanges for the same AC, as long as it does not exceed the TXOP maximum duration. This multiple frame exchange within a given TXOP is usually called a contention free burst (CFB), as the transmitting station enjoys free access to the channel during the TXOP, without any contention process. This procedure differs from the legacy 802.11 standard, where stations can only transmit one frame and receive its respective ACK after gaining access to the medium.

The different ACs, within a given QSTA, contend for the channel in an independent way. The prioritisation is guaranteed by having different values, in each AC, for some parameters, like the CW and the TXOP duration limit. A different IFS, called Arbitration Interframe Space (AIFS), which is also calculated based on SIFS, is also used for each AC. Different values for AIFS allow further differentiation among different ACs. Default values for these parameters for a given QSTA are defined in the standard, Table 2.4, but these values can be changed whenever needed by the QAP, which also uses different default values for some of the parameters. This gives a lot of flexibility to network management, as it can adapt the parameters values to the usage profile.

Finally, as one can see, the AIFS value is not shown in Table 2.4, rather appearing the AIFS Number (AIFSN). The AIFSN value represents the number of timeslots that are added to SIFS to obtain AIFS.

Table 2.4 gives an idea of the impact of the different values on traffic prioritisation:

- AC(0) is for background traffic, and so CW values remain the same as for the DCF, AIFS being the larger of all the ACs, the TXOP limit being zero, which shows that only a single frame is allowed to be transmitted;

- AC(1), which has higher priority, is for best effort traffic and it is very similar to DCF, but with smaller AIFS, thus, giving it a higher priority over AC(0) traffic;

- AC(2) is used, *e.g.*, for video streaming, and has a smaller CW. This allows the traffic from this AC to have smaller backoff periods, in order to decrease the waiting time before accessing the channel; together with this, larger TXOP durations are used, allowing larger frame exchanges;

- AC(3) is used for voice or network control traffic; in this AC, the shortest CW is used, but it also has a smaller TXOP limit, because voice and network control frames are usually smaller, and do not require much time for transmission.

Table 2.4. QSTAs Access Category medium access default parameters (adapted from [IEEE05]).

AC	QSTA/QAP	min CW	max CW	AIFSN	TXOP limit (802.11b) [ms]	TXOP limit (802.11a/g) [ms]
0	QSTA/QAP	CW_{min}	CW_{max}	7	0	0
1	QSTA	CW_{min}	CW_{max}	3	0	0
	QAP		$4\times(CW_{\min}+1)-1$			
2	QSTA	$((CW_{min}+1)/2)-1$	CW_{min}	2	6.0	3.0
	QAP			1		
3	QSTA	$((CW_{min}+1)/4)-1$	$((CW_{min}+1)/2)-1$	2	3.0	1.5
	QAP			1		

It is possible that packets from two of more ACs collide internally in the same QSTA, which is known as a virtual collision. In this situation, it is up to the QSTA to resolve this issue, granting precedence to the higher AC and increasing the backoff period of the lower AC. One should also note that the QAP can use different parameters from the ones that the QSTAs use. In Figure 2.5, the EDCA reference implementation model is presented, as defined in the standard.

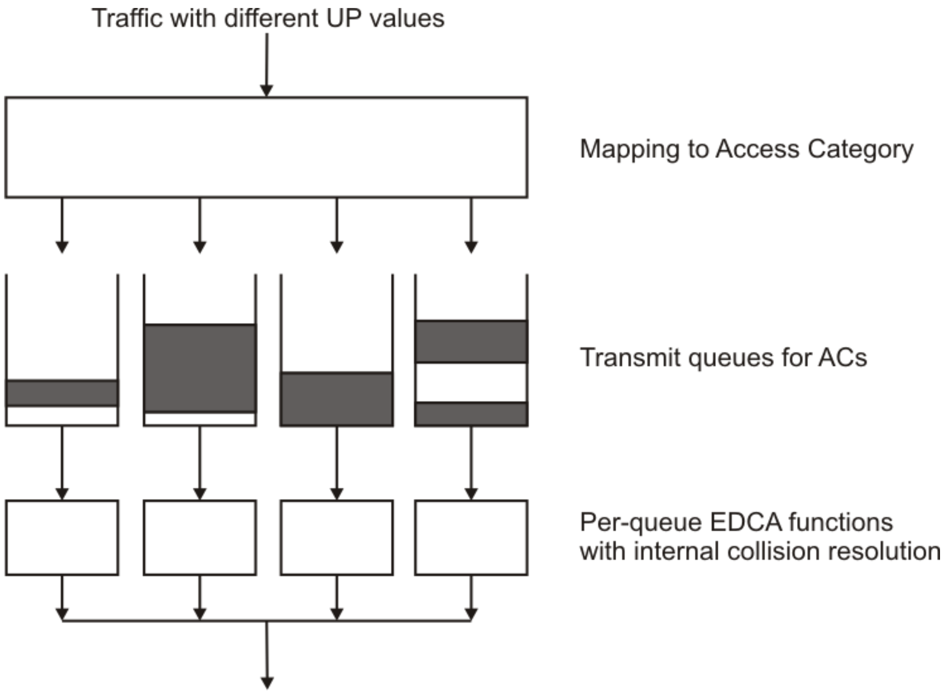


Figure 2.5. EDCA reference implementation model (extracted from [IEEE05]).

The other coordination function defined in 802.11e is HCCA, as previously mentioned, which is very similar to PCF. Within HCCA, a QoS centralised coordinator called Hybrid Coordinator (HC), which is collocated with the QAP, controls the access to the medium. The HC uses its higher priority (by waiting only a PIFS interval) to gain access to the medium and to allocate TXOPS to QSTAs, or to itself, in order to provide contention free data transfer. Unlike PCF, the HC may allocate TXOPs during the Contention Period (CP) or in any locally generated Contention Free Period (CFP), which is generated optionally by the HC.

Besides the fact that HCCA is a contention free coordination function, and EDCA is a contention based coordination function, what truly differentiates these two functions is that HCCA uses a different admission control procedure. Within EDCA, the admission control is distributed among the different QSTAs, but the decision to use it or not is taken centrally by the QAP. The QAP has also the ability to choose which of the ACs should use admission control procedures. HCCA works in a different way, requiring that a given QSTA requests the HC specific traffic parameters for each application stream, such as VoIP. With these supplied parameters, the HC evaluates if the network can accommodate this new application stream without degrading the performance of any of the existing streams. After performing this evaluation, the HC can accept, deny, or even suggest a different set of parameters to the source QSTA. If a traffic stream is accepted, then, the HC will allocate TXOPs (called polled TXOPs) to the requesting QSTA in order to fulfil its traffic needs as requested. This way, there is a high degree of confidence that there will be enough resources for all the streams to have the desired performance.

As long as the station request is fulfilled, the HC may coordinate traffic in any way it wants. For that, the HC must maintain a strict schedule of all the traffic in the network, in order to adjust its QoS policies accordingly. The HC implementation is very important for the overall performance of the network, but it is not defined in the 802.11e standard, rather being left to vendors' implementation. The standard just includes reference implementations for both the scheduler and admission control procedures, but these issues can be changed and optimised by any vendor.

The characteristics of the data flow to and from the station are known as Traffic Specification (TSPEC). A TSPEC frame is defined, which allows the client to specify some parameters, like the frame size, priority, frame, and data rate. The TSPEC frame has a set of parameters that is more extensive than it may be needed, or may be available, for any particular instance of parameterised QoS traffic. The HC may change the value of the parameters that have been left unspecified by the QSTAs to any value that it deems appropriate, including leaving them unspecified. The TSPEC data is enough for the HC to determine if the wireless medium can sustain this newly requested stream, without degrading any of the existing ones. A given QSTA must generate a unique TSPEC for each of the traffic streams it wants to transmit and receive, and for each direction of the stream, *i.e.*, bidirectional traffic belonging to the same application requires two TSPECs. The TSPEC frame format is presented in Figure 2.6.

When a given traffic stream ends, it can be removed by the QSTA or the HC, by explicitly deleting the TSPEC request or if the TSPEC timeout elapses.

Within the 802.11e standard, some other enhancements are defined. These enhancements are not directly related to the provisioning of QoS guarantees to applications, but are improvements that aim at providing better network performance or better wireless medium usage. These enhancements are:

- Block Acknowledgment (Block ACK);
- No Acknowledgment (No ACK);
- Direct-Link Setup (DLS);
- Automatic Power Save Delivery (APSD).

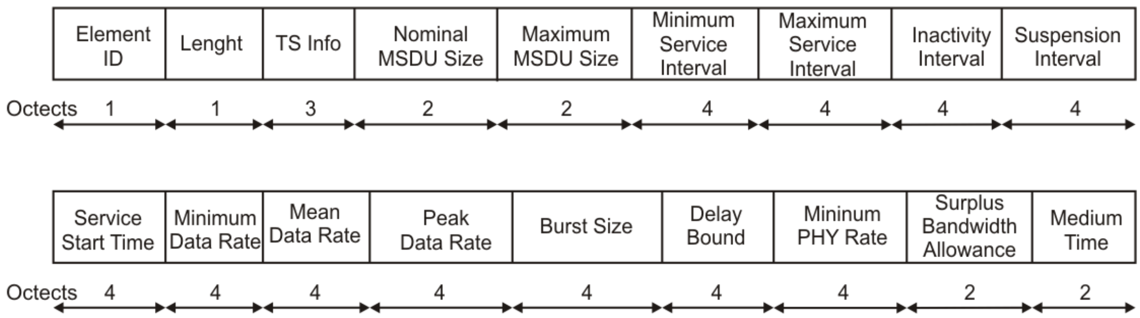


Figure 2.6. TSPEC frame format (extracted from [IEEE05]).

Block ACK allows a given station to acknowledge multiple frames by using a single ACK frame. This is used to improve performance by reducing the overhead caused by the transmission of multiple ACK frames. There are two types of Block ACK: immediate and delayed. Immediate Block ACK is suitable for high bandwidth, low latency traffic, while delayed Block ACK is suitable for applications that tolerate moderate latency, [IEEE05].

The No ACK mechanism, as the name implies, is characterised by the absence of ACK frame transmission during the exchange of data. This helps to decrease even further the overhead caused by the ACK frames, but with this mechanism the reliability of the traffic is reduced, compared to other ACK policies. Even so, this can be helpful in some specific situations, as with highly time-critical applications, its usage being determined by each QSTA.

The DLS mechanism allows direct frame transfer between two stations, within a BSS. This is not a performance enhancement, rather being a way to better use the resources of the wireless medium.

Finally, the Automatic Power Delivery Service is a more efficient power management method. APSD is useful for battery-operated devices, allowing the devices to turn off their radio for the majority of the time, still being capable of maintaining a reasonable low latency response to data sent by the QAP.

2.4. QoS Mechanisms Survey

Multimedia applications, like video, VoIP, and other real-time applications, are becoming more and more popular. These kinds of applications pose new challenges to networks, especially to WLANs, because they have more stringent requirements for delay and throughput. With the high demands and varying requirements of these applications, there is the need to support QoS in WLANs, [RaPD05].

As mentioned before, a new standard was defined by the 802.11 Working Group, IEEE 802.11e, in order to provide differentiation mechanisms at the MAC layer, enabling QoS guarantees. IEEE 802.11e defines a set of changes to channel access mechanisms, but how to apply these mechanisms is beyond the scope of the standard, reference implementations being proposed for some issues, like the scheduler of the AP, or admission control mechanisms.

There has been some work to analyse and evaluate the effects of the proposed changes in 802.11e, *e.g.*, [CPSM03], [PrSh04], [MCMK02] and [GuZh03]. In [CPSM03], the contention-based channel access mechanism (EDCA) is evaluated. Different scenarios are studied, in order to analyse the performance of both EDCA and the legacy DCF, being shown that EDCA can provide differentiated channel access among different priority traffic. In [GuZh03], the EDCA channel access mechanism is described and discussed. Simple simulations were used to evaluate its performance, allowing the comparison of both throughput and delay of the different ACs. It is shown that for light traffic conditions, the EDCA mechanism is able to differentiate the higher priority traffic from the lower priority one. In [PrSh04], a comparative analysis is made among the different channel access mechanisms (DCF, EDCA and HCCA), the effect of using different ACK policies being addressed. Different scenarios are analysed, the new channel access mechanisms achieving some improvement over DCF, *e.g.*, on throughput efficiency. It is also shown that HCCA achieves better performance in a heavy loaded network and when there is a bad link (*i.e.*, a link with lower throughput). In [MCMK02], a comprehensive overview of the most important 802.11e new features is presented. Performance of EDCA, QoS support with EDCA, QoS support with EDCA contending with DCF, and QoS guarantees with HCCA is analysed. The results show that 802.11e is an efficient mean for QoS support, even with the presence of legacy stations operating without QoS support, *i.e.*, using DCF.

IEEE 802.11e standard enhancements deal with the problem of QoS provision to applications, improvements being possible in providing QoS support to applications. Even so, some issues must still be addressed to enable a comprehensive QoS support to applications [RaPD05], Figure 2.7:

- handling time-varying network conditions;
- adapting to varying application profiles;
- managing resources.

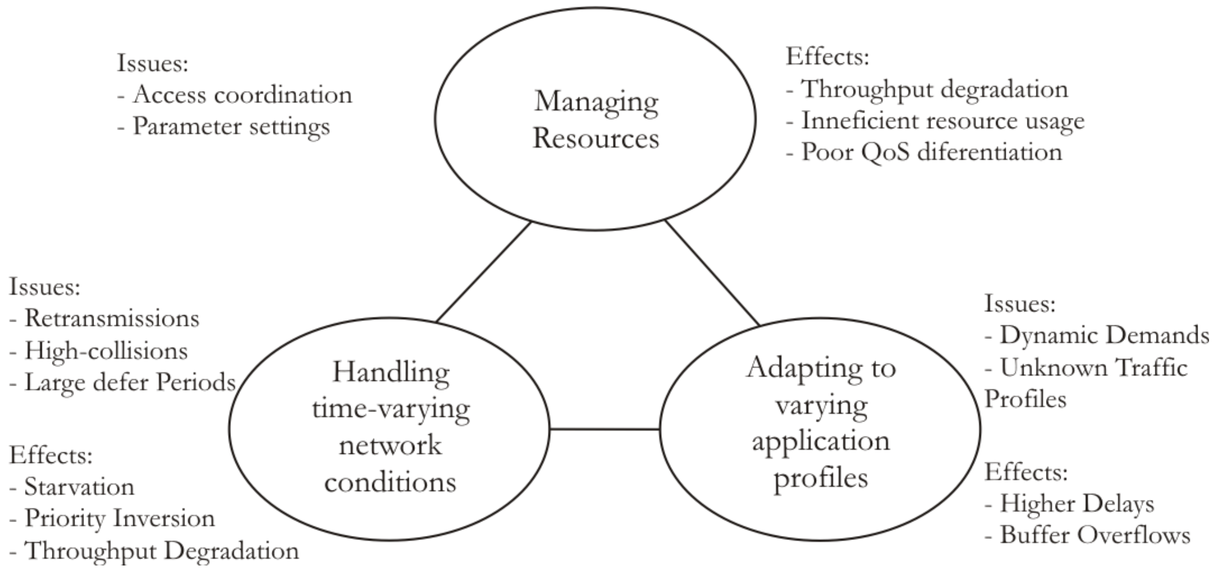


Figure 2.7. Issues in QoS support (adapted from [RaPD05]).

There are two different factors related to time-varying network conditions that are not directly addressed by the 802.11e standard: channel conditions and network load. Varying channel conditions, caused by propagations losses, multipath effects, interference, and mobility, may degrade the performance of a WLAN. This leads to the increase of retransmissions, packet losses and throughput reduction. Varying channel conditions may potentially degrade the differentiation among priorities, as high-priority stations experiencing poor channel conditions may achieve lower performance than a lower priority station. Network load, *i.e.*, the number of contending stations on a given WLAN, may also have an impact on service differentiation. WLANs use a shared channel access mechanism, hence, an increase in the number of stations within a network may affect the performance of the whole network. This effect may be reduced by using some kind of admission control, therefore, imposing some control on the number of stations accessing the network, or by adapting link-layer parameters to improve service differentiation under heavy load.

There has been some work regarding admission control in 802.11e, *e.g.*, [GaCN05], [ZLCP04], [GuMB06] and [GaCZ05]. The two different channel access mechanisms work in a different manner, admission control procedures being different for the contention based and the contention free modes. In [GaCZ05], the effect of using admission control in the contention free mechanism (HCCA) is analysed, a new admission control scheme being proposed that considers both average and instantaneous physical rates. This new scheme is compared with the reference one, a significant improvement in system capacity being shown with little QoS degradation. In [GuMB06], the performance of EDCA-based networks is analysed. Some enterprise specific scenarios were simulated in order to estimate channel capacity and acceptable load conditions. The effects of using admission control are addressed, being shown that admission control helps to maintain the quality of the admitted flows, allowing them to retain their characteristics, while un-admitted flows suffer from poor quality as the channel load increases. Finally, in [GaCN05] and [ZLCP04], a broader look is given to different admission control schemes that can be implemented in WLANs in general, and, more specifically, in the 802.11e ones. Reference implementations from 802.11e are used as a starting point, but some new solutions are also analysed, sometimes requiring changes in both PHY and MAC layers. Regarding link-layer parameter adaptation, some solutions are proposed in [MQTB04], focusing on improving performance by adapting both the CW values and the backoff time when the network load varies. This adaptation policy depends on AC, and so service differentiation is also improved.

The second analysed issue is the adaptation to varying application profiles. The traffic generated by applications, and their specific QoS requirements, define application profiles. Different applications will have different requirements for throughput, delay, or jitter, which may also vary over time, especially for multimedia applications, which, many times, use Variable Bit Rate (VBR) encoding. This creates some problems in the estimation of networks' available capacity, because traffic can vary significantly over time. This problem may affect both EDCA and HCCA, but it is more critical for very time-sensitive applications, which use primarily the HCCA mechanism, as this can increase delay during bursts of traffic. The HCCA reference scheduler uses a round-robin method to allocate transmission time (TXOP) to the different stations, based on the TSPEC information sent by each one of them. The reference scheduler takes only the average values into account, such as mean packet size or mean required throughput, which only works well for Constant Bit Rate (CBR) or near CBR applications. This is unsuitable for the dynamic traffic requirements of VBR applications, and so, QoS guarantees may be difficult to provide in some scenarios if the reference scheduler is used.

Some work has been done in order to find solutions for this problem, with the development of new scheduling algorithms that take traffic variability into account, [Ni05], [GrMN03] and [AnNT03]. The goal of these schemes is to adapt some of the scheduling parameters, such as the TXOP duration or the Service Interval (SI), *i.e.*, channel access mechanisms, to traffic variability. In [Ni05], a global overview is given to the enhancements proposed by the 802.11e standard, being seen that reference implementations, for both EDCA and HCCA mechanisms, have some weaknesses. Some adaptive approaches are then proposed to enhance the performance of both EDCA and HCCA modes in varying traffic conditions. In [GrMN03] and [AnNT03], it is shown that the 802.11e reference scheduler only works well for relatively constant traffic, some new solutions being proposed to enhance it. In [GrMN03], a solution that allows some variations in the TXOP duration is proposed, while still enforcing its mean duration; this way, the delay is kept fairly constant, even for bursty traffic, as long as the applications' mean sending rate does not increase. In [AnNT03], a different approach is taken, using queue length estimations to tune its time allocations (TXOPs) to the stations. If the traffic demand of a given station increases at a given instant, then, its TXOP duration is increased, the opposite being also true. This provides some degree of adaptation to the variability of traffic.

Finally, the last issue is the management of resources. The most important issue is the coordination between the EDCA and HCCA channel access techniques. For example, it should be useful to have some kind of mechanism to distribute the different flows by the channel access mechanisms, looking not only at the flows' characteristics, but also at the network load both in EDCA and HCCA. The usual approach is to restrict the more time sensitive applications to the HCCA mechanism, and leave the other applications to EDCA. However, this approach has some drawbacks, *e.g.*, when a higher priority flow from a given station experiences low throughput in EDCA, despite having available time in the HCCA period.

