



**QOS SUPPORT IN SATELLITE AND WIRELESS NETWORKS**  
**STUDY UNDER THE NETWORK SIMULATOR (NS-2)**

Memòria del Treball Final de Carrera  
d'Enginyeria de Telecomunicació,  
realitzat per  
Albert Bel i Pereira  
i dirigit per  
María Ángeles Vázquez Castro i  
Giovanni Giambene  
Bellaterra, 12 de setembre de 2007







El sotasignat, Maria Ángeles Vázquez Castro

Professor/a de l'Escola Tècnica Superior d'Enginyeria de la UAB,

**CERTIFICA:**

Que el treball a què correspon aquesta memòria ha estat realitzat sota la seva direcció per n' Albert Bel i Pereira

I per tal que consti firma la present.

Signat: .....

Bellaterra, 3 de setembre de 2007



*I would like to thank you to  
my family and friends,  
Maryan and Giovanni,  
and my family of Siena  
Pepa, Joan Enric and Alba*





# Index

<b>1. Introduction.....</b>	<b>13</b>
<b>2. Satellite part.....</b>	<b>17</b>
2.1. Introduction.....	18
2.2. QoS requirements on the Application layer.....	19
2.2.1. Conversational services.....	19
2.2.2. Interactive services.....	23
2.2.3 Streaming services.....	24
2.2.4. Background services - applications.....	25
2.3. Transport Layer.....	26
2.3.1. TCP protocol: mechanisms and versions.....	27
2.4. QoS Architecture for IP networks.....	28
2.4.1. QoS Network building blocks.....	30
2.5. Proposed QoS support techniques on MAC layer.....	31
2.6. Relation between QoS Traffic Classes and Capacity Categories.....	34
<b>3. WiFi part.....</b>	<b>37</b>
3.1. Introduction to 802.11.....	38
3.1.1. 802.11 lacks.....	43
3.2. The 802.11e, QoS support mechanisms.....	44
3.2.1. HCF Contention-Based Medium Access (EDCA).....	45
3.2.2. Hybrid Coordination Function, Controlled Medium Access (HCCA)...	49
3.2.3. Another improvements.....	52
3.2.3.1. Block Acknowledgment.....	52
3.2.3.2. Direct Link Protocol.....	52

<b>4. The <i>Network Simulator</i></b> .....	53
<b>4.1. Satellite networks in ns-2</b> .....	56
<b>4.1.1. Geostationary satellites</b> .....	56
<b>4.1.2. Satellite nodes</b> .....	57
<b>4.1.3. Satellite links</b> .....	57
<b>4.1.4. Interconnection of wired and wireless nodes</b> .....	59
<b>4.2. Mobile networks in ns-2</b> .....	59
 <b>5. Simulation results</b> .....	63
<b>5.1. Satellite scenario</b> .....	64
<b>5.2. Wireless simulations</b> .....	70
 <b>6. Conclusions</b> .....	75
 <b>7. Bibliography</b> .....	77







# 1. Introduction

Nowadays we are involved in a society in which the global communications are the major issue for the industry and the investigators.

An important part of this concept of global communications is the Internet. This network is growing up thanks to the introduction of different types of networks. Two kinds of these are the wireless and the satellite that allow a global coverage. We can serve all users in any place.

For the study of this two networks the standard DVB-S<sup>1</sup>, and DVB-RCS<sup>2</sup>, for the satellite part, and the IEEE<sup>3</sup> standard 802.11, for the WiFi part, are essential in the development of the simulator used to know the behavior of these kind of networks and their interconnection.

---

<sup>1</sup> DVB-S is the acronym of Digital Video Broadcasting-Satellite

<sup>2</sup> DVB-RCS is the acronym of Digital Video Broadcasting-Return Channel via Satellite

<sup>3</sup> IEEE is the acronym of Institute of Electrical and Electronics Engineers

But, giving the possibility of a Internet connection to the users everywhere is not the only issue to solve. Everyday, appears new applications that requires specific resources. In a wired environment it is easier to give this quality of services (QoS), but in a wireless world is more difficult to achieve them.

Many techniques are discussed in a satellite network. One possible solution is the use of cross-layer techniques, i.e. interaction between non-adjacent layers in the stack layers, but here we mainly talk about the use of Integrated Services and the Differentiated Services, that allow to give QoS to an IP network. The major issue is mapping this QoS of IP to the Capacity Category of the DVB standard.