To address this issue, an algorithm for the coordination of the two access mechanisms is proposed in [RaPD04]. The proposed solution improves the QoS of multimedia applications by dynamically associating traffic flows to the channel access mechanisms and adjusting the duration of access in each one, *e.g.*, allowing an EDCA flow to transmit during the HCCA period if there is high load in EDCA, or by allocating additional HCCA time to a given already existing HCCA flow, if its delay is seen to be increasing.

Chapter 3

QoS Models and Algorithms

The solutions proposed within the 802.11e standard aim at providing QoS guarantees to applications, but some issues to enable a comprehensive QoS support to applications remain. In this chapter the OPNET Modeler simulation engine is described, together with its functionalities, and based on Modeler, some new solutions are proposed in order to improve the QoS support by 802.11e.

3.1. OPNET Simulation Engine

OPNET Modeler [OPNE07] is a powerful network simulation and modelling environment. It allows the modelling of a wide range of network types, technologies and topologies. Specifically, with the Modeler Wireless Suite, it is possible to analyse a broad range of wireless network types, namely the IEEE 802.11 networks under analysis in this work. In the following sub-sections, the most important features of Modeler are presented and analysed.

3.1.1. Modeler Architecture

The architecture of Modeler is based on a hierarchical structure with different modelling domains. Each modelling domain has a given editor associated to it, which focuses on a given set of objects, operations, and possible tasks, Figure 3.1.

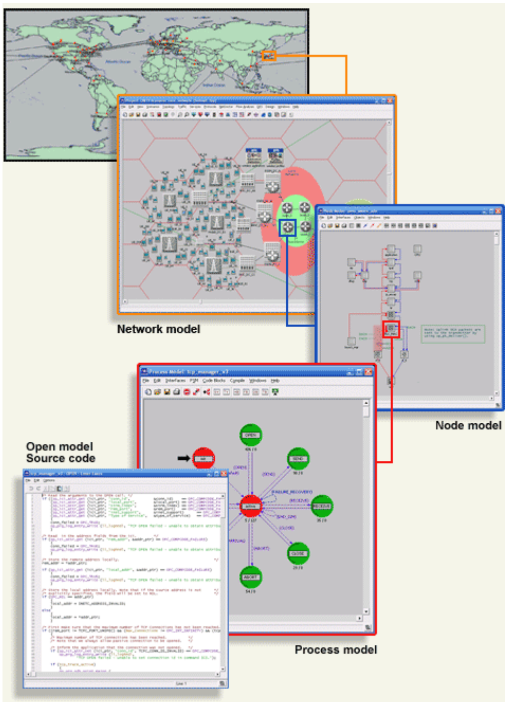


Figure 3.1. Modeler hierarchical structure (extracted from [OPNE07]).

The top level is the network domain and the associated project editor. The project editor has a central role in the modelling environment, and with it one can graphically represent the topology of an entire communications network. The network consists of node and link objects, which are instances of the lower level models, Figure 3.2. The network model also specifies the physical locations, interconnections, and configurations of the objects in the system.

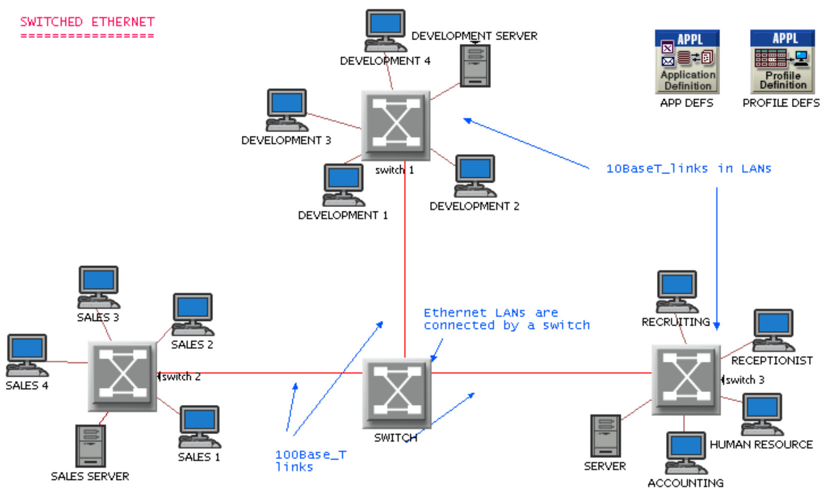


Figure 3.2. Network domain (extracted from [OPNE07]).

To reduce the complexity of large networks, the project editor has also the possibility to define sub-networks, which may be also viewed as a network with its own nodes and links. Sub-networks can be spread across many levels with the project editor, as long as the bottom level is composed only by nodes and links.

The second level in the Modeler Architecture is the node domain and the associated node editor. Node models are defined in the node editor and specify the internal structure of the node. They are composed of small functional elements, called modules, and of connections among them. Modules are used to represent, for instance, protocol layers and physical resources, such as buffers. Each module can generate, send, and receive packets from other modules to perform its function. In Figure 3.3, an example of a node model is shown, where modules and their connections are depicted.

Different kinds of modules are defined in Modeler. Modules like transmitters and receivers are predefined and cannot be changed, being configured by a set of specific parameters. But, there are other modules, like processors, that are programmable and that can be used to represent a wide range of different modules, like the MAC layer of a given station.

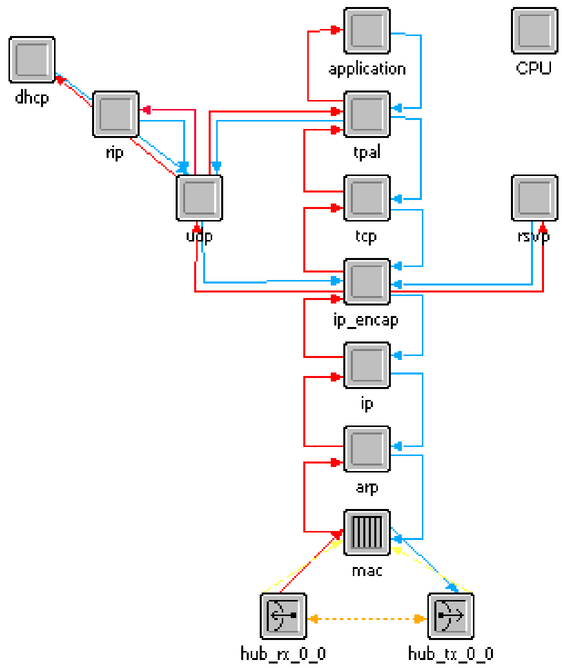


Figure 3.3. Node domain (extracted from [OPNE07]).

Regarding connections, they are used to allow interaction between the different modules. Three types exist: packet streams, statistic wires, and logical associations. The last one is used to connect transmitters and receivers, in order to indicate that they should be used together when attaching the node to a link in the network domain. The statistic wires are used to provide a one-way connection between two modules. They are used to transfer individual values that are dynamically updated by the source module. Finally, the packet streams are the most important connections, essentially being one-way pipes that are used to transmit packets between modules.

The last level in the Modeler architecture is the process domain, Figure 3.4. Within the process domain, process models are defined, which are used to specify the behaviour of processor and queue modules that exist in the node domain. Process models are driven by events, and as a given event occurs, an interrupt is generated and delivered to the process. This allows the process model to react and take some actions in response to the event.

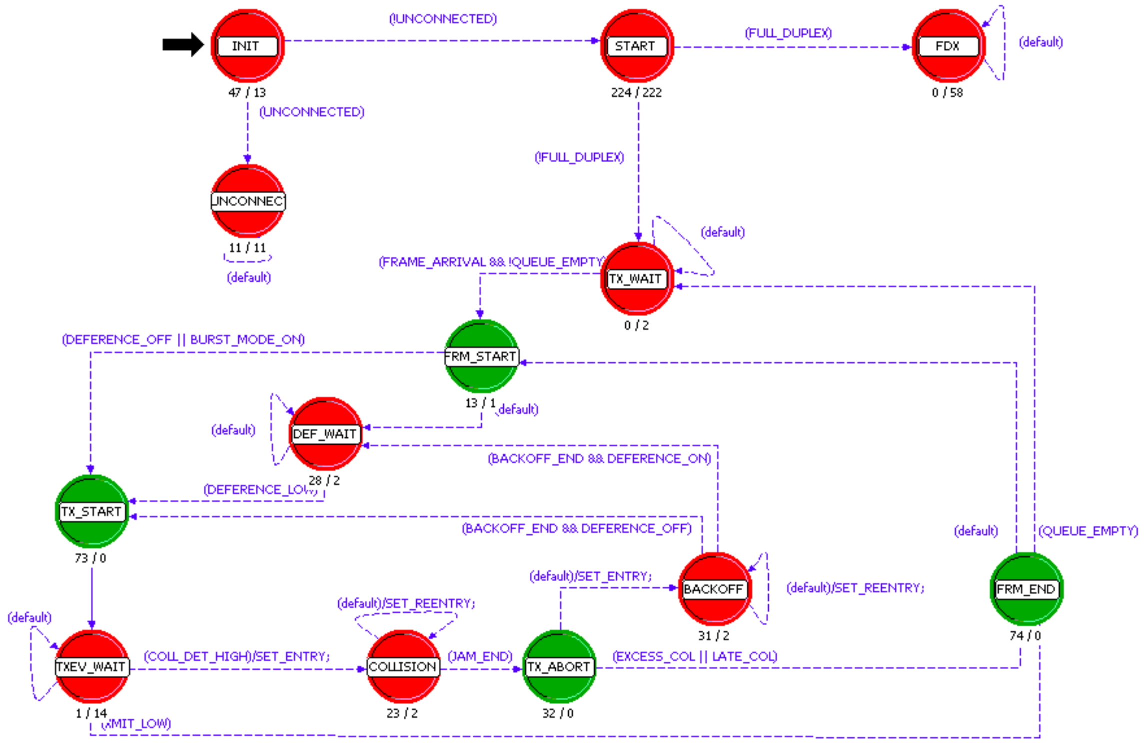


Figure 3.4. Process domain (extracted from [OPNE07]).

The process editor is the editor associated with the process domain, being used to develop process models. This is done by using a programming language called the Proto-C, which is a combination of finite state machines (FSMs), libraries of kernel procedures, and C/C++ programming language functions and variables. States and transitions within the FSMs are used to graphically represent the progression of a process in response to the various events. The kernel procedures and the C/C++ code are then used within the states/transitions to perform all the tasks related to the interrupt/event that has just arrived.

Within the process domain, there is also a number of extensions to expand the capabilities of the FSM, as state variables and state executives, and transition conditions and transition executives. Regarding the state variables, they are private variables within a process used for the process actions, while state executives are the actions performed by the process when a state is entered (enter executives) of left (exit executives). These actions are defined by using the C/C++ language and can be used, for example, to update some kind of statistics. The transition conditions and transition executives are used, as the name implies, within the transitions between states. Transition conditions are expressions that determine whether a given transition can occur or not, while transition executives represent the actions that are performed while executing a given transition.

The three domains briefly mentioned represent the basis of the Modeler simulation engine, but there are also some other tools available that allow to develop or to perform some specific tasks. Just to name a few, there is the packet format editor to design and specify new packet formats, the link model editor to create new link models, or the analysis configuration editor that allows one to plot and process the numerical data generated by simulations.

3.1.2. Modelling Events and Traffic

As previously mentioned, all the nodes in a given network being simulated are represented by modules, which in turn are instances of process models and their FSMs. The simulation evolves through the states of the FSMs as a function of time, as events occur, this evolution being representative of the way the actual system performs over time. The events symbolise the evolution of the simulation time, which has no direct relation with the actual time. The simulation time can just be considered as a variable that jumps among the scheduled events, Figure 3.5.

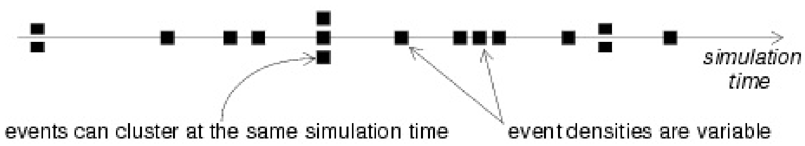


Figure 3.5. Evolution of simulation time (extracted from [OPNE07]).

All these discrete events compose the simulation structure, where the progression of the models through time is decomposed into individual points (events) where actions can take place. Each event represents a point where a given action must be taken, or where a given state change should be performed.

All events are managed by Modeler by using an event list, where they are ordered according to their execution time and priority, and the simulation kernel is the entity responsible for the insertion and removal of events from this list. Multiple events can occur simultaneously, *i.e.*, at the same simulation time, hence, all events must have duration equal to zero, even if they represent real events with a given duration. This duration can be represented, *e.g.*, with another event scheduled for a future time. As events are executed, simulation time is increased.

As the simulation starts, all the initially scheduled events are placed on the list, and the simulation control is delivered to the kernel. As a given event occurs, the simulation control is passed to the appropriate module, and its process model executes all the necessary actions in response to the interrupt generated by the event. These actions can result in the generation of new future events or in the cancellation of events already in the list. At the end of this event, the simulation control returns to the kernel, which deletes this already executed event from the list and advances the simulation to the next one. This cycle goes on until there are no more events scheduled, or until the simulated time reaches the predefined simulation duration.

As one may see, the size of the event list is continuously varying, as new events are generated/deleted and consumed by the simulation itself. These events are more than just a simple indication regarding the simulation activity and its associated simulation time. In fact, each event is a complex entity and a wide range of events types exist, representing different situations within a simulation. Each event by itself has a number of attributes describing how it should be executed, including, among others, the time, the identification, or the type of the process that will receive the event. These attributes are enough to differentiate and characterise the event. As a given process model receives an event to process, it uses these attributes to know which actions it must perform.

In order to simulate the behaviour of a given network and obtain performance results, one must add traffic to it. All the traffic being generated, transmitted, and received within the network is represented by events.

Modeler allows the addition of traffic in the network mainly in two different ways: either manually, by setting the attributes from the various applications/packet generators, or by importing traffic data from external files or programs.

Modeler has multiple options for importing traffic. This allows the simulator user to use real traffic in the network to obtain better and more realistic results for some scenarios. For example, one can import end-to-end background traffic from a captured traffic archive, which may represent the traffic from a real network. This may be helpful to understand what is the impact of changing the network configuration or of having traffic variations in a given existing network. Using real traffic data from an existing network makes results much more reliable and accurate.

Regarding the manual setup of traffic, two different types can be modelled: explicit and background. Background traffic is analytically modelled, thus, affecting the performance of explicit traffic by introducing additional delays. The effects of background traffic are modelled by calculating the increase in queue sizes and the additional delays based on queue lengths. Because packets are not explicitly modelled, the simulation speed is sharply increased.

Explicit traffic is "packet-by-packet traffic", where the simulation engine models each event related to every packet: generation, queuing, transmission, reception, etc.. This is the most realistic approach, as it allows every protocol effect to be modelled, but it has the drawback of resulting in longer simulations. Background traffic is not supported in the Modeler WLAN module, so only explicit traffic is analysed in further detail.

There are three general methods for the generation of explicit traffic:

- Packet generation: this is the most basic type, being the only possible one with certain simplified node types. These nodes have the ability to be configured to generate streams of generic packets, using arguments like the interarrival time or the packet size.

- Application demands: this traffic generation method can be created to represent a given traffic flowing between two nodes in the network. These demands characterise traffic in terms of the size and rate of the requests, and responses between these two nodes. This method is simpler than configuring application traffic models, because all that is needed for configuration is the demand itself, no additional one at the nodes being necessary;

- Application traffic models: this is the most complete method for the generation of traffic. Modeler includes a set of pre-defined common applications, such as FTP, HTTP and voice, that can be fully customised in order to represent a given usage profile within the network. There is also the possibility of defining custom applications, which allow one to create a given application that does not correspond to the patterns of the standard network applications already available in Modeler.

Modeler has two special node objects, defined to characterise applications traffic models, *i.e.*, the Application Definitions and the Profile Definitions nodes. The Application Definition node, as the name implies, is responsible for the definition of the available applications within the network. It allows the simulator user to choose which applications can be used with several common predefined applications; each application can be configured in great detail, with some usage patterns defined as well. After the definition of the available applications, it is possible in the Profile Definition node to define profiles that identify which of the available applications can be used, and how these applications are used, *e.g.*, its repeatability. Again, some predefined profiles are available.

3.1.3. Results Collection

After the careful planning and design of the network to be simulated, the objective is to obtain results to see how the network behaves under various circumstances. Modeler has several different mechanisms that allow the simulator user to have flexibility in the way results are obtained and shown. There are four output types:

- Output vectors: a given variable can vary as the simulation time increases, and output vectors are used to record this behaviour. This way, each output vector is composed of a series of values and associated times. Multiple independent output vectors can be collected simultaneously.

- Output scalars: certain metrics of interest do not vary over time; instead, they have one value that is representative of some kind of system performance, as averages or standard deviations. Each scalar is usually recorded only once for a given simulation, but scalars from multiple simulations can be combined to analyse the dependency on a given simulation input.

- Animations: this type of output is not a numerical statistic. This is a method to visualise system behaviour and interactions among system components. Simulations can generate animations as they run in order to be shown after the end of simulation. With these animations, it is possible to see, *e.g.*, packets flowing across the nodes of the network, the movement of the nodes, or state transitions within a given process.

- Proprietary reports: a given user can include user-defined processes or link models, and define its own outputs and reports to be generated by Modeler. This can be done by using the general functionalities of the C/C++ programming language.

Before a given simulation starts, the simulator user must select which parameters should be collected from a very large number of possible statistics and metrics, Figure 3.6. Each of the modules at the node and link levels is a source of a significant number of available statistics. The selection of statistics is performed by specifying a list of probes, which perform the selection and control the flow of data from the selected statistics into output files. All these statistics and performance metrics are clearly defined in Modeler, the definitions of the ones used in this work being presented in Annex D. For the scope of this work, several statistics, like throughput, delay, and dropped data, both global and by application, are recorded, in order to assess the effects of using QoS mechanisms

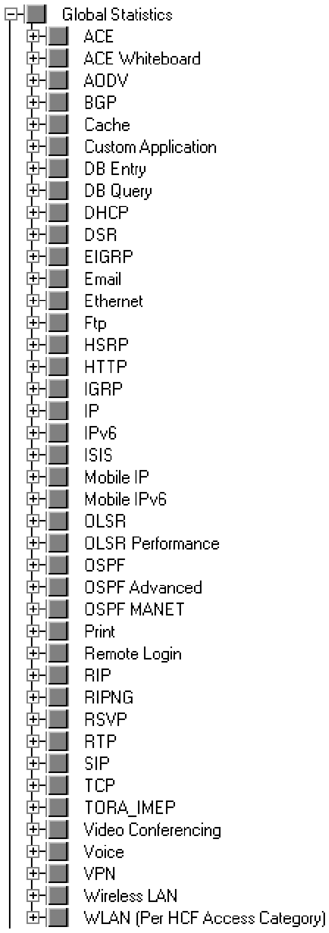


Figure 3.6. Available statistics.

After the collection of all the desired data during a given simulation, Modeler gives the ability to analyse and visualise it in various ways, *e.g.*, traces of a given output vector can be presented to the user, and these traces can be changed to show the average along the time, the Probability Density Function (PDF), Cumulative Density Function (CDF), and so on. Modeler also gives the option to export the data from the trace to an outside viewer, *e.g.*, Excel.

3.1.4. Modeler Wireless Module

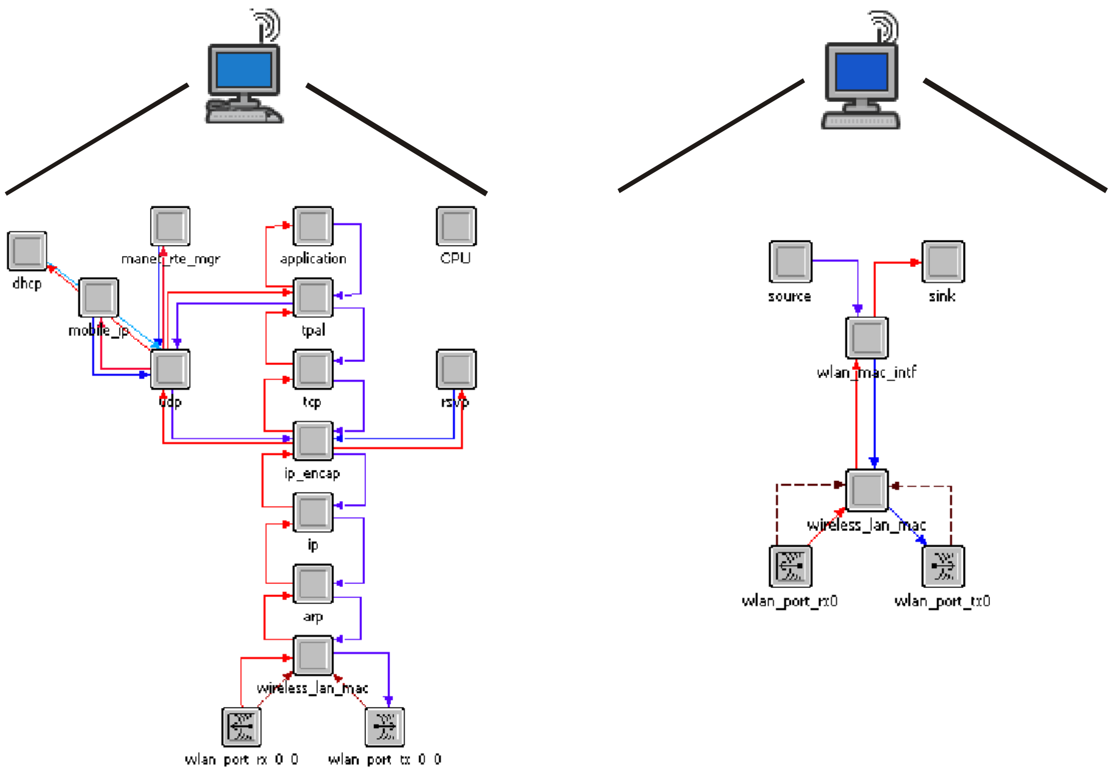
This subsection is dedicated to the Modeler wireless module, and more specifically to its WLAN models. The Modeler wireless module makes available a wide set of models that allow the simulation of scenarios with both node mobility and radio communications. These include cellular networks, WLANs, satellite communications, and others.

Modeler's WLAN module supports the IEEE 802.11, 802.11a, 802.11b, 802.11g and 802.11e standards. Some of the main features of the model include:

- DCF and PCF access mechanisms;
- IEEE 802.11e support (EDCA mechanism);
- RTS and CTS protection mechanisms;
- Fragmentation and reassembly of packets;
- Roaming;
- Normal ACK/Block ACK/No ACK mechanisms;
- Support for mixed environments (IEEE 802.11b, 802.11g and 802.11e can coexist in the same environment);
- Multiple physical layer technologies;
- Multiple data rates/modulations.

The WLAN module has several models of routers, stations, or bridges available, and all can be configured in order to satisfy different needs from the simulator user. For example, regarding the stations, there are two different standard models available: *wlan_wkstn* and *wlan_station*, Figure 3.7.

The difference between these two modules is that the former implements all layers from the physical layer to the application one, while the latter just implements the 802.11 protocol together with a simple packet generator and a sink to generate and receive traffic. These models, as well as, *e.g.*, the router models, can also be configured to act as the AP.



a) *wlan_wkstn* model b) *wlan_station* model

Figure 3.7. Standard stations models.

The two station models presented are just an example of the available models in the Modeler standard library. Depending on the scenario, other modules can be used, as routers, servers, or bridges, or it may be necessary to change any of the already existing models, or even to define a new one to add some features.

Regarding the IEEE 802.11e standard implementation, Modeler (version 12.0 PL5) only supports the EDCA channel access mechanism, as it has been shown. New features can be implemented by any given simulator user, thus, the HCCA channel access mechanism can be implemented. But this is an enormous task, as the already existing MAC layer process model with EDCA support has over 15 000 lines of code. So, given the framework of this thesis, it was decided to focus just on the EDCA mechanism.

All features regarding the EDCA mechanism are implemented, with four ACs defined (Voice, Video, Best Effort and Background) for prioritised contention-based access. Additionally, the TXOP frame bursting is also supported. In a given network with QoS-enabled stations, all the usual parameters, *e.g.*, CW_{min}, CW_{max}, AIFSN or the TXOP limits can be individually configured for each AC in every station. Also, Modeler supports the EDCA Parameter Set distribution, which allows the AP to use a set of different parameters for the various ACs compared to the stations.

Different MAC-level acknowledgment mechanisms can be used as well. Modeler supports the normal ACK, the Block ACK and the No ACK mechanisms. Different ACK requirements can be used for each TC. When using the Block ACK mechanism, there is also the option to use immediate or delayed Block ACK, and it supports roaming.

Modeler's wireless module has also some limitations, *e.g.*, it has no authentication and security procedures, DLS is not supported, and the IR PHY is not available. But the most important limitation, regarding this work, is that the transmission rate used for data transmission by a WLAN node is fixed during the simulation, *i.e.*, there is no transmission rate variation due to channel variations. This means that the selected transmission rate is used in all frames, by all nodes, during the simulations.

3.2. QoS Parameters and Algorithms

As mentioned before, the 802.11e standard proposed some new features in order to provide QoS guarantees to applications. In this section, the influence of these parameters on the behaviour of the network is addressed and possible QoS algorithms are analysed.

The system parameters, which are responsible for the QoS guarantees and services prioritisation, are the AIFSN, CW_{min} and CW_{max}, TXOP duration and the ACK mechanism used. All these parameters have default values defined in the 802.11e standard, which aim at providing a certain degree of compromise between QoS provision to applications and overall network efficiency. This work aims at evaluating the effects of the variations in these parameters, in order to choose the best possible set of parameters for QoS provision. But the traffic in the network is also of great importance, as different traffic profiles also have a different impact on the overall QoS within the network. So, the effect of the traffic to QoS provision is also analysed.

Concerning system parameters, we start by looking to the most important ones, *i.e.*, AIFSN, CW_{min} and CW_{max}. These three parameters control how a given station/AC will access the medium, how long it will wait, and how often it will try again to use it. Starting by AIFSN, using a larger value, in a given AC, will increase the amount of time that the AC must wait, while trying to use the wireless medium. This may be helpful to differentiate traffic, as higher priority traffic can use lower values for AIFNS, thus, waiting less time before transmitting. But, it is expected that varying AIFSN will only have visible effects if its variation is large, because the AIFSN value is directly linked to the time-slot duration, which is very small. Also, if a large value is used, then the waiting time to access the medium will also be large, which is somewhat inefficient. The AIFSN value can be used to enhance the differentiation between traffic priorities, but in order to keep the overall efficiency of the network, it will not be possible to vary it much.

Concerning CW_{min} and CW_{max} values, they will have an impact on the waiting time to access the channel, when the channel is found to be busy, or when collisions occur. Changing the CW_{max} value will only have an impact when the network is highly loaded, as the stations only reach the CW_{max} value after several collisions. So, the option to change CW_{max} (increasing it) is only useful when the network is highly loaded and for low priority traffic. As for the CW_{min} value, it will have much more impact on the network performance, and on QoS in particular. Changing this value for a given traffic category will grant it more or less priority to access the medium, as the value is changed. This happens because the CW_{min} value is the parameter with the highest impact on the time a given station must wait before accessing the channel, when multiple stations are competing for the medium. The smaller the CW_{min} value is, the higher priority a given station will have; however, if all stations have very small CW_{min} values, then, more collisions are likely to occur, especially as the network load increases.

One other parameter that has influence on the QoS performance is the definition of the TXOPs, specifically the definition of the TXOP maximum duration. Larger TXOP values help to increase the overall network throughput, but also decrease the number of stations that are able to use the medium on a given time period. Also, if the TXOP values are large, then, the more time-sensitive applications may have problems, because they are unable to transmit for some time when another station is transmitting. A trade-off must be obtained between the overall network efficiency and stations QoS requirements.

All parameters mentioned so far can be set independently for each of the ACs for each station, which gives an extra level of flexibility while setting QoS parameters. The default parameters defined in the standard take this into account, in order to give more priority to the Voice and Video traffics, Table 2.4. For example, as it has been seen, Voice traffic will also have smaller TXOP duration than the Video one, because usually it generates smaller frames.

Modeler allows each station to have its own set of values for the parameters just described. But there is also the possibility of having the AP choosing the values for all stations, and using a different set of values for all the parameters. This may be helpful to give more priority to the downlink traffic, to decrease the bottleneck effect caused by the fact that all traffic must go through the AP.

Finally, there is another parameter, not directly related to QoS provision, but that has more an efficiency enhancement effect, which is the ACK mechanism. The 802.11e standard defined new Block ACK and No ACK mechanisms in order to enhance the efficiency of the network in some conditions. The No ACK mechanism, as the name implies, disables the sending of an ACK frame for every data/control frame received. This may sound an odd policy, as the wireless medium is highly variable and unstable, but sometimes, for very delay-critical applications, it is more important to send the data with the less possible delay, than to have a reliable exchange of data.

The Block ACK mechanism works in a different manner, replacing the sending of an ACK frame for every data/control frame received by the sending of a single ACK frame for a given block of frames. This works while using TXOPs, as only a single ACK is required for all the frames received during the TXOP. This will increase the overall efficiency of the network. There are two different possibilities regarding the Block ACK mechanism, *i.e.*, one can send the ACK frame in the same TXOP (immediate Block ACK), or send it in the next TXOP (delayed Block ACK).

3.3. Simulator Implementation in OPNET

In the previous sections, a global overview of OPNET Modeler and of QoS algorithms is presented. In this section, the focus is on the implementation in the simulator, namely the insertion of algorithms in Modeler.

The first step is to setup the scenario and network topology. The considered scenario is a square office with 100 m side (predefined office topology within Modeler), then, the next step is to place some stations in the scenario. This number varies in the various simulations, but one starts by placing just one station (*wlan_wkstn* mode, Figure 3.7).

Stations use applications, so one needs a server for the stations to communicate to. The first option is to have a server with wireless capabilities and make it the AP of the network, but this option is not a very realistic one, because, usually, the AP and the server are two distinct elements. Another option is to keep the server with wireless capabilities so that it can communicate with the AP using the wireless medium, but, again, this is not a very common approach. So, the final option is to use a simple server (*ethernet_server* model) with an Ethernet connection, and use a different element as the AP. As the server has only an Ethernet interface, the best option is to use a router with two interfaces: wireless and Ethernet ones. Such device is already available in Modeler, *i.e.*, the *wlan_ethernet_router* model, which has also the possibility to be configured as an AP, so it was set as the AP of the network. To connect the router and the server, an Ethernet link was used; since the focus of this work is QoS in WLANs, the fastest Ethernet link available (gigabit Ethernet – *1000baseX* model) was chosen, to minimise the effects of the Ethernet link on the final results.

Once completed the network basic configuration with all the needed communication nodes, one adds traffic to the network, via the Application Definition and Profile Definition nodes, Figure 3.8.

Within the Application Definition node, the definition of the available applications/services is performed and each individual application can be configured in detail. A set of 8 types of services are predefined in Modeler: HTTP, FTP, VoIP, Database Access, E-mail, Video Streaming, Remote Login, and Print. Each of these services can be fully configured, *e.g.*, in HTTP one can configure the protocol version used, the page interarrival time, and the page properties, just to name a few. For these predefined services, Modeler has already defined some types of application usage, that is to say, some application definitions. For example, in HTTP, one can select the Light Browsing, Heavy Browsing, Searching and Image Browsing application definitions, which can all be changed to adapt to some usage scenario. If this is not enough, or if there is the need to define some new applications, there is the possibility for the simulator user to create new custom application definitions and to configure them.

After the definition of the applications, one needs to define how they are used by the users (the stations). For such, the Profile Definition node is used, and Modeler has already a set of profiles defined: Engineer, Researcher, E-commerce Customer, Sales Person and Multimedia User. These profiles, as the name implies, describe which of the available applications are used by the users, and how these applications are used, *i.e.*, when they start, its duration, its repeatability, and so on.

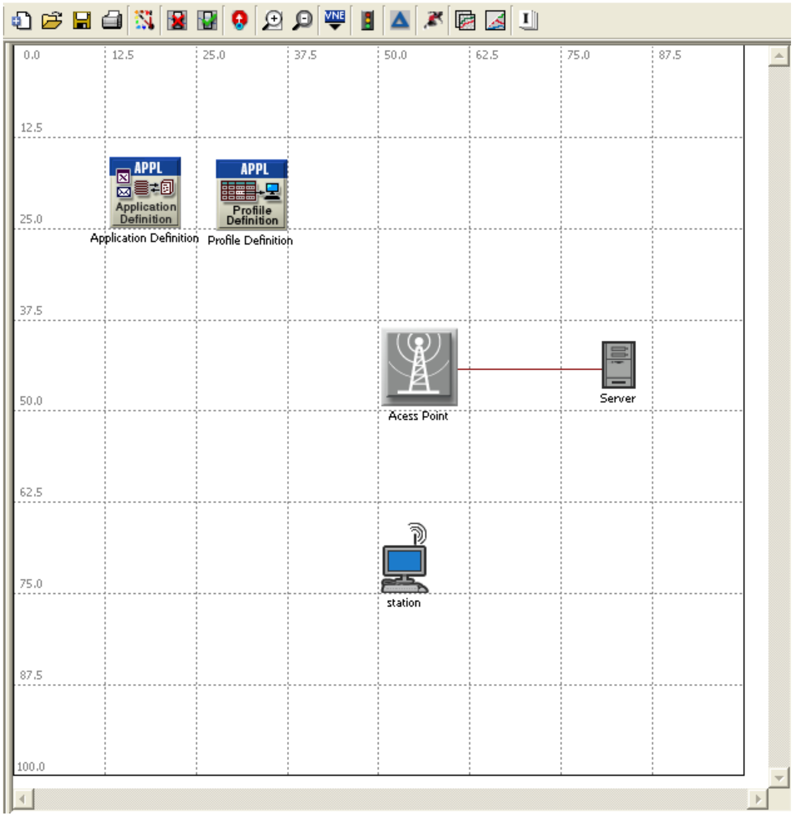


Figure 3.8. Simulation scenario.

In this work, several types of applications are used, with the main focus on real-time or near real-time applications (voice and video), together with common traffic, such as HTTP and FTP. Some profiles are defined as well. Most of these parameters are based on scenarios proposed in the AROMA Project, [Ljun06], the specification of all these applications, profiles and usage scenarios, being presented in the next chapter.

One needs also to select which statistics are used to evaluate network performance, and how the different parameters impact on these results. The main statistics for this purpose are the delay and throughput of the applications/stations, but other parameters, like the traffic sent and received by the stations, or the size of their queues, are also important. When using the 802.11e standard, all these statistics can be collected by each of the ACs, which helps to understand if the differentiation of traffic according to its priorities is working as it should.

All the various parameters that define the behaviour of the network nodes (stations, AP and Server) can be accessed and changed directly through the main interface of the project editor. These include all the WLAN parameters, Table 3.1, and the system parameters analysed in the previous section, Figure 3.9.

Table 3.1. WLAN Stations/AP parameters.

Attribute	Value
BSS Identifier	1
Access Point Functionality	Disabled (stations) / Enabled (AP)
Physical Characteristics	Direct Sequence (802.11b)
Data Rate [Mbps]	11
Bandwidth [MHz]	Physical Technology Dependent
Min. Frequency [MHz]	BSS Based
Transmit Power [W]	0.005
Packet Reception-Power Threshold [dBm]	-95
RTS Threshold [bytes]	None
Fragmentation Threshold [bytes]	None
CTS-to-self Option	Enabled
Short Retry Limit	7
Long Retry Limit	4
AP Beacon Interval [s]	0.02
Max Receive Lifetime [s]	0.5
Buffer Size [bits]	256000
Roaming Capability	Disabled
Large Packet Processing	Drop
PCF Parameters	Disabled
HCF Parameters	Enabled/Disabled (Scenario Dependent)

Each parameter can be defined when the network is configured, but Modeler has also the ability to "promote" one or more parameters, *i.e.*, it allows one to delay its definition until the setup of the simulation itself. This makes it easier to run a set of independent simulations sequentially, using multiple values for the promoted parameters. The results from this set of simulations can then be presented together to the simulator user, allowing for a faster and easier comparison of the changes caused by the variation of the parameter.

⊟ HCF Parameters	(...)
├ Status	Supported
⊟ EDCA Parameters	(...)
⊟ Access Category Parameters	(...)
⊟ Voice	(...)
├ CWmin	(PHY CWmin + 1) / 4 - 1
├ CWmax	(PHY CWmin + 1) / 2 - 1
├ AIFSN	2
⊞ TXOP Limits	(...)
⊞ Video	Default
⊞ Best Effort	Default
⊞ Background	Default
⊞ Traffic Category Parameters	(...)
├ Block ACK Capability	Supported
⊟ AP Specific Parameters	(...)
⊞ Parameters Advertised in BSS	Default

Figure 3.9. EDCA parameters setup.

This base scenario is the basis for all other scenarios that are analysed, but before the definition of these simulation scenarios and profiles, one must define the main aspects of the simulations, *i.e.*, the stabilisation period and the minimum number of simulations that must be performed in each scenario, in order to have some statistical relevance. The objective is to make one hour simulations for each scenario, as it is considered that one hour is a large enough period to have meaningful and stable results in the network.

In order to do so, 30 simulations were performed for the base scenario (defined in detail in the next chapter) with 75 minutes each, using 30 different and randomly chosen seeds. In order to see the length of the stabilisation period, the global delay of the simulations was calculated, Figure 3.10. As one may see, as the simulation begins, a peak can be observed in the global delay, but as time goes on, the simulation stabilises, and delay drops to near zero values. Based on these observations, it was decided to remove the first 5 minutes from the simulation results, in order to give time to the network to stabilise, and to remove these initial fluctuations of the results.

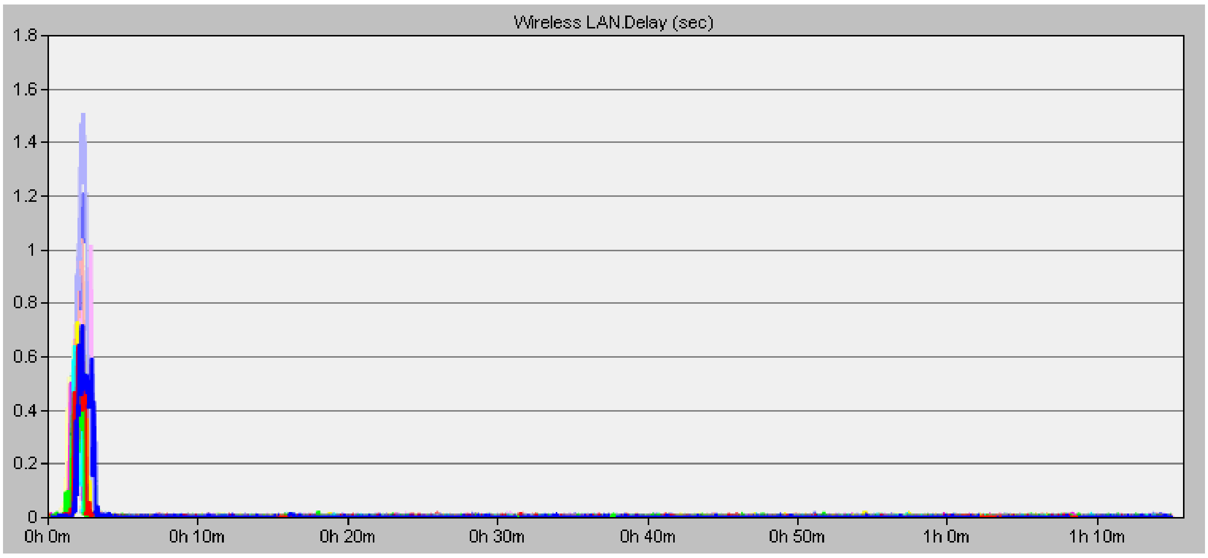


Figure 3.10. Global WLAN delay from the 30 simulations set.