But also the IEEE 802.11 has been improved to achieve the QoS. We have based our work in the IEEE standard 802.11e, that allows the wireless standard giving QoS to the users. In this part of the standard exists two kinds of access, one is contention-based and the other polling-based.

The work is presented here will be divided in four mainly parts.

In a first part we present the study of QoS in a satellite part, i.e. the requirements of transmitting different kind of services via satellite and solutions above mentioned in order to improve the satellite network in terms of QoS.

Secondly, is presented the wifi part. A briefly summary of the 802.11 standard in order to compare with the proposal solution, the 802.11e. In this part we want present the achievements done comparing the basic standard with the enhancements introduced.

The third part consists in an introduction of the network simulator. We present its general function, remarking the characteristics of the satellite nodes and parameters, and also the wireless node and parameters involved in a simulation of this networks.

Finally we show the results achieve with the simulations done. We can see a first set of simulations under a satellite environment, proving the TCP performance in this kind of

scenarios. Then a study of a wireless network under the HCCA access mode. And finally a prove of the interconnection the two networks.

Thanks to the network simulator employed in the simulations and the patch created in the University of Pisa we can see effects of the different configurable parameters of the standards.





## 2. Satellite part

Satellite networks provide a way to easily and economically provide long-distance communication links (especially intercontinental ones), and hence, their usage is growing. Because of their nature, when interconnected together with traditional LAN networks (and in almost all scenarios they are) they can be the bottleneck of the entire system because of the delay and throughput that they provide. For that reason, getting the maximum performance out of them is very important, and one ways to do so (the one we are exploring) is to apply cross-layer design techniques in satellite networks. If our proposal is giving Quality of Service to all users we have to achieve a better performance of the satellite network and for this reason using cross-layer techniques in its design is an advanced for our study.

For this we have to focuses on cross-layer techniques for satellite networks. We start by exploring the BSM protocol stack, and by giving a brief introduction on Quality of Service (QoS).

Then, we focus on more details by looking at *SatLab*<sup>1</sup> recommendations for implementing the interoperability between different layers in satellite networks, and we emphasize the importance of it with a concrete example (VoIP scenario for satellite networks).

## **2.1. Introduction**

It is not easy to treat the subject of resource allocation in satellite communications, since it comprise practically all layers of a protocol architecture.

The QoS requirements of the applications is the basis upon which all the actions undertaken by the lower layers depend. Immediately under the application layer, the transport layer must manage allocation problems to satisfy the upper levels' requirements.

The transport level is generally based on the TCP protocol. Whatever TCP version one uses, it may work inefficiently on a satellite link, due to the effect that the long satellite round-trip-time (RTT) has on the congestion window. But it is not the only the problem that one has to solve in the satellite network, although TCP has a great influence on the MAC layer, which runs the algorithms for the bandwidth resource allocation. We will not explicitly deal with the QoS solutions at the network layer, as they are not specific for the satellite environment; nevertheless, there is an interaction in mapping, for instance IP QoS classes onto DVB classes. The MAC layer must manage the type of access to the satellite link, and the algorithms for utilizing the common resource. Most of those algorithms implement some optimization of the bandwidth allocation, which influences the TCP goodput. Call Admission Control (CAC) is also managed at the MAC level, being itself a way to block requests that cannot be satisfied, according to the QoS specified by the application.

---

<sup>1</sup> *SatLab is the SatLabs Group, a group promoted by the European Space Agency*

If we study the physical link level we have to remark that the transmission is often affected by a signal fade due to bad atmospheric conditions. Nevertheless, the original QoS of the data, which at this level is expressed in terms of Bit Error Rate (BER), must be maintained as much as possible. The physical link must deal with several problems, such as the signal fade, which must be counteracted by means of techniques that are transparent to the upper layers, but which interfere with the MAC layer, in the sense that there must be a great correlation between the two layers in order to select the best countermeasure.

## **2.2. QoS requirements on the Application layer**

As we said supporting quality of service in a telecommunication system is, nowadays, of a great importance. But this task should be transparent to the user. It does not have to know how it is done but it notes the final degree of quality that it enjoys. In the next sections we will present the different requirements needed according to the chosen multimedia traffic.