The average global throughput of the whole network was then calculated, for each simulation, without considering the initial 5 minutes in each one, and finally, the cumulative average was also computed. In Figure 3.11, one presents the relative difference between the cumulative average and the global average as the number of simulations increases. This error is calculated as the relative difference between the cumulative average and the global average throughputs, *i.e.*, the relative error at simulation N (ε_N) is calculated as the difference between the cumulative average of the N simulations (μ_N) and the global average ($\bar{\mu}$), as follows:

$$\varepsilon_N = \frac{\mu_N - \bar{\mu}}{\bar{\mu}} \tag{3.1}$$

As one may see, with at least 10 simulations the relative error is low (less than 3 %) and the variation of the error between simulations is also small, hence, it was decided to take 10 simulations in each of the analysed scenarios. This number of simulations was also chosen as it was seen that simulation durations are sometimes very large in some scenarios/profiles, and this would be a limiting factor for testing more scenarios if more simulations would be performed for every scenario.

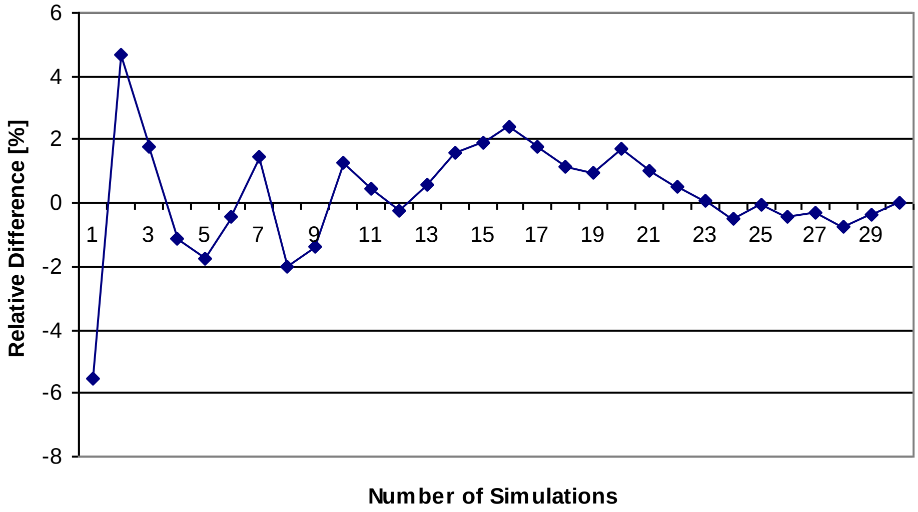


Figure 3.11. Relative difference between the cumulative average and global average as the number of simulations increases.

Chapter 4

Simulation Scenarios and Results

In this chapter, all simulated scenarios are presented, starting by the definition of all the parameters of the reference scenario, *i.e.*, services, their definition and usage profiles, and also the number of users. Then, using the reference scenarios, variations are performed on the number of users, usage profiles and system parameters. Finally, the results from all these simulations are presented and analysed.

4.1. Simulation Scenarios

In this section, the definition of the simulation scenarios is detailed. One starts to define the base/reference scenario, identifying the number of users, describing the applications and the usage profiles. Then, the simulated scenarios are described, identifying the changes made on the base scenario. Two major situations are considered in the definition of scenarios: variation of users (number, applications and usage profiles), with and without using QoS (EDCA enabled and disabled), and variation of system parameters (QoS parameters, like CW duration, AIFS or TXOP duration). The first situation aims at analysing the effect of users' parameters, and the advantages and disadvantages of using EDCA in the network, while the latter aims at improving the QoS provided to applications compared to the default values defined in the standard.

4.1.1. Reference Scenario Definition

The base scenario used in the simulations is, as previously stated, based on the AROMA Project, [Ljun06]: a "hotspot scenario" is defined, with 11 APs and 250 WLAN-capable users. For the scope of this thesis, a simple scenario with only one AP is needed, so, the average number of users per AP (23 users) was selected for the simulation base scenario.

There are 6 different applications in the set defined in AROMA: VoIP, Video Streaming, Video Telephony, HTTP, FTP and Email. The distribution of users by the different applications (as each user has only one application) is presented in Figure 4.1. The business profile defined in AROMA was followed, but the values used in this work do not exactly match the ones presented in the project, because AROMA has one extra application defined (Multimedia Messaging Service - MMS), which is only for mobile cellular communication users.

As one can see in Figure 4.1, real-time applications (VoIP and Video Conference) are responsible for 60% of the number of users, which is quite a large value for a typical WLAN scenario, and can be explained by the fact that the project considers not only WLAN users, but also mobile cellular ones. Even so, for the scope of this work, it is interesting to have a large number of users with these higher priority applications, in order to analyse the effect of the QoS algorithms.

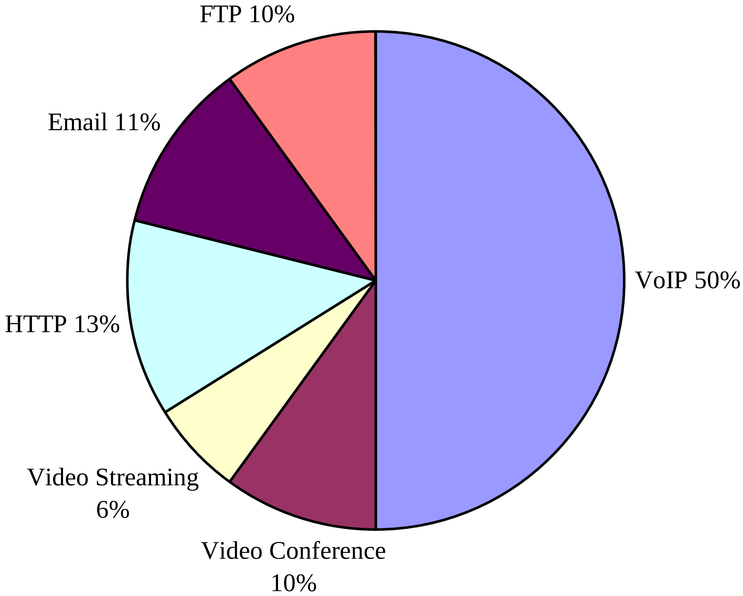


Figure 4.1. Services distribution (adapted from [Ljun06]).

The default values of Modeler for characterising applications were used as a starting point, different values and application models being used when the default values were found to be unrealistic (Video Streaming, Video Conference and Email), [PeEb02], or when better models were found in the literature (HTTP and Voice), [ChLi99], [Ljun06]. In Table 4.1, the most important values used to characterise each of the applications are presented, with a complete description of all parameters used in these application models available in Annex A, and the usage profiles presented in Annex B.

Table 4.1. Applications characterisation.

Application	Mean Call Duration [s]	Bitrate [kbps]	Mean Time between Requests [s]	Mean Request Volume [kB]
VoIP	120	8 (max)	--	--
Video Conference	120	64	--	--
Video Streaming	250	400	--	--
HTTP	--	--	39.5	189.4
FTP	--	--	360.0	2450.0
Email	--	--	600.0	100.0

The final aspect of the scenario, with all 23 stations distributed by the 6 defined applications, is presented in Figure 4.2. The position of all users is randomly distributed over all the working space, and the applications distribution by the users is made by assigning applications sequentially to the generated users.

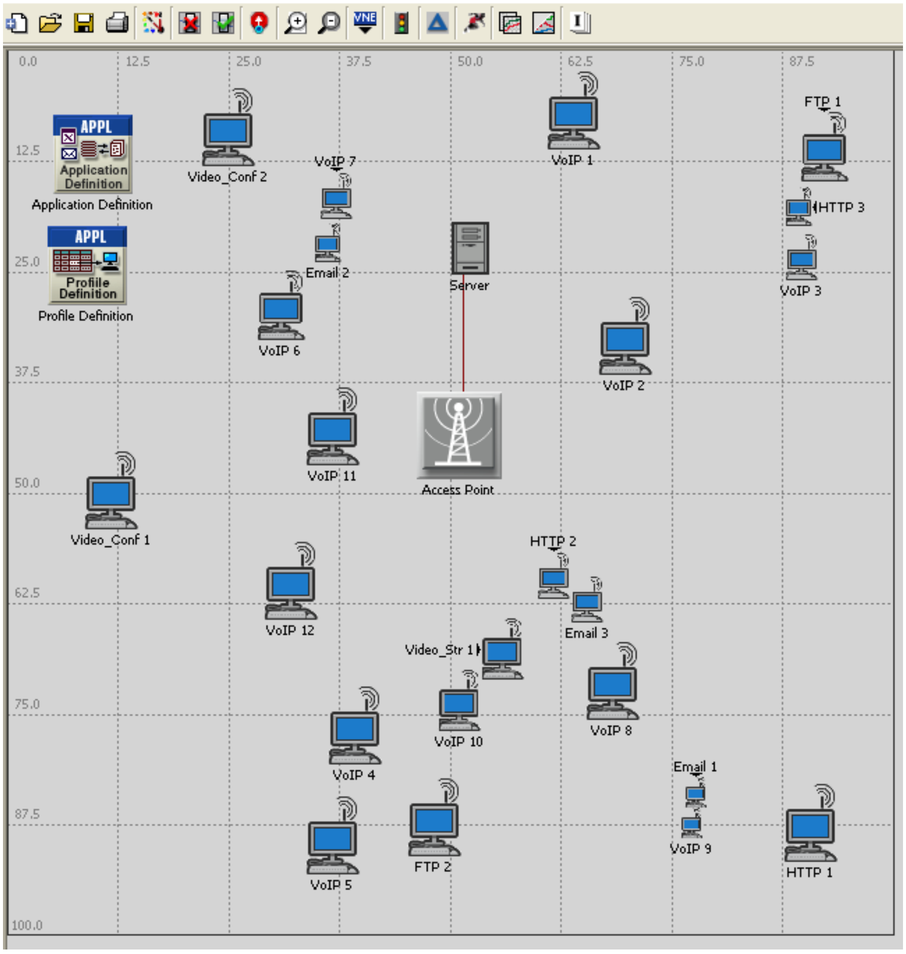


Figure 4.2. Base Scenario.

4.1.2. Users and Profiles Variations

Taking the base scenario, some more scenarios were defined by varying user profiles, that is to say, having scenarios with different weights on real and non real-time applications, Table 4.2. These scenarios are useful to analyse the impact of real-time applications on the overall network performance. As the relative weights of the real and non real-time applications were varied among scenarios, the percentages of the individual applications were varied accordingly in the base scenario, keeping their relative weight within their own category, Table C.1.

Table 4.2. Simulation scenarios: user profile variation.

Scenarios	Real-Time Maximum (RTM)	Real-Time Centric (RTC)	Base Scenario (BS)	Non Real-Time (NRT)	Non Real-Time Centric (NRTC)	Non Real-Time Maximum (NRTM)
Real-Time [%]	80	70	60	40	30	20
Non Real-Time [%]	20	30	40	60	70	80

In all the previous scenarios, each application maintains always its relative weight inside its category, but, some scenarios with different applications weights were defined as well. In these scenarios, some applications had their weights increased in order to create different usage scenarios, *e.g.*, a HTTP + Email centric scenario, with an increased number of HTTP and Email users, Table 4.3 and Table C.1.

Table 4.3. Simulation scenarios: services profile variation.

Scenarios	HTTP + Email Centric (HEC)	HTTP + Email Maximum (HEM)	HTTP + Video Str. Centric (HVSC)	HTTP + Video Str. Maximum (HVSM)
Real-Time [%]	20	10	20	10
Non Real-Time [%]	80	90	80	90

All these scenarios are analysed for the same number of users, 23. To evaluate the impact of the increase of the number of users, the same scenarios were simulated when this number is varied, Table 4.4. One must also note that, for every scenario, there is always, at least, one user in each of the available applications, regardless of the weight of the applications in the scenario. For the 150 % and 200 % increase cases, only the most representative scenarios (RTM, BS and HVSM) were simulated, because the simulation duration for these scenarios is very large (up to 5 hours per simulation), and because the network will be so much loaded that it will be unusable.

Table 4.4. Simulation scenarios: number of users variation.

Scenarios	Number of Users
Reference	23
30% increase	30
50% increase	35
80% increase	41
100% increase	46
150% increase	58
200% increase	69

4.1.3. System Parameters Variations

Scenarios were also established to analyse the impact of the various system parameters defined within the 802.11e standard. The QoS mechanisms system parameters were varied to analyse their effect on the network global performance. The default values for the TXOP duration, AIFSN, CW_{min} and CW_{max} were used as reference, and then, scenarios were established by varying each one of these parameters.

These variations were studied on some of the profile variations defined earlier. Three profile variations were chosen as the most representative ones: the Base Scenario, the Real-Time Maximum scenario (the most demanding in terms of delay) and the HTTP + Video Streaming Maximum (the most demanding in terms of throughput). So, all of the following scenarios were simulated with these three application profiles variations, in order to analyse the impact of system parameters variation when the traffic mix varies.

For each of the four parameters previously referred to, several variations were considered by changing the relative priorities of each of the ACs, Table 4.5 to Table 4.8. These variations include the increase of not only both voice and video priorities, but also of the relative priorities of the lower priority classes. This way, one can analyse the global impact of each of the system parameters. Also, it is expected that the system parameters variation will have a greater impact with higher network loads and so, the number of users is varied, with simulations being made using the reference number of users, 50 % increase, and finally 100 % increase.

Table 4.5. Simulation scenarios: TXOP variation.

	Scenario				
AC	**TXOP1**	**TXOP2**	**TXOP3**	**TXOP4**	**TXOP5**
0	0	0	0	0	1
1	0	0	0	1	1.5
2	3	3	6	6	6
3	1.5	3	1.5	3	3

Table 4.6. Simulation scenarios: AIFSN variation.

	Scenario		
AC	**AIFSN1**	**AIFSN2**	**AIFSN3**
0	10	14	14
1	5	5	10
2	2	2	5
3	2	2	2

Table 4.7. Simulation scenarios: CW_{min} variation.

	Scenario			
AC	**CWmin1**	**CWmin2**	**CWmin3**	**CWmin4**
0	63	63	31	31
1	31	63	31	7
2	15	15	7	3
3	7	7	3	1

Table 4.8. Simulations scenarios: CW_{max} variation.

	Scenario			
AC	**CWmax1**	**CWmax2**	**CWmax3**	**CWmax4**
0	2047	2047	1023	1023
1	1023	2047	1023	511
2	31	63	15	15
3	15	15	7	7

Finally, it was decided to perform some simulations to evaluate the effect of the ACK policies on the overall network performance: the Block ACK (immediate and delayed ones) and the No ACK policy. Scenarios were defined in Modeler to enable their comparison, but, unfortunately, it was impossible to perform these simulations, as Modeler always crashed without apparent reason when they were running. So, it was impossible to study the impact of these new ACK policies defined in 802.11e standard.

4.2. Results from the Users and Profiles Variation

As previously mentioned, several scenarios were defined for a varying number of users in the network, as well as for various usage profiles, the existence of QoS (EDCA mechanism enabled/disabled) being considered in parallel. The combination of these variations led to a total of 112 simulation scenarios, which amounts to a total of 1 120 simulations. The total time spent performing all these simulations reached more than full 24 days, with average times per simulation varying between 5 minutes to more than 3 hours, using a dual-core AMD Athlon X2 3800+ with 1 GB of RAM, with all these data occupying 1.3 GB.

It is expected that the most demanding scenarios, in terms of resources usage, are the ones with more streaming service (more bandwidth demanding), and the ones with more real-time applications (more delay stringent and more uplink bandwidth required). Looking, for instance, at the global average throughput and global average delay, one can identify these most resource demanding scenarios.

By analysing the throughput, Figure 4.3, one can see the clear increase with the number of users, with similar results in all of the scenarios and with larger variations when there are more non real-time users. The network load is not very high in the majority of the scenarios, with the maximum observed average global throughput of around 3.5 Mbps, when increasing the number of users by 200 %. For a number of users up to the 100 % increase, the throughput is always below the 2.5 Mbps mark.

Also, one can see no clear distinction between the QoS enabled/disabled scenarios, even when increasing the number of users. Although this might lead one to think that there are no throughput gains when using the QoS mechanisms, this is not true. Figure 4.3 presents the global average throughput over the whole simulation, and in each scenario, just by enabling/disabling the QoS mechanisms the traffic that each of the users generates does not change. To analyse the throughput variation, one also presents the ratio between the throughput average and standard deviation, Figure 4.4. Values are above the unity for most of the scenarios, meaning that simulations are more stable. This can be explained by the fact that as the number of users increases, there are less silent periods in the network due to the load increase, and global throughput becomes more stable, as there are always users transmitting data.

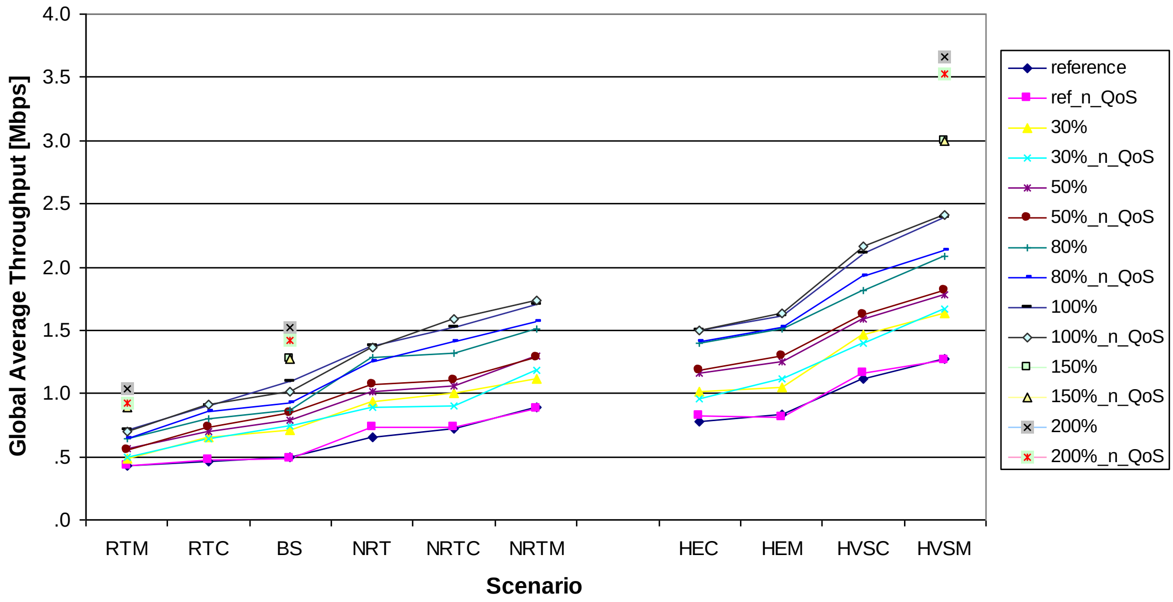


Figure 4.3. Global Throughput results by scenario and number of users.

However, there are other results that show that QoS mechanisms can, in fact, improve the throughput of the network, at least in some of the scenarios. Looking, for instance, at the maximum throughput achieved in all scenarios, one can see a clear effect of the QoS mechanisms, even while taking into account that these values are obtained from a single simulation, *i.e.*, one single simulation can vary a lot, but even so, the overall tendency in all scenarios shows the effect of using QoS. In Figure 4.5, the comparison between the maximum throughput per scenario with and without QoS is depicted: in the majority of the situations, maximum throughput values are higher when using QoS mechanisms, this being more tangible in the scenarios with less real-time applications, *i.e.*, higher average throughputs, and especially when the number of users increases.

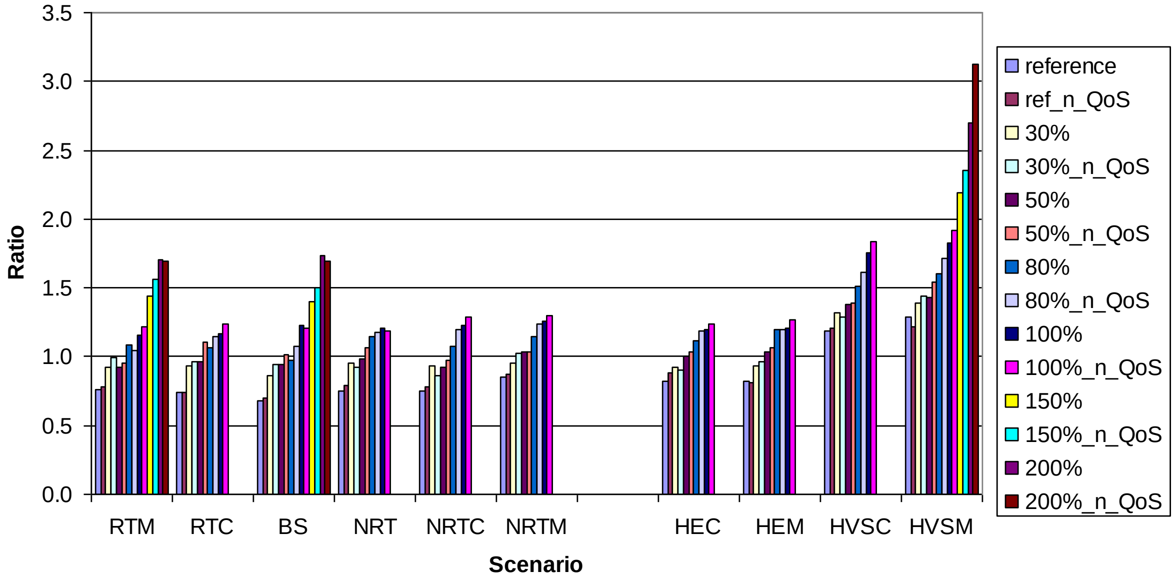


Figure 4.4. Ratio between the average and the standard deviation for the global throughput by scenario and number of users

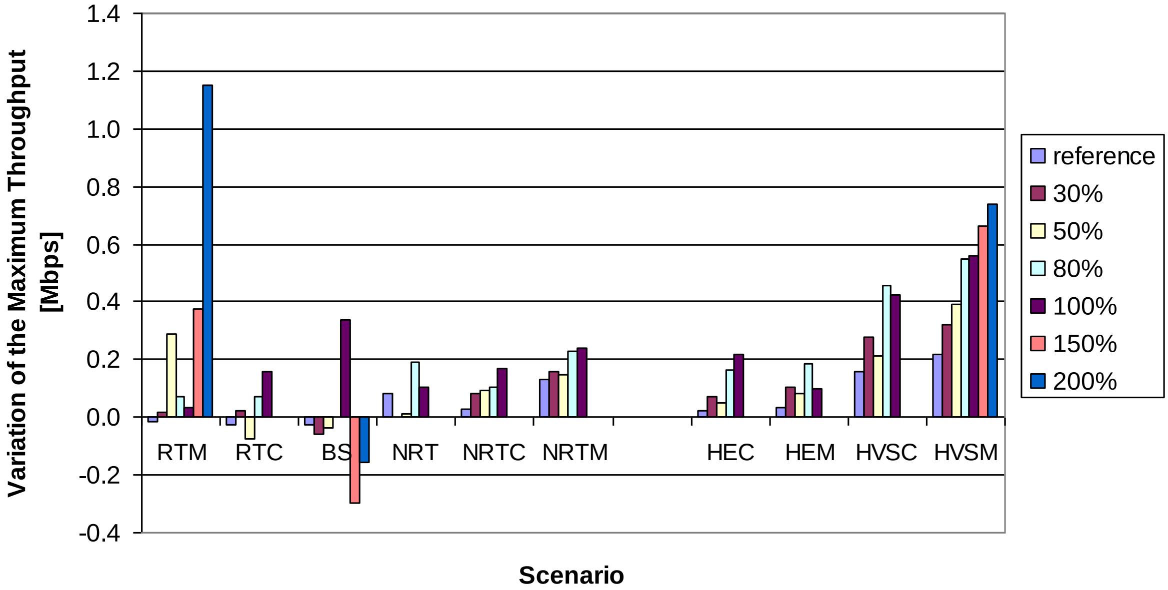


Figure 4.5. Variation of the maximum throughput by scenario and number of users when using QoS.

Regarding delay, in Figure 4.6 the delay in all scenarios with QoS mechanisms disabled is presented. As shown in Annex D, it is an end-to-end delay for all the packets received at the

MAC layer of all WLAN stations in the network. For the scenarios with up to 100 % increase in the number of users, global delay values are somewhat small in the majority of the scenarios, again with increasing values when the number of users increases, especially for the scenarios with more real-time users, where the values increase sharply, reaching almost 30 ms. When increasing even more the number of users, delay values increase in all situations reaching almost 180 ms, which are very high values, leading to severe performance degradation.

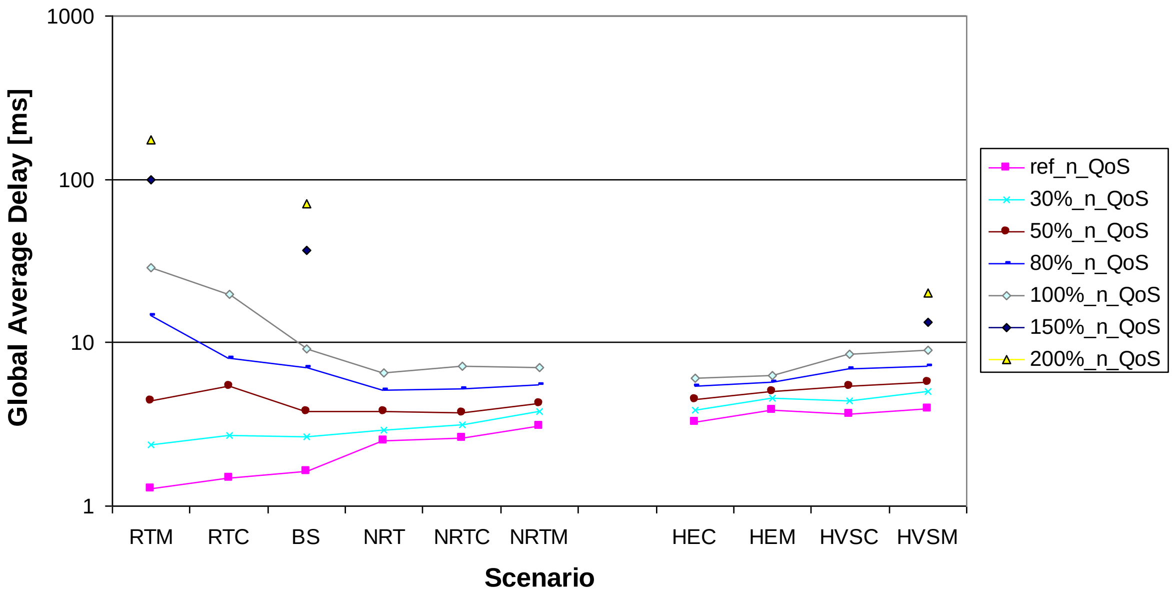


Figure 4.6. Global Delay results without QoS by scenario and number of users.

It is expected that QoS mechanisms can reduce the global overall delay, especially when there are more users with higher priority applications (real-time ones). In Figure 4.7, one shows the reduction obtained in the global average delay, in all scenarios, when QoS mechanisms are enabled, *i.e.*, how many times the delay is smaller when using QoS. As expected, QoS mechanisms decrease delay in all situations, as the reduction values are always above unity. Also, the reduction is even higher when the number of real-time users increases, with values reaching more than 7, *i.e.*, the delay values can be up to 7 times smaller when using QoS in the analysed scenarios. Again, one can see very high performance degradation for the scenarios with 150 % and 200 % increase in the number of users. The delay values without QoS are already very large, and on top of that, the delay reduction in the RTM scenario is much smaller comparing to the scenarios with less users. So, one can conclude that 100 % is the maximum increase in the number of users that the network can sustain, while still being able to maintain good performance. For this reason only the scenarios with up to 100 % increase (46 users) in the number of users are further analysed. But one must take into account that this maximum number of users depends on the scenario that is being considered. The delay is very high in only the more real-time loaded scenarios. In scenarios with fewer real-time users, still fairly good results are obtained, even when increasing the number of users by 200 % (69 users), *e.g.*, delay in the HVSM scenario is still around 10 ms, when using QoS.

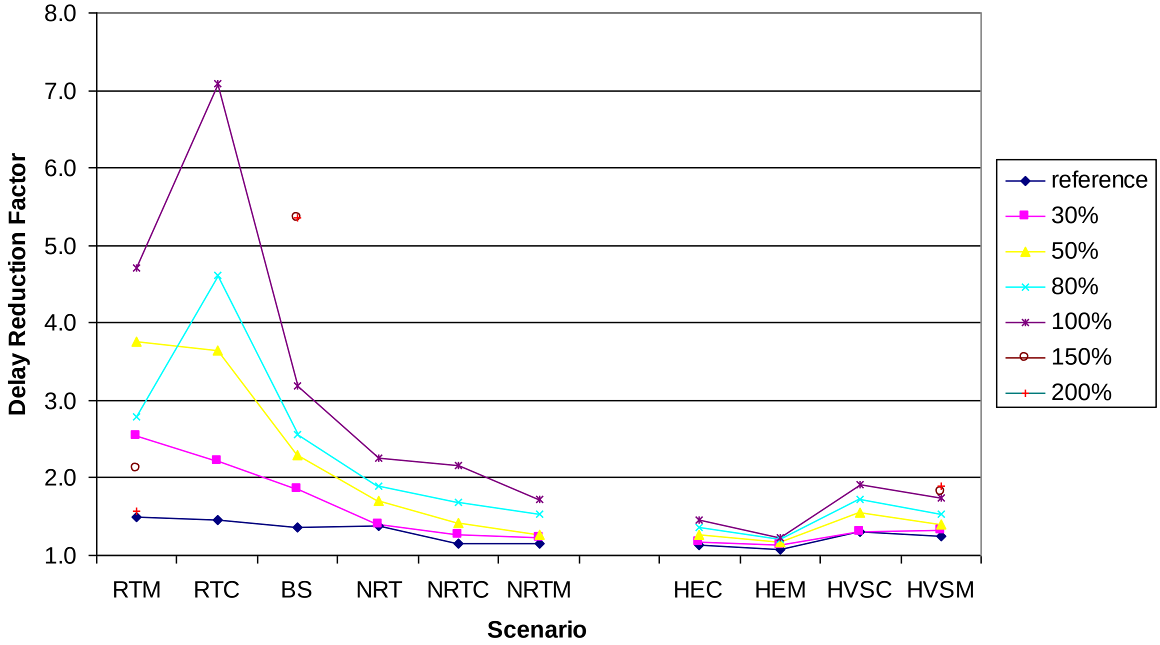


Figure 4.7. Global average delay reduction when using QoS mechanisms by scenario and number of users.

These results show that QoS mechanisms can have a great impact on network performance. Gains are observed in both global throughput and global delay. QoS mechanisms are seen to increase the maximum achievable throughput, especially in the non real-time centric scenarios, while delay gains are observed in all scenarios, but with larger variations in the real-time centric ones, for up to 100 % increase in the number of users.

But, not only the average is important; variation needs to be analysed as well. In Figure 4.8, the ratio between the average and the standard deviation of the global delay for all scenarios is presented. One can see that in almost all situations, values are higher in the scenarios with the QoS mechanisms active, meaning that in these situations not only the average delay is smaller, but also it varies less than in the scenarios without QoS mechanisms.

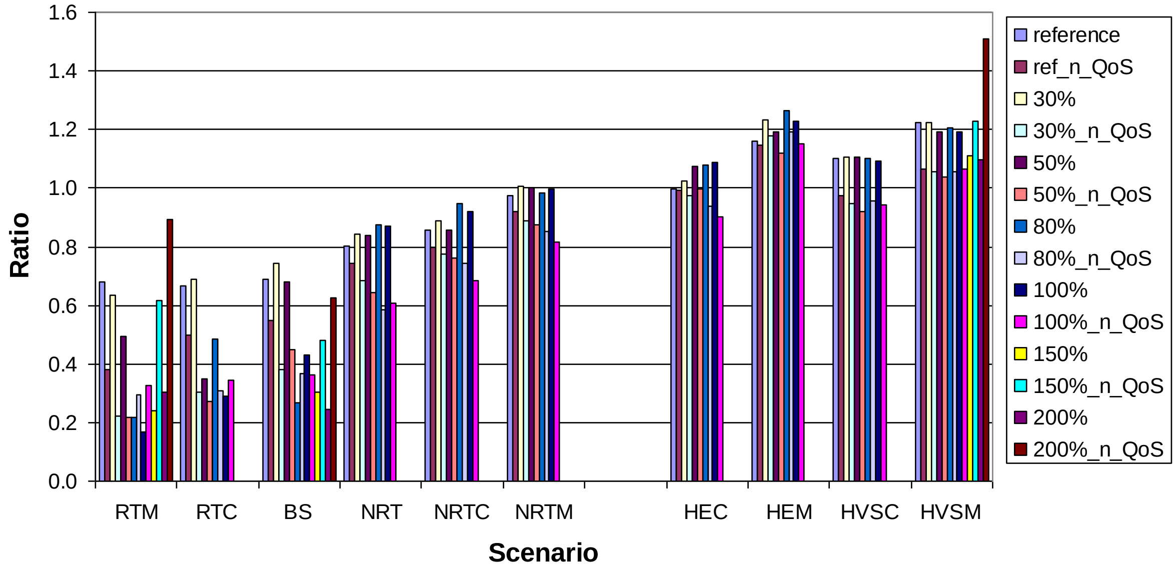


Figure 4.8. Ratio between the average and the standard deviation for the global delay by scenario and number of users.

To analyse the global behaviour of the network, one must consider not only the throughput and delay, but also the number of retransmission attempts and the dropped data, in order to check if eventual performance gains in the throughput or delay caused by QoS mechanisms are not causing undesired effects. Dropped data (by either buffer overflow or retry threshold exceeded) leads to performance losses, thus, increasing the number of retransmission attempts.

In Figure 4.9, dropped data values for all profiles with up to 100 % increase in the number of users is presented. In the scenarios with fewer users, the values are residual in all situations, but when the number of users increases, one sees an increase in dropped data, especially for the most resource demanding scenarios. Again, it can be seen that the QoS enabled scenarios have a much better performance with values always below 1 kbps, while the QoS disabled scenarios almost reach 9 kbps. One should notice that these values, similarly to the global throughput and global delay ones, just take MAC layer issues into account, *i.e.*, eventual retransmissions made/caused by the higher layers, *e.g.*, application layer retransmissions are not considered in these values.

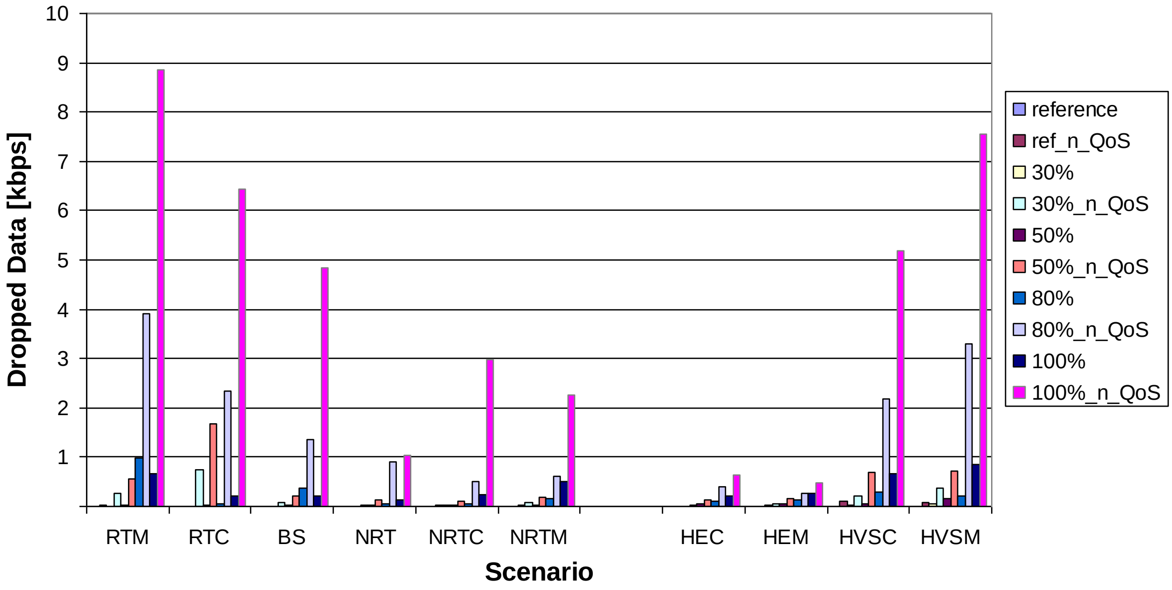


Figure 4.9. Dropped Data (buffer overflow + retry threshold exceeded) by scenario and number of users.

The results shown so far point out that QoS mechanisms can improve the overall performance of the network, enabling smaller global delays, while still being able to achieve higher maximum throughputs, at least in some situations. QoS mechanisms aim at improving the priority of real and near real-time applications, hence, as expected, larger gains are observed in real-time centric/maximum scenarios. But all studied profiles show some improvement, meaning that QoS mechanisms can improve network performance, even when network traffic is mainly non real-time one. These improvements are more tangible when the network load increases, where better results are achieved.

One might think that these improvements may cause undesired effects, like an increase of signalling and control traffic, but Figure 4.10 shows that, in any given scenario, the ratio between the signalling and control traffic and data one remains almost constant, with or without QoS, even when varying the number of users. Therefore, one can conclude that QoS mechanisms do not impose a significative increment in network signalling and control traffic. It is also possible to see that as real-time traffic increases so does the percentage of signalling and control, because real-time traffic has a much more symmetric behaviour than the non real-time one.

Finally, one must also look individually at each of the applications, in order to analyse the effects of QoS mechanisms. Generally, it is shown that global results improve, but it is also necessary to quantify, by application, the effect of using QoS, and to check if eventual gains in some applications are not surpassed by larger losses in others.

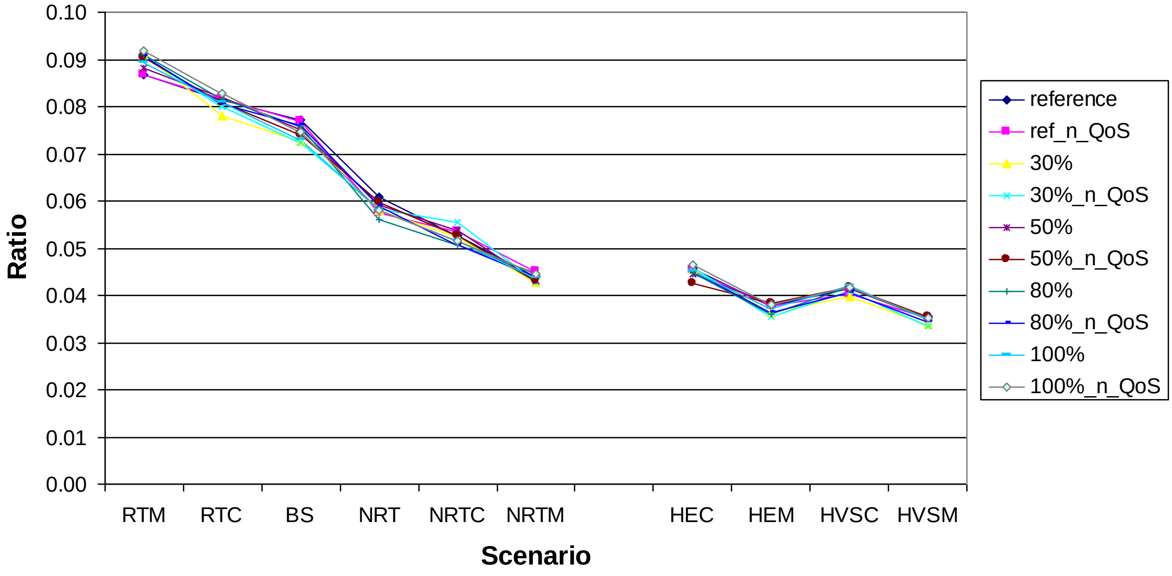


Figure 4.10. Ratio between AP Control Traffic and Data Traffic by scenario and number of users.