### **2.2.1. Conversational services**

The most common use of this scheme is the real-time conversation. The two newest applications, included in this scheme, are VoIP<sup>1</sup> and video conferencing.

This scheme raises the most stringent QoS requirement. The transfer time must be short while, at the same time, variation between information entities of the stream must be preserved in the same way as for real-time streams. The limit in which the delay transfer is acceptable, is very strict, since failure to provide low enough transfer delays results in unacceptable lack of quality. When one talks about real-time conversation, the principal

---

<sup>1</sup> VoIP is the acronym of Voice over IP

points that one has to take care for QoS are to preserve the time relation between information entities of the stream, and the conversational pattern, that has to be stringent and of low delay. This kind of application delivers time-based information in real-time, where time-based information is user data that has an intrinsic time component. Video, audio and animation are examples of time-based information, as they consist of a continuous sequence of data blocks that are presented to the user in the right sequence at pre-determined instants.

### ***Conversational voice***

Audio transfer delay requirements depend on the level of interactivity of end users. To preclude difficulties related to the dynamics of voice communications, ITU-T<sup>1</sup> Recommendation G.114 recommends the following general limits for one-way transmission time. These are shown in the following table.

0 to 150 ms	preferred range[< 30 ms, user does not notice any delay at all; < 100 ms, user does not notice delay if echo cancellation is provided and there are no distortions on the link]
150 to 400 ms	acceptable range (but with increasing degradation)
above 400 ms	unacceptable range

*table 2.1. ITU-T recommendations for voice communications<sup>2</sup>*

The human ear is highly intolerant of short-term delay variation, known as jitter, so the latter should be kept below a very low limit (1ms is suggested). In the other hand, the human ear is tolerant of a certain amount of distortion of the speech signal. Finally, this type of connections requires the allocation of symmetrical communication resources.

---

<sup>1</sup> ITU-T is the acronym of International Telecommunication Union - Terrestrial

<sup>2</sup> extract from (VI)

As we are talking of a satellite connection, it is important to talk about the possible satellite constellation used. The use of GEO<sup>1</sup> satellites imposes severe constraints in order to achieve good QoS for conversational voice (its round-trip delay is about 250 ms). Instead, LEO<sup>2</sup> or MEO<sup>3</sup> satellites can be used without problems.

### ***Videophone***

Videophone implies a full-duplex system, which carries both video and audio and is intended for use in a conversational environment. Therefore, the delay requirements are the same as for the conversational voice. The only difference is that the video must be synchronized with the audio in order to provide “lyp-synch”, i.e., synchronization of the speaker’s lips. In fact it is difficult to meet these requirements, due to the long delays incurred in video codecs. Also, the human eye is tolerant to the loss of some information, so it is possible to accept some packet loss. GEO satellites introduce the same constraints for QoS in videophone as in conversational voice services. But if we take into account the lyp-synch problem, then GEO cannot support this service.

### ***Interactive games***

The requirements for games depend on the specific game, but normally this kind of services requires very short delays, and normally this value is around 250 ms. For that reason, the use of GEO satellites depends on the interactive game application.

### ***Two-way control telemetry***

This service is an example of the not requirement of real-time. It can suffer a delay value of 250 ms but the application does not tolerate information loss. In this situation the inconvenience of using GEO satellites is the possible information loss that can introduce the noisy links. It must be necessary the use of error control techniques to solve this problem.

---

<sup>1</sup> GEO is the acronym of Geostationary Earth Orbit

<sup>2</sup> LEO is the acronym of Low Earth Orbit

<sup>3</sup> MEO is the acronym of Medium Earth Orbit

**Telnet**

This application can require a short delay in order to provide essentially instantaneous character echo-back. So the GEO satellites could have a high RTT for this kind of service, so the user can perceive a bad telnet application. In order to solve this problem a terminal auto-echo can be used.

The following table shows a summary on the above applications.