Each application, especially the real-time or near real-time ones, has some parameters that allow the analysis of its behaviour. For instance, for the characterisation of VoIP applications, one can look at the delay, jitter, or number of dropped packets. As previously seen, globally, both the number of dropped packets and delay are very small. Regarding delay, it is difficult to compare it among applications, because different applications behave differently, due to their different profiles. Also, within Modeler, the delay values that one can measure in each application represent different things, *i.e.*, in the Voice and Video applications one can measure the end-to-end delay, while, for instance, in HTTP one can measure the object or the page download response times. These are different things that cannot be compared directly, so a different approach was taken.

What is important to study is how better applications behave when QoS mechanisms are enabled. This way one can compare directly the results from the different applications. The variation of delay was calculated for the three most representative scenarios (RTM, BS and HVSM) using four applications, each one representing the ACs: VoIP (end-to-end delay), Video (end-to-end delay including both Video Streaming and Video Conference applications together), HTTP (object response time) and Email (download response time), Figure 4.11. This figure shows how better or worse the delay is, for each scenario and for each of the selected applications, when using QoS mechanisms, *e.g.*, a 20 % value means that the delay when using QoS mechanisms is 20 % smaller.

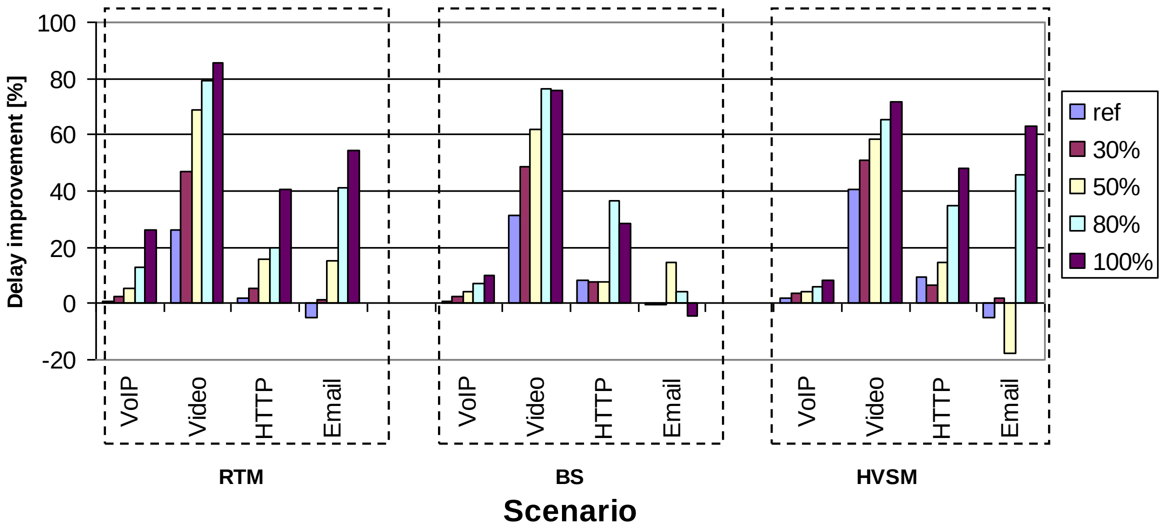


Figure 4.11. Delay improvement by using the QoS mechanisms in three different scenarios.

It would be expected that the applications with higher priorities would have better results than the less priority ones. Video applications are the ones who have a larger performance increase when using QoS mechanisms, for all scenarios, with delay improvement reaching more than 80 % in some scenarios. For VoIP, delay improvement is not so high, which can be explained by the fact that it considers encoding and decoding delays, which are not considered in the Video one, limiting the minimum value that delay can achieve. Also, one can see a clear trend in the increase of the gain values as the number of users increases. For the remaining applications, the overall trend is the same, *i.e.*, one can seen an improvement in almost all situations, with delay improvement increasing as the number of users increases. Hence, QoS mechanisms perform better for higher network loads, and gains can be observed even for lower priority applications.

It was expected, and it was confirmed, that the performance of QoS mechanisms improves when the network load is higher, but even so, very substantial gains can be seen in almost all scenarios, even when increasing the number of users to the double of the reference scenario (100 % increase). After that, the network load starts to be so high that all performance metrics start to show a much degraded performance, with packet end-to-end delay values reaching more than 100 ms, even while using QoS mechanisms. Even so, realistic traffic profiles were used in the simulations, and so, the observed gains are the ones that one may expect to see in real networks.

Looking just at the scenarios with the number of users up to 100 % increase, very good results are observed when using QoS mechanisms, with improvements seen in all scenarios and with increasing number of users. All these results show that QoS mechanisms can improve the overall network performance. Concerning the global throughput, one can see an increase in the maximum achievable throughput in the non real-time centric/maximum scenarios, with QoS scenarios reaching values of around 600 kbps more than the non-QoS ones, reaching almost 5.8 Mbps. Regarding the global packet end-to-end delay, one can also see some positive effects of QoS mechanisms, as delay decreases (up to 7 times smaller when using QoS, with 6 ms maximum average delay) and becomes more stable when using QoS mechanisms. Finally, analysing individually the applications, all of them see benefits in delay by using QoS. For instance, for the Video applications, an improvement of up to 85% is seen in delay, also with improvements in almost all scenarios for all applications.

4.3. Results from the System Parameters Variation

For the variation of system parameters, more scenarios were defined. The four considered system parameters were varied, which led to the definition of 16 basic scenarios. Then, these scenarios were simulated using 3 different application profiles (the RTM, BS and HVSM scenarios analysed in the previous section) varying also 3 times the number of users in each profile/scenario (reference, 50 % and 100 % increases), which amounts to a total of 144 different scenarios, leading to the execution of 1 440 new simulations. These simulations took about full 30 days to complete, using the same computer previously stated, and occupying around 3 GB.

The main objective of all simulations performed in this section is to evaluate the influence of system parameters variation on global network performance, and to verify if better performance can be achieved by tuning up these parameters. So, this way, the results obtained in all scenarios are compared directly with the ones obtained by using the default parameters.

Starting by the global results, one can observe the average delay for all profiles and scenarios, Figure 4.12. One can see that the variation of system parameters has only tangible effects when the number of users increases 100 %, compared to the reference value, and, mainly in the profiles with more users with real-time applications (RTM and BS). The Default scenario is defined as the one where default system parameters values are used, for all profiles studied. Using the "RTM – 100 %" profile as reference, one can compare all new scenarios with the Default one, where the delay is around 6 ms. Some slight improvements can be seen on some of them, with the lowest delay being below 3 ms, but also some serious performance degradation is seen on others, with delay reaching almost 30 ms in the worst situations.

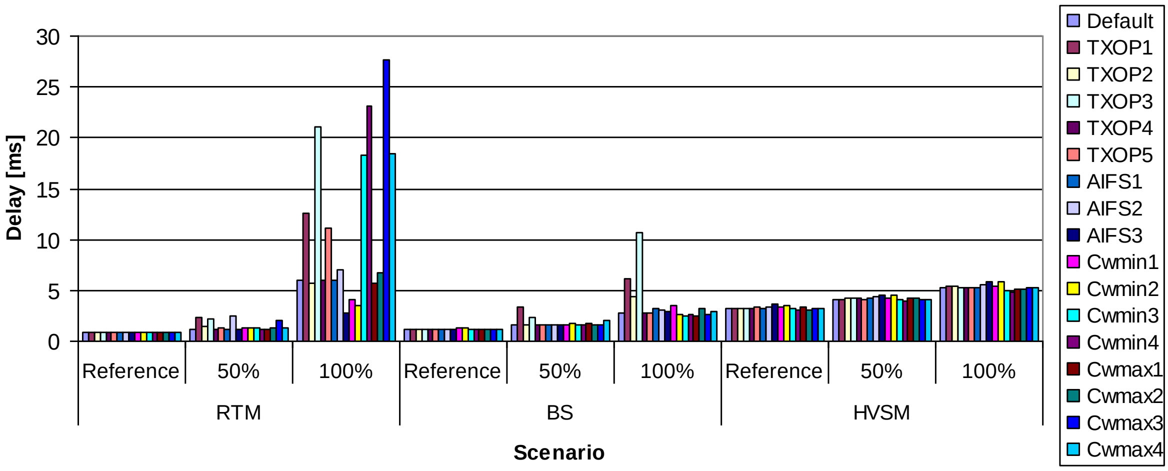


Figure 4.12. Global Delay by scenario.

Concerning dropped data, Figure 4.13, results are very similar to the delay ones, with only the more loaded scenarios showing some differences on the results as a consequence of system parameters variation, but in this situation, default scenarios are, almost always, the ones with best performance, although dropped data values are quite small in all analysed situations. For the "RTM – 100 %" profile, one can see variations between 0.3 and 3.7 kbps, when the Default scenario value is around 0.6 kbps.

Additionally, one must also consider individually each one of the system parameters, in order to analyse their effect on network performance, both globally and by application. Starting by the TXOP parameter, it allows a given station, after gaining access to the medium, to transmit sequentially a burst of packets during a certain amount of time (the TXOP duration), without contending for the medium between packet transmissions. This allows the network to achieve higher throughput and also lower delay, when a given application produces packets in a more or less constant way, *e.g.*, Voice or Video traffic. In the TXOP variation scenarios, the TXOP1, TXOP2 and TXOP3 scenarios are used to analyse the impact of TXOP size in the higher priority applications, while the TXOP4 and TXOP5 aim at analysing the impact of using TXOPs in the lower priority ones.

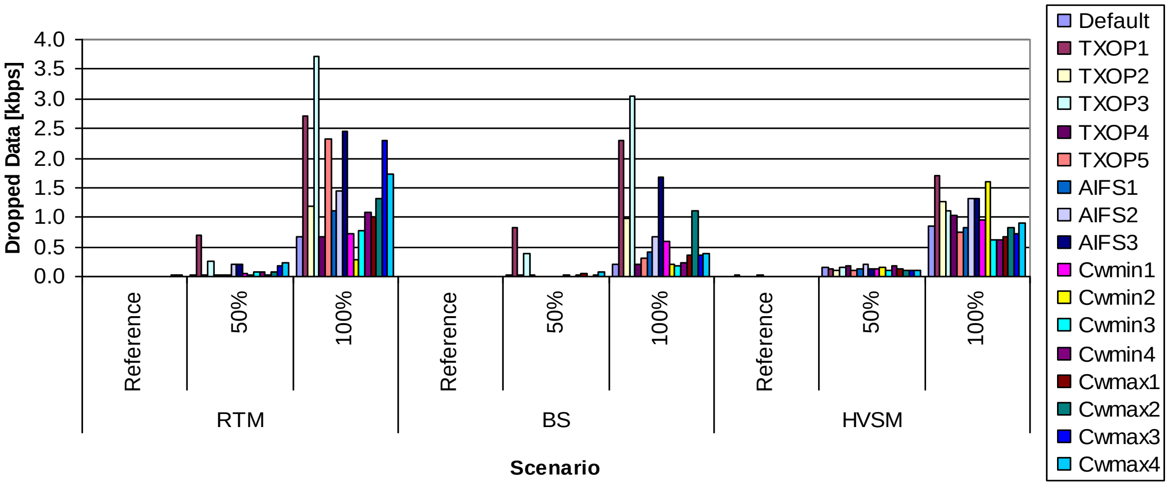


Figure 4.13. Dropped Data (buffer overflow + retry threshold exceeded) by scenario.

The maximum throughput results in the TXOP scenarios, Figure 4.14, show some mixed results, with very bad results in all scenarios for the "RTM – 100 %" profile, as values decrease almost 600 kbps comparing to the Default one. In contrast, there are several profiles/scenarios where there are some increases, albeit smaller, reaching 130 kbps more than the Default one. On the other hand, regarding delay, Figure 4.15, one can observe that results are very bad in all scenarios/profiles, as it increases in almost all situations, especially in the same "RTM – 100 %" profile, where delay is around 15 ms higher than in the Default scenario.

The first three TXOP scenarios were defined to analyse the influence of TXOP duration variation in the two higher priority ACs, but results are negative with decreasing performance in most of the profiles, especially when the number of users increases, and for those using real-time applications. The two other TXOP scenarios are intended to analyse if better performance can be achieved by allocating TXOPs to the two lower priority ACs, as no TXOP is allocated to these ones in the default situation. The TXOP4 scenario results show that the influence of using TXOPs in the AC(1) category (best effort traffic) is negligible, as results are quite similar to the Default scenario. Regarding the TXOP5 scenario, the results obtained are again not very good, being similar to the first three scenarios, with a large performance drop in the "RTM - 100 %" profile, even if the other profiles show some improvement in the maximum throughput.

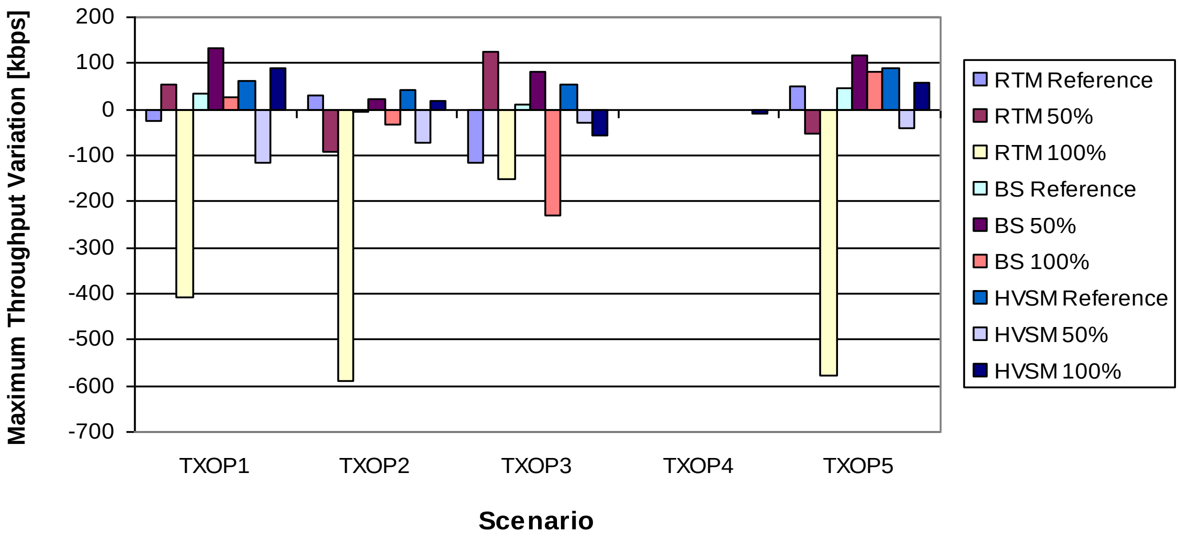


Figure 4.14. Maximum Throughput variation by TXOP scenario.

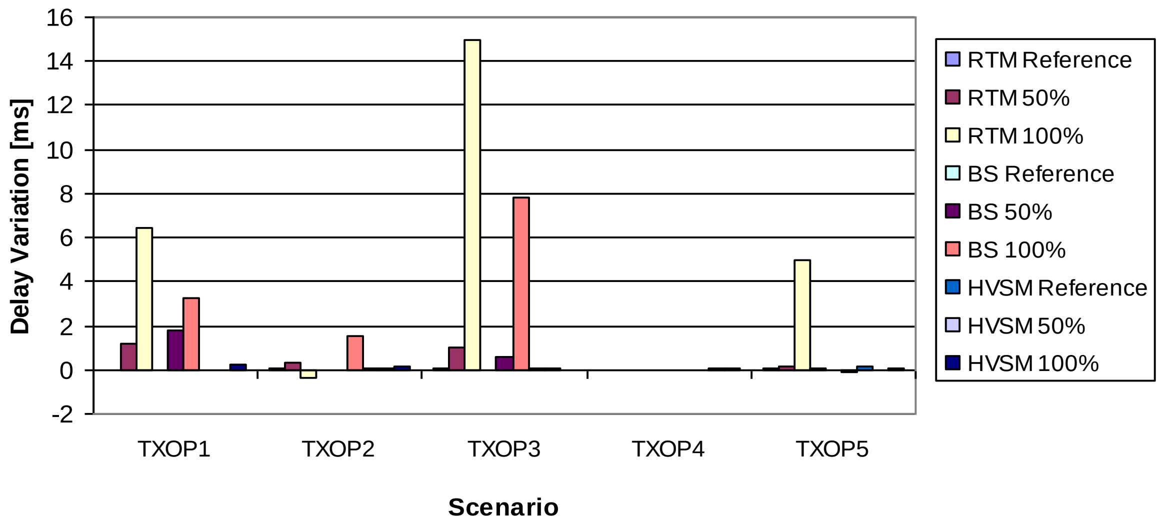


Figure 4.15. Delay variation by TXOP scenario.

As far as the CW variation profiles are concerned, one expects that these two parameters have an impact mainly in delay (more or less time waiting before accessing the channel). CW values have an impact whenever there are collisions between transmissions; a smaller CW_{min} value makes retransmissions faster, but in turn, increases the number of collisions, while the increase in CW_{max} can help to reduce the number of collisions, but increases delay. The defined scenarios are quite similar in both parameters, as the two first ones in each of the parameters (CWmin1, CWmin2, CWmax1 and, CWmax2) increase the values of the lower priority ACs, thus, increasing the priority of the higher priority ones, while the latter ones decrease the values of all ACs, trying to see if the overall performance can be improved.

In Figure 4.16, one presents the Media Access Delay improvement (representing the total of queuing and contention delays, as defined in Annex D) for the CW_{min} and CW_{max} scenarios, very mixed results in all scenarios being observed. The CWmax1 and CWmax2 scenarios have almost no influence in these results, but the CWmin1 and CWmin2 ones show some good results for the "RTM – 100 %" profile, with delay improvements reaching 45 %. For the remaining profiles, results are the opposite, with increasing delay in almost all situations, even if the variation is small. For the CWmin3, CWmin4, CWmax3, and CWmax4 scenarios, results are very disappointing, as they have a huge impact in the "RTM – 100 %" profile, with delay values increasing more than 300 %. So, one may conclude that the CW_{min} and CW_{max} variation can in fact improve delay, in some situations, but in turn, it can lead to severe performance degradation in some other situations, depending on the number of users and on application profiles.

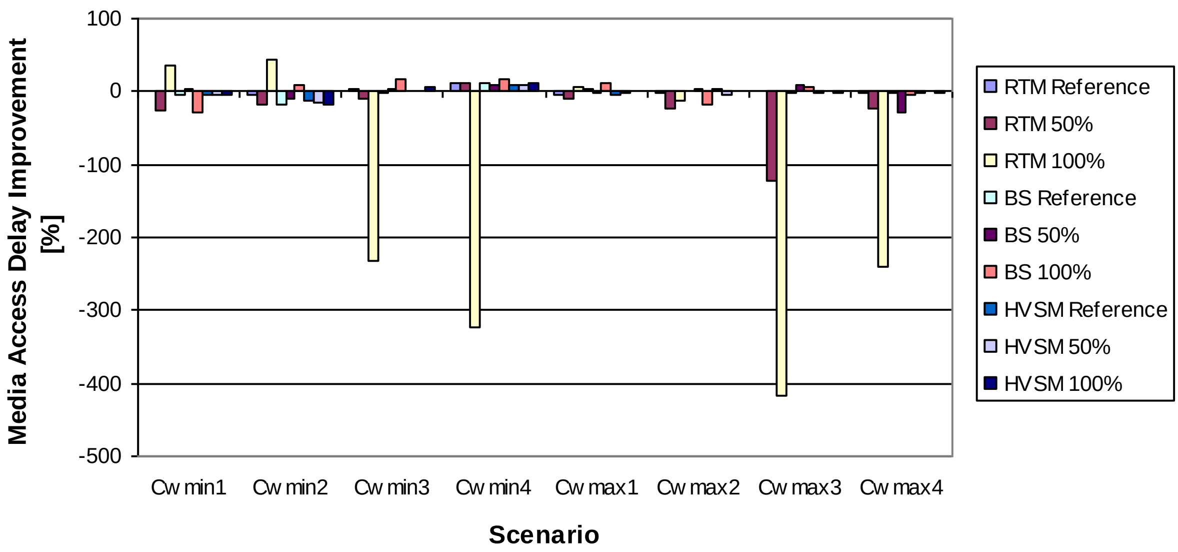


Figure 4.16. Media Access Delay improvement by CW_{min} and CW_{max} scenarios.

Finally, one takes a look at the AIFSN variation scenarios. AIFSN is the parameter with the most direct influence on the minimum time that each one of the applications must wait before accessing the wireless medium; any given station must always wait the AIFSN duration before transmitting, even if the wireless medium is sensed to be idle. Three more scenarios were defined, increasing the values of the lower priority ACs, thus, granting more priority to the real-time ones.

The changes in these scenarios are not very large, and so it was expected that their influence on the results would be rather small, as the AIFSN values are linked to time-slot duration, which is just a few microseconds. But, as it can be seen, Figure 4.17, AIFS may have a great impact in the delay experienced by all applications. In the first two scenarios (AIFSN1 and AIFSN2), there is just an increase of the corresponding values of the two lower priority ACs, and results show little impact on the overall performance of all ACs; but, the third scenario (AIFSN3) shows an interesting result: the AIFSN value for AC(2) (Video traffic) is increased from 2 to 5, which is a very small duration, as this value represents the number of time-slots that are added to the SIFS value before starting the contention duration. This 3 time-slots difference does not seem much, but, when looking at the results obtained for the AC(2), one can see that the results are catastrophic as the delay increases almost 1 000%. This leads in turn to some good results in the AC(3) results (Voice traffic), with a 70 % improvement in delay for this AC. Hence, one can conclude that this parameter definition would only make sense in a network without or with very little real-time video, as the performance degradation for the video applications is too high to balance the gains in the voice application (AC(3)).

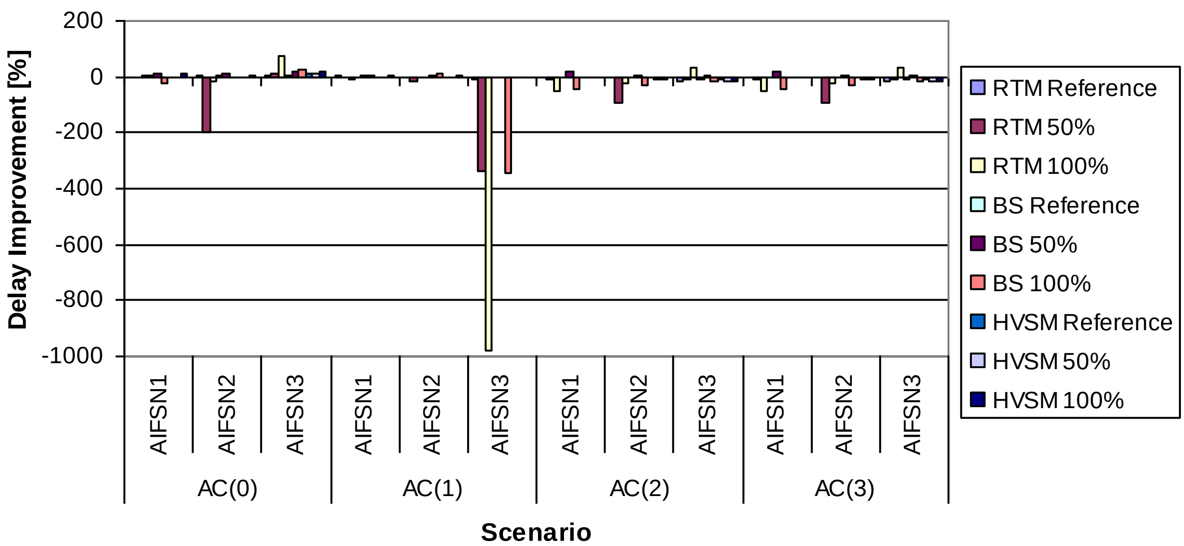


Figure 4.17. Delay improvement by AIFSN scenario and by AC.

The scenarios analysed in this section looked at the effects of changing the system parameters values. Several scenarios were defined with variation in all these parameters (TXOP duration, AIFSN, CW_{min} and, CW_{max}) in order to access their influence on the global network performance, and individually by AC. These scenarios were simulated using the three most representative scenarios defined earlier (RTM, BS and HVSM), with three variations on the number of users (reference, 50 % and 100 % increase).

For the scenarios with a smaller number of users, there is almost no difference in the results in all of them, differences being only tangible for scenarios with 100 % increase in the number of users, and especially for the RTM and BS profiles.

The comparison between the results obtained in these scenarios and the default ones has shown that some improvements can be made in some particular situations (70% delay improvement for the voice application in AIFSN3 scenario). But, in some scenarios, these performance gains lead to performance drops in other applications or profiles (1 000 % delay increase in the same AIFSN3 scenario for the video applications).

The main conclusion is that the default system parameters are a good “all round” balance between all applications and scenarios, even when varying the number of users and the application profiles. It is seen that they provide a good performance balance for all studied profiles. On the other hand, all analysed scenarios allow one to see that one might manage to improve network performance by tuning up system parameters, increasing performance for a given application or under a given profile, but this being only possible in specific situations. This way, WLANs can be tuned up, according to their location and typical usage profile, *e.g.*, one can boost VoIP application performance if there are no real-time video users present. System parameters values can be changed whenever needed, as they are chosen and distributed to the stations by the AP, which gives a lot of flexibility to the network management.

Based on all these observations, one may add some conclusions regarding the setting up of system parameters. This selection is highly dependent on the usage profile, and should be changed in order to have a better network performance for varying usage profiles. For instance, when throughput is more important than delay (*e.g.*, when FTP is mostly used), one can increase the TXOP duration, especially for the lower priority applications (TXOP5 scenario), which increases global throughput. On the other hand, when delay is more important (*e.g.*, when VoIP is mostly used), the option is to make changes in the CW_{min} values, increasing them in the lower priority ACs (CWmin1 and CWmin2 scenarios). All these conclusions are valid only if there is a mix of traffic in the network, *i.e.*, if there is traffic in all ACs. When this does not occur, one can decrease the priority of the less significant services, by changing the AIFSN parameter, *e.g.*, VoIP service performance can be greatly improved if no real-time video is present (AIFSN3 scenario).

Chapter 5

Conclusions

This chapter finalises this work, drawing some conclusions, with some ideas for future work being pointed out as well.

In this thesis, a complete and comprehensive analysis of QoS mechanisms in WLANs is performed. This analysis is made for multi-service, multi-user scenarios, in order to emulate real networks, which enables one to have a good know-how about QoS mechanisms performance in WLANs.

WLANs in general, was well as services and applications classification and characterisation, and also the basic principles of the IEEE 802.11e standard for QoS provision, are studied in the first part of the work. Then, one takes a look at the state of the art on QoS mechanisms in WLANs and on the algorithms that can be used for such. Each of the system parameters influence is analysed in detail, and also their expected effects in network performance.

A simulator was developed based on the OPNET Modeler simulation engine and its wireless module. This wireless module includes all nodes needed to implement any given 802.11 network, including the majority of the 802.11 new features: complete EDCA mechanism, with all the available ACK mechanisms as well. HCCA is not available in Modeler, hence not being studied in this work. The definition of the reference scenario was the starting point of the simulator implementation. For such, the AROMA Project, [Ljun06], "hotspot scenario" (23 users) was used, together with the defined "business profile" (6 applications). The 6 different applications available in this scenario are: VoIP, Video Conference, Video Streaming, HTTP, FTP and Email. A square scenario with 100 m (the predefined office scenario in Modeler) was used, and the distribution of the stations by this workspace was done randomly.

After the definition of the reference scenario, one defined all the simulation scenarios. The first set of simulations had the objective of analysing the effects of using QoS mechanisms in a wide range of scenarios, varying both the number of users and usage profiles. The reference "business profile" has 60 % real-time users and 40 % non real-time ones. Based on this scenario, several scenarios were defined by varying applications profiles, while still maintaining the relative weight of each application within each group. These scenarios range from the RTM one (80 % real-time and 20 % non real-time) to the NRTM one (20 % real-time and 80 % non real-time). Four more non real-time centric scenarios were defined, this time by varying applications relative weights, with more HTTP and Email users in two of these scenarios, *e.g.*, HEM scenario (10 % real-time and 90 % non real-time) and more HTTP and Video Streaming users in the other two, *e.g.*, HVSM one (also 10 % real-time and 90 % non real-time). The number of users was also increased several times, taking the reference value of 23 users. So, scenarios with 30 %, 50 %, 80 %, 100 %, 150 % and 200 % increase were also defined, but the 150 % and 200 % increase scenarios were only simulated for some of the profiles (RTM, BS and HVSM, considered the most important and representative ones) due to simulation duration issues, and because it was seen that with this number of users network performance is not very good. For all simulations, IEEE 802.11b PHY was used, with its maximum throughput of 11 Mbps.

All these variations led to the definition of 112 different scenarios, which amounts to a total of 1 120 simulations (10 simulations per scenario). Some of the scenarios are very computationally demanding and so, the total simulation time reached more than full 24 days, using a dual-core AMD Athlon X2 3800+ with 1 GB of RAM, with all this data occupying around 1.3 GB.

Results from all these simulations show that for the reference number of users, there is a slight performance increase when using QoS mechanisms, although a smaller one as the network load is still very low, and the results are quite good both with and without QoS mechanisms. As the number of users increases, up to 100 %, the performance improvement by using QoS mechanisms also increases a lot; depending on the scenario, delay improves up to a factor of seven, and the maximum throughput reaches 600 kbps more than the non-QoS ones, getting to almost 5.8 Mbps. Dropped data values show similar results, with all scenarios using QoS mechanisms showing better performance.

Results from the simulated scenarios with 150 % and 200 % increase in the number of users are found to be very bad, as even with QoS, delay is very large (more than 100 ms packet end-to-end delay in some situations), and the gains obtained by using QoS mechanisms start to decrease as well. This way, 100 % is the maximum increase in the number of users that the network can sustain, while still being able to maintain good performance. Also, it was seen that using QoS mechanisms does not increase signalling and control traffic, as the ratio between signalling and control traffic and data one remains almost constant, with and without QoS, even when varying the number of users. Finally, one can add that realistic traffic profiles were used in simulations, and so, the gains observed are the ones that one may expect to see in real networks.

Finally, analysing individually applications, all of them see benefits in delay by using QoS. For instance, for the Video applications, an improvement of up to 85% is seen in delay, also with improvements in almost all scenarios for all applications. The main conclusion for the first set of simulations is that all results show that QoS mechanisms can improve the overall network performance.

The second group of simulations had the objective of studying and analysing the impact of system parameters variations on network performance. This had the aim of improving network performance by tuning up the values of several parameters. The most important profiles defined earlier (RTM, BS and HVSM ones) were selected to be used in this second group of simulations, again with a varying number of users: reference, 50 % and 100 % increases. Several scenarios were then defined, varying all four system parameters multiple times (5 TXOP, 3 AIFSN, 4 CW_{min} and 4 CW_{max} variation scenarios), this way creating an extra 144 scenarios. Performing 10 simulations per scenario as before, led to the execution of 1 440 new simulations, which took about 30 full days to complete and occupying around 3 GB.

Results of all these simulations show that it is possible to increase the maximum throughput by varying the TXOP value (up to 130 kbps more) or decrease media access delay by varying the CW_{min} and CW_{max} values (up to 45 % improvement), at least in some usage profiles. AIFSN variation scenarios have also shown that it is possible to increase performance of some applications by tuning up system parameters accordingly.

The main conclusion from all these simulations show that the default system parameters are a good "all round" balance between all applications and scenarios, even when varying the number of users and application profiles. It is seen that they provide a good performance balance for all studied profiles. On the other hand, all analysed scenarios allow one to observe that one might manage to improve network performance by tuning up system parameters, increasing performance for a given application or under given application profiles, but this being only possible in very specific situations. This way, WLANs can be tuned up, according to their location and typical usage profile, *e.g.*, one can boost VoIP application performance if there are no real-time video users present. Also, system parameters values can be changed whenever needed, as they can be chosen and distributed to the stations by the AP, which gives a lot of flexibility to the network management.

It is expected as well that some of the new ACK policies (both Block ACK options and the No ACK one) may help on increasing performance, at least in some situations. Unfortunately, it was impossible to perform these simulations, as Modeler always crashed without apparent reason when they were running. So, it was impossible to study the impact of these new ACK policies defined in 802.11e standard.

Based on all these observations, one may add some conclusions regarding the setting up of system parameters. This selection is highly dependent on the usage profile, and should be changed in order to have a better network performance for varying usage profiles. For instance, when throughput is more important than delay (*e.g.*, when FTP is mostly used), one can increase the TXOP duration, especially for the lower priority applications (TXOP5 scenario), which increases global throughput. On the other hand, when delay is more important (*e.g.*, when VoIP is mostly used), the option is to make changes in the CW_{min} values, increasing them in the lower priority ACs (CWmin1 and CWmin2 scenarios). All these conclusions are valid only if there is a mix of traffic in the network, *i.e.*, if there is traffic in all ACs. When this does not occur, one can decrease the priority of the less significant services, by changing the AIFSN parameter, *e.g.*, VoIP service performance can be greatly improved if no real-time video is present (AIFSN3 scenario).

Regarding future work, one might suggest some additions to this study:

- To study the effect of using different ACK policies, as it is expected that they may have some positive performance effects.
- To include the questions related to propagation, *i.e.*, in real environments, channel characteristics vary along time, causing the network to adapt the transmission rate, this way increasing resilience against errors. Modeler does not support these channel adaptation techniques, and because of this, it always transmits using the predefined transmission rates.
- To study the coexistence of QoS and non-QoS nodes in the same network.
- To analyse the effect of interference from other WLANs or other interfering nodes in the neighbourhood.
- To include automatic adaptation of system parameters, *i.e.*, allow the AP to change system parameters on the fly, to act in response to users and profiles variations.
- To study HCCA performance.
- To analyse the provision of QoS guarantees using both HCCA and EDCA simultaneously in the same network.

Annex A

Applications Parameters

In this annex, a complete overview of applications parameters is presented. All 6 applications used in this work are presented, with all the values necessary to characterise them in Modeler.

A.1 VoIP

VoIP is one of the applications that can be implemented in Modeler, and some parameters are used to characterise it, Table A.1.

Table A.1. VoIP parameters.

Incoming Silence Length [s]	Exponential (0.456)[1]
Outgoing Silence Length [s]	Exponential (0.456)
Incoming Talk Spurt Length [s]	Exponential (0.854)
Outgoing Talk Spurt Length [s]	Exponential (0.854)
Symbolic Destination Name	Voice Destination
Encoder Scheme	G.729 A (silence)
Voice Frames per Packet	1
Type of Service	Interactive Voice (6)
RSVP Parameters	None
Traffic Mix [%]	All Discrete
Signalling	None
Compression Delay [s]	0.02
Decompression Delay [s]	0.02

All parameters presented in Table A.1 are the default ones for the "IP Telephony and Silence Suppressed" scheme defined in Modeler, except the values for the Silence and Talk Spurt Lengths (both incoming and outgoing), which are the ones defined in AROMA, [Ljun06]. The values used for the encoder scheme are the default ones for the "G.729 A (silence)" encoder,

[1] When describing statistical distributions, the values presented follow the OPNET Modeler reference:

. Exponential (average);
. Constant (value);
. Uniform (minimum value, maximum value);
. Uniform_int (minimum value, maximum value);
. Lognormal (average, standard deviation);
. Gamma (scale, shape).

Table A.2. One should note that the 8 kbps bitrate value is the maximum, as the codec does not produce bits during silence periods.

Table A.2. VoIP encoder scheme.

Codec Type	CS-ACELP
Name	G.729 A (silence)
Frame Size [ms]	10
Lookahead Size [ms]	5
DSP Processing Ratio	1.0
Coding Rate [kbps]	8
Speech Activity Detection	Enabled
Equipment Impairment Factor	Unknown
Packet Loss Robustness Factor	Default

A.2 Video Conference/Video Streaming

Both video conference and video streaming applications are defined very similarly in Modeler, with just some differences on the frame interarrival time (15 frames/s and 25 frames/s for video conference and video streaming, respectively), and on bitrates (64 kbps and 400 kbps), [PeEb02]. All the values used to model these applications are presented in Table A.3.

A.3 HTTP

The fourth application used in this work is HTTP, which can be defined in detail in Modeler. It is possible to characterise how are the typical websites that one may visit, by defining, *e.g.*, their size, the number of inline objects, and so on. The values used in this work are based on a previous work, [ChLi99], where a very detailed analysis of HTTP traffic was performed, and some typical values were proposed. As this reference is quite old, after some real website observations, it was decided to double the values related to the website size, the number of inline objects, and also their size, Table A.4 and Table A.5. Modeler also has the ability of defining the number of pages per server, and the probability of remaining in the website/server after the first page is downloaded, Table A.6. This probability value is used because the HTTP 1.1 specification defines that clients can maintain persistent connections to website servers, so, when a user is browsing between different pages, if the next fetched page is on the same server, there is no need to make a new connection to the server.

Table A.3. Video Conference/Video Streaming parameters.

	Video Conference	**Video Streaming**
Incoming Stream Interarrival Time [s]	0.0667	0.04
Outgoing Stream Interarrival Time [s]	0.0667	None
Incoming Stream Frame Size [bytes]	constant (533)	Constant (2000)
Outgoing Stream Frame Size [bytes]	Constant (533)	Constant (2000)
Symbolic Destination Name	Video Destination	Video Destination
Type of Service	Interactive Multimedia (5)	Streaming Multimedia (4)
RSVP Parameters	None	None
Traffic Mix [%]	All Discrete	All Discrete

Table A.4. HTTP application specification.

HTTP Specification	HTTP 1.1
Page Interarrival Time [s]	Exponential (39.5)
RSVP Parameters	None
Type of Service	Best Effort (0)

Table A.5. Page properties.

Object Size [byte]	**Number of Objects**	**Location**	**Back-End Custom Application**	**Object Group Name**
Lognormal (20000, 50000)	Constant (1)	HTTP Server	Not Used	Not Used
Lognormal (14400,252000)	Gamma(47.258, 0.232)	HTTP Server	Not Used	Not Used

Table A.6. Server selection parameters.

Initial Repeat Probability	Browse (equivalent to 0.6)
Pages Per Server	Exponential (10)

A.4 Email

For the Email application, one considered that the default Modeler values were not very realistic, but no good email models were found in the literature. So, a new approach was taken, by collecting the sizes of all the emails sent and received in a real network (Instituto de Telecomunicações, [IT07], Lisbon site network) on a given working day. More than 4000 emails with an aggregate size of over 420 MB were recorded, and then the average (100 kB) and standard deviation values (660 kB) were calculated and used in this work, Table A.7.

Table A.7. Email application parameters.