Medium	Application	Degree of symmetry	Data rate	Key performance parameters and target values		
				End-to-end One-way Delay	Delay variation within a call	Information loss
Audio	Conversational voice	Two-way	4-25 kbps	< 150 ms preferred < 400 ms limit	< 1 ms	< 3% FER
Video	Videophone	Two-way	32-384 kbps	< 150 ms preferred < 400 ms limit Lip-synch < 100 ms		< 1% FER
Data	Telemetry two-way control	Two-way	<28.8 kbps	< 250 ms	NA (Not applicable)	Zero
Data	Interactive games	Two-way	< 1 kB	< 250 ms	NA	Zero
Data	Telnet	Two-way (asymm.)	< 1 kB	< 250 ms	NA	Zero

*table 2.2. Conversational services. End-user performance expectations<sup>1</sup>*

---

<sup>1</sup> extract from (VI)

### 2.2.2. Interactive services

This service class is applied when a human or machine is on-line requesting data from a remote server, and is characterized by the request/response pattern of the end user. This application has to support interactive non-real time services with low end-to-end round trip delay. So the fundamental requirements are the request-response pattern and the preservation of the payload content. In the next points we will present examples and their requirements.

#### *Voice messaging and dictation*

This service can tolerate the same information loss requirements as for the conversational voice. But here one has a high tolerance to delay, since the conversation is no direct form.

#### *Web-browsing*

The main consideration here is the delay in which a web page appears to the user. The best improvement is reducing this time achieving a value of 2-4 s.

Medium	Application	Degree of symmetry	Data rate	Key performance parameters and target values		
				<i>One-way Delay</i>	<i>Delay variation</i>	<i>Information loss</i>
Audio	Voice messaging	Primarily one-way	4-13 kbps	< 1 s (playback) < 2 s (record)	< 1 ms	< 3% FER
Data	Web-browsing	Primarily one-way		< 4 s/page	NA	Zero
Data	Transaction services - high priority	Two-way		< 4 s	NA	Zero
Data	E-mail	Primarily one-way		< 4 s	NA	Zero

table 2.3. Interactive services. End-user performance expectations<sup>1</sup>

<sup>1</sup> extract from (VI)



### **2.2.3. Streaming services**

The streaming services are principally unidirectional and with a high rate of utilization and short time variations between information entities within a flow. but with this services does not exist a strict limit of delay and jitter. And also is important to remark that the packet loss has not a limit. The fundamental QoS requirements of real-time streams are the unidirectional continuous streaming and the reservation of the time relation between information entities of the stream. We have to maintain the data flows continuously.

#### ***Audio streaming***

One expects to achieve a better performance than conventional telephony so one has to impose a higher packet loss requirement. On the other hand the delay requirements are not too strict.

#### ***One-way video***

In this occasion no conversational element is involved, so the delay requirements can be relaxed.

#### ***Still image***

In receiving images the human eye is very tolerant, but the loss of one bit can cause disturbances in some still images format. So one has to achieve zero errors but not to control so much the delay.

Medium	Application	Degree of symmetry	Data rate	Key performance parameters and target values		
				<i>Start-up Delay</i>	<i>Transport delay variation</i>	<i>Packet loss at session layer</i>
Audio	Speech, mixed speech and music, medium and high quality music	Primarily One-way	5 - 128 kbps	< 10 s	< 2 s	< 1% Packet loss ratio
Video	Movie clips, surveillance, real-time video	Primarily One-way	32-384 kbps	< 10 s	< 2 s	< 2% Packet loss ratio
Data	Bulk data transfer/retrieval, layout and synchronization information	Primarily One-way	<384 kbps	< 10 s	NA (Not applicable)	Zero
Data	Still image	Primarily One-way		< 10 s	NA	Zero

table 2.4. Streaming service. End-user performance expectations<sup>1</sup>

#### 2.2.4. Background services - applications

This services refers to data that is not expected in a certain time. Also, the packets' content must be transparently transmit so one has to maintain a very low level of error rate.

The fundamental QoS elements for background traffic are: the destination is not expecting the data within a certain time; and to preserve the payload content. So the requirements are the same that we explained for the non-real time services without any special requirement on delay.

Essentially one has to received the information without any error without any delay in it. But this is not really true, because the information contained could be not useful if it was

<sup>1</sup> Extract from (VI)

received too late. Services like fax, email, SMS follows this description. Here the GEO satellites can be employed without the tight constraint on delay of conversational services.

In order to summary the services explained above here we show a table with the possible applications depending on their QoS requirements.

Service class	Conversational (delay $\ll 1s$ )	Interactive (delay $\sim 1s$ )	Streaming (delay $< 10s$ )	Background (delay $> 10s$ )
Error tolerant	<i>Conversational voice and video</i>	<i>Voice messaging</i>	<i>Streaming audio and video</i>	<i>Fax</i>
Error