Send Interarrival Time [s]	Exponential (360)
Send Group Size	Uniform_int (1, 5)
Receive Interarrival Time [s]	Exponential (360)
Receive Group Size	Uniform_int (1, 5)
E-Mail Size [byte]	Lognormal (100000, 660000)
Symbolic Server Name	Email Server
Type of Service	Background (1)
RSVP Parameters	None
Back-End Custom Application	Not Used

A.5 FTP

The final application used in this work is the FTP one, which definition is quite simple. As with the Email application, the default Modeler values were found to be unrealistic, as it considers a 50% ratio between downlink and uplink, with very small file sizes. The values that were used are presented in Table A.8.

Table A.8. FTP application parameters.

Command Mix (Get/Total)	0.95
Inter-Request Time [s]	Exponential (600)
File Size [byte]	Uniform_int (100000, 5000000)
Symbolic Server Name	FTP Server
Type of Service	Best Effort (0)
RSVP Parameters	None
Back-End Custom Application	Not Used

Annex B

Applications Profiles

In this annex, a complete overview of applications usage profiles is presented. The values used for the 6 applications profiles are presented here, with all the values necessary to characterise them in Modeler.

Each of the 6 applications was defined in terms of the size of downloaded files, codecs used, bitrates, and so on. In order to fully characterise an application, one must also define usage profiles that describe how applications behave through time, *e.g.*, when they start, their repetition, or their duration. Each of the applications has its own usage profile defined, but all these values can be changed.

Some of the parameters, like the application start time or the repetition of the application profile itself, were defined equally in all profiles. Others, like the applications duration or repetition in the profile, were defined accordingly in each application, Table B.1.

Table B.1. Application profiles.

	VoIP	**Video Conference**	**Video Streaming**	**HTTP**	**Email**	**FTP**
Operation Mode	Serial (Ordered)					
Start Time [s]	Uniform (0, 120)					
Profile Repetition	None					
Start Time Offset	None					
Duration [s]	Uniform (100, 140)	Uniform (100, 140)	Uniform (100, 600)	End of Profile	End of Profile	End of Profile
Inter-repetition time [s]	Exponential (600)	Exponential (900)	Exponential (900)	-	-	-
Number of repetitions	Unlimited	Unlimited	Unlimited	-	-	-

The duration values for the HTTP, Email and FTP applications were set to the “End of Profile”, because, in these applications, the repetition of the applications is defined in the application definition itself as inter-request/interarrival time, so one does not needed to define another repetition time within the application profile. The values for the durations of the VoIP application are based on the values proposed in AROMA, [Ljun06], and the values for the Video Conference and Video Streaming were then based on these values.

Annex C

Simulation Scenarios

In this annex, the complete description of the simulation scenarios is presented.

The simulation scenarios are briefly presented in Chapter 4. In this annex, a complete overview of all simulated scenarios is presented, *i.e.*, the distribution of users by each application, Table C.1, and by the number of users, Table C.2, to Table C.8.

Table C.1. Applications user distribution (percentages).

Scenario	VoIP [%]	Video Conference [%]	Video Streaming [%]	HTTP [%]	Email [%]	FTP [%]
RTM	67	13	3	7	5	5
RTC	58	12	5	10	8	7
BS	50	10	6	13	11	10
NRT	33	7	9	19	17	15
NRTC	25	5	10	23	19	18
NRTM	17	3	12	26	22	20
HEC	17	3	8	40	25	7
HEM	7	5	10	45	30	5
HVSC	17	3	20	40	15	5
HVSM	7	3	25	45	15	5

Table C.2. Applications user distribution (number of users – Reference).

Scenario	VoIP	Video Conference	Video Streaming	HTTP	Email	FTP
RTM	15	3	1	2	1	1
RTC	13	3	1	2	2	2
BS	12	2	1	3	3	2
NRT	8	1	2	5	4	3
NRTC	6	1	2	5	5	4
NRTM	4	1	3	6	5	4
HEC	4	1	2	9	6	1
HEM	2	1	2	10	7	1
HVSC	4	1	5	9	3	1
HVSM	2	1	6	10	3	1

Table C.3. Applications user distribution (number of users – Reference + 30 %).

Scenario	VoIP	Video Conference	Video Streaming	HTTP	Email	FTP
RTM	20	4	1	2	2	1
RTC	17	4	2	3	2	2
BS	15	3	2	4	3	3
NRT	10	2	3	6	5	4
NRTC	8	1	3	7	6	5
NRTM	5	1	4	8	6	6
HEC	5	1	2	12	8	2
HEM	2	1	3	13	9	2
HVSC	5	1	6	12	4	2
HVSM	2	1	8	13	4	2

Table C.4. Applications user distribution (number of users – Reference + 50 %).

Scenario	VoIP	Video Conference	Video Streaming	HTTP	Email	FTP
RTM	23	5	1	2	2	2
RTC	21	4	2	3	3	2
BS	17	4	2	5	4	3
NRT	12	2	3	7	6	5
NRTC	9	2	3	8	7	6
NRTM	6	1	4	9	8	7
HEC	6	1	3	14	9	2
HEM	3	1	3	16	10	2
HVSC	6	1	7	14	5	2
HVSM	3	1	8	16	5	2

Table C.5. Applications user distribution (number of users – Reference + 80 %).

Scenario	VoIP	Video Conference	Video Streaming	HTTP	Email	FTP
RTM	28	5	1	3	2	2
RTC	24	5	2	4	3	3
BS	21	4	2	5	5	4
NRT	13	3	4	8	7	6
NRTC	10	2	4	10	8	7
NRTM	7	1	5	11	9	8
HEC	7	1	4	16	10	3
HEM	3	1	4	19	12	2
HVSC	7	1	8	17	6	2
HVSM	3	1	10	19	6	2

Table C.6. Applications user distribution (number of users – Reference + 100 %).

Scenario	VoIP	Video Conference	Video Streaming	HTTP	Email	FTP
RTM	31	6	1	3	3	2
RTC	27	5	2	5	4	3
BS	23	5	3	6	5	4
NRT	15	3	4	9	8	7
NRTC	12	2	5	10	9	8
NRTM	8	1	6	12	10	9
HEC	8	1	4	18	12	3
HEM	4	1	4	21	14	2
HVSC	8	1	9	19	7	2
HVSM	4	1	11	21	7	2

Table C.7. Applications user distribution (number of users – Reference + 150 %).

Scenario	VoIP	Video Conference	Video Streaming	HTTP	Email	FTP
RTM	39	7	2	4	3	3
RTC	-	-	-	-	-	-
BS	29	6	3	8	6	6
NRT	-	-	-	-	-	-
NRTC	-	-	-	-	-	-
NRTM	-	-	-	-	-	-
HEC	-	-	-	-	-	-
HEM	-	-	-	-	-	-
HVSC	-	-	-	-	-	-
HVSM	4	2	14	26	9	3

Table C.8. Applications user distribution (number of users – Reference + 200 %).

Scenario	VoIP	Video Conference	Video Streaming	HTTP	Email	FTP
RTM	46	9	2	5	4	3
RTC	-	-	-	-	-	-
BS	34	7	4	9	8	7
NRT	-	-	-	-	-	-
NRTC	-	-	-	-	-	-
NRTM	-	-	-	-	-	-
HEC	-	-	-	-	-	-
HEM	-	-	-	-	-	-
HVSC	-	-	-	-	-	-
HVSM	5	2	17	31	10	4

Annex D

Performance Metrics

In this annex, the complete description of the performance metrics used is presented.

A large number of statistics and metrics can be recorded in each simulation. In this annex, the description of the metrics recorded in the simulations performed is presented, as defined in Modeler, [OPNE07]. Both global statistics and node statistics were recorded in all simulations, all the metrics that are used in this thesis being presented here.

D.1 Global Statistics

- **Throughput:** Represents the total number of bits (in bps) forwarded from WLAN layers to higher layers in all WLAN nodes of the network.
- **Delay**: Represents the end-to-end delay (in seconds) of all the packets received by the WLAN MACs of all WLAN nodes in the network and forwarded to the higher layer. This delay includes medium access delay at the source MAC, reception of all the fragments individually, and transfer of the frames via AP, if access point functionality is enabled.
- **Dropped data (buffer overflow):** The total size of higher layer data packets (in bps) dropped by all the WLAN MACs in the network due to:

  a) full higher layer data buffer, or

  b) the size of the higher layer packet, which is greater than the maximum allowed data size defined in the IEEE 802.11 standard.

- **Dropped data (retry threshold exceeded):** Total higher layer data traffic (in bps) dropped by all WLAN MACs in the network as a result of consistently failing retransmissions. This statistic reports the number of the higher layer packets that are dropped because the MAC couldn't receive any ACKs for the (re)transmissions of those packets or their fragments, and the packets' short or long retry counts reached the MAC's short retry limit or long retry limit, respectively. If the network contains QSTAs (802.11e-capable WLAN MACs) that use Block ACK mechanism as originator for some or all of their communication with other QSTAs, then, this statistic will also include any higher layer data traffic of those MACs that is transmitted and retransmitted within blocks, not acknowledged in following Block ACKs and dropped due to reaching the transmit lifetime limit.

- **Email Download Response Time:** Time elapsed (in seconds) between sending request for emails and receiving emails from email server in the network. This time includes signalling delay for the connection setup.

- **HTTP Object Response Time:** Specifies response time (in seconds) for each inlined object from the Hypertext Markup Language (HTML) page.

- **Video Packet End-to-End Delay:** The time (in seconds) taken to send a video application packet to a destination node application layer. This statistic records data from all the nodes in the network.

- **Voice Packet End-to-End Delay:** The total voice packet delay (in seconds) called "analog-to-analog" or "mouth-to-ear" delay = network_delay + encoding_delay + decoding_delay + compression_delay + decompression_delay:

  - Network delay is the time at which the sender node gave the packet to Real-Time Transport Protocol (RTP) to the time the receiver got it from RTP.

  - Encoding delay (on the sender node) is computed from the encoder scheme.

  - Decoding delay (on the receiver node) is assumed to be equal to the encoding delay.

  - Compression and Decompression delays come from the corresponding attributes in the Voice application configuration.

  This statistic records data from all the nodes in the network.

- **Media Access Delay:** Represents the global statistic for the total of queuing and contention delays (in seconds) of the data, management, delayed Block ACK and Block ACK Request frames transmitted by all WLAN MACs in the network. For each frame, this delay is calculated as the duration from the time when it is inserted into the transmission queue, which is arrival time for higher layer data packets and creation time for all other frames types, until the time when the frame is sent to the physical layer for the first time. Hence, it also includes the period for the successful RTS/CTS exchange, if this exchange is used prior to the transmission of that frame. Similarly, it may also include multiple number of backoff periods, if the MAC is 802.11e-capable and the initial transmission of the frame is delayed due to one or more internal collisions.

- **Delay (per HCF Access Category):** Represents the end-to-end delay (in seconds) of all the data packets, collected separately for each 802.11e EDCA access category, that are successfully received by all the 802.11e-capable WLAN MACs in the network and forwarded to higher layer. For the sake of this statistic, it is assumed that a packet is received by the same access category that is used for the transmission of the packet at the source node. This assumption is needed because access categories have no role within the MAC during the reception, processing and forwarding of the packets arrived from the physical layer.

  This delay includes queuing and medium access delays at the source MAC, reception of all the fragments individually, and the relay of the frame via AP, if the source and destination MACs are non-AP MACs of the same infrastructure BSS.

D.2 Node Statistics

- **Control Traffic Received:** WLAN control (RTS, CTS or ACK) traffic received by the MAC in bps. This statistic will also include Contention Free-End (CF-End) and CF-End+CF-ACK frames received during PCF periods, if PCF operation was enabled for the BSS of this MAC. Similarly, it will record received Block ACK Request (BAR) and Block ACK frames, too, if this MAC establishes Block ACK agreements with its peers or if there are 802.11e capable MACs in the BSS that are using Block ACK mechanism for some or all of their data transmissions.

- **Control Traffic Sent:** WLAN control (RTS, CTS or ACK) traffic sent by the MAC in bps. This statistic will also include CF-End and CF-End+CF-ACK frames sent by this MAC to end the contention free periods, if this MAC is an AP and it is configured to operate also in PCF mode. Similarly, it will record transmitted BAR and Block ACK frames, too, if this is an 802.11e MACs that is using Block ACK mechanism for some or all of its data transmissions.

- **Data Traffic Received:** WLAN data traffic successfully received by the MAC from the physical layer in bps. This statistic includes all data traffic received regardless of the destination of the received frames. While computing the size of the received packets for this statistic, the physical layer and MAC headers of the packet are also included. This statistic also includes Data-Null, CF-ACK, CF-Poll and CF-Poll+CF-ACK frames, which are specified as data frames in the IEEE 802.11 standard, received during the contention free periods, if PCF operation was enabled for the BSS of this MAC.

- **Data Traffic Sent:** WLAN data traffic transmitted by the MAC in bps. While computing the size of the transmitted packets for this statistic, the physical layer and MAC headers of the packet are also included. This statistic also includes Data-Null, CF-ACK, CF-Poll and CF-Poll+CF-ACK frames, which are specified as data frames in the IEEE 802.11 standard, sent during the contention free periods, if PCF operation was enabled for the BSS of this MAC.

References

[3GPP06a] 3GPP, *Services and service capabilities*, Report TS 22.105, V8.1.0, Sep. 2006 (http://www.3gpp.org).

[3GPP06b] 3GPP, *Quality of Service (QoS) concept and architecture*, Report TS 23.107, V6.4.0, Mar. 2006 (http://www.3gpp.org).

[ANAC07] http://www.anacom.pt, Nov. 2007.

[AnNT03] Ansel,P., Ni,Q. and Turletti,T., "FHCF: A Fair Scheduling Scheme for 802.11e WLAN", *INRIA – Research Report n. 4883*, INRIA Sophia-Antipolis Research Unit, Sophia-Antipolis, France, July 2003.

[AROM07] http://www.aroma-ist.upc.edu/, Nov. 2007.

[Caei04] Caeiro, M., *Quality of Service Support in IEEE802.11 wireless local area networks* (in Portuguese), Master Thesis, IST-TUL, Lisbon, Portugal, June 2004.

[ChLi99] Choi,H. and Limb,J, "A Behavioural Model of Web Traffic", in *Proc. of ICNP99 – 7th International Conference on Network Protocols*, Toronto, Canada, Nov. 1999.

[CPSM03] Choi,S., Prado,J., Shankar,S. and Mangold,S., "IEEE 802.11e Contention-Based Channel Access (EDCF) Performance Evaluation", in *Proc. of ICC'03 – IEEE International Conference on Communications,* Anchorage, Alaska, USA, May 2003.

[FeSC02] Ferreira,L., Serrador,A. and Correia,L.M., *Characterisation parameters for UMTS Services and Applications,* IST-MOMENTUM Project, Internal Report IST-TUL_WP1_DR_INT_025_WL_04, IST-TUL, Lisbon, Portugal, Mar. 2002.

[GaCN05] Gao,D., Cai,J. and Ngan, K., "Admission Control in IEEE 802.11e Wireless LANs", *IEEE Network*, Vol. 19, No. 4, July-Aug. 2005, pp. 6-13.

[GaCZ05] Gao,D., Cai,J. and Zhang,L., "Physical Rate Based Admission Control for HCCA in IEEE 802.11e WLANs", in *Proc. of AINA'05 – 19th International Conference on Advanced Information Networking and Applications*, Taipei, Taiwan, Mar. 2005.

[GrMN03] Grilo,A., Macedo,M. and Nunes,M., "A Scheduling Algorithm for QoS Support in IEEE 802.11e Networks", *IEEE Wireless Communications*, Vol. 10, No. 3, June 2003, pp. 36-43.

[GuMB06] Guchhait,A., Muralidhar,R. and Bakre,A., "Performance of Real-Time Traffic in EDCA-Based IEEE 802.11b/g WLANs", in *Proc. of COMSWARE 2006 – First International Conference on Communication System Software and Middleware*, New Dehli, India, Jan. 2006.

[GuZh03] Gu,D. and Zhang,J., "QoS Enhancement in IEEE 802.11 Wireless Local Area Networks", *IEEE Communications Magazine*, Vol. 41, No. 6, June 2003, pp. 120-124.

[IEEE03] IEEE 802.11 WG, *"Information technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications"*, ANSI/IEEE Std 802.11, 1999 Edition (R2003), June 2003 (http://www.ieee.org).

[IEEE04] IEEE 802.1 WG, *"802.1D – IEEE Standard for Local and metropolitan area networks. Media Access Control (MAC) Bridges"*, IEEE Std 802.1D – 2004, June 2004 (http://www.ieee.org).

[IEEE05] IEEE 802.11 WG, *"IEEE Standard for Information Technology – Telecommunications and information exchange between systems – Local and metropolitan area networks – specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications. Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements"*, IEEE Std 802.11e – 2005, Sep. 2005 (http://www.ieee.org).

[IEEE08] http://grouper.ieee.org/groups/802/11/, April. 2008.

[IT07] Instituto de Telecomunicações, http://www.it.pt, May 2007.

[Ljun06] Ljung, R. (ed.), *Target Scenarios Specification: vision at project stage 1*, IST-AROMA Project, Deliverable D05, Telia-Sonera, Malmo, Sweden, Mar. 2006.

[MCMK02] Mangold,S., Choi,S., May,P., Klein,O., Hiertz,G. and Stibor,L., "IEEE 802.11e Wireless LAN for Quality of Service", in *Proc. of EW2002 – European Wireless 2002*, Florence, Italy, Feb. 2002.

[MQTB04] Malli,M., Qiang,N., Turletti,T. and Barakat,C., "Adaptive fair channel allocation for QoS enhancement in IEEE 802.11 wireless LANs", in *Proc. of ICC'04 –IEEE International Conference on Communications*, Paris, France, June 2004.

[Ni05] Ni,Q., "Performance Analysis and Enhancements for IEEE 802.11e Wireless Networks", *IEEE Network*, Vol. 19, No. 4, July-Aug. 2005, pp. 21-27.

[OPNE07] *OPNET Modeler 12.0 PL5 Documentation*, OPNET Technologies Inc., Bethesda, Maryland, USA, 2007 (http://www.opnet.com).

[PeEb02] Pereira,F. and Ebrahimi,T. (eds.), *The MPEG-4 Book*, IMSC Press, Upper Saddle River, New Jersey, USA, 2002.

[PrSh04] Prado,J. and Shankar,S., "Impact of Frame Size, Number of Stations and Mobility on the Throughput Performance of IEEE 802.11e", in *Proc. of WCNC 2004 – IEEE Wireless Communications and Networking Conference*, Atlanta, Georgia, USA, Mar. 2004.

[RaPD04] Ramos,N., Panigrahi,D. and Dey,S., "Dynamic Adaptation Policies to Improve Quality of Service of Multimedia Applications in WLAN Networks", in *Proc. of BroadWIM 2004 – First International Workshop on Broadband Wireless Multimedia*, San Jose, California, USA, Oct. 2004.

[RaPD05] Ramos,N., Panigrahi,D. and Dey,S., "Quality of Service Provisioning in 802.11e Networks: Challenges, Approaches and Future Directions", *IEEE Network*, Vol. 19, No. 4, July-Aug. 2005, pp. 14-20.

[ROLE03] Roshan,P. and Leary,J., *802.11 Wireless LAN Fundamentals*, Cisco Press, Indianapolis, Indiana, USA, Dec. 2003.

[SKYP07] http://www.skype.com, Nov. 2007.

[STAL05] Stallings,W., *Wireless Communications* & *Networks – Second Edition*, Pearson Prentice Hall, Upper Saddle River, New Jersey, USA, 2005.

[WIKI06] http://en.wikipedia.org/wiki/WLAN, Oct. 2006.

[YOUT07] http://www.youtube.com, Nov. 2007.

[ZLCP04] Zhu,H., Li,M., Chlamtac,I. and Prabhakaran,B., "A Survey of Quality of Service IEEE 802.11 networks", *IEEE Wireless Communications*, Vol. 11, No. 4, Aug. 2004, pp. 6-14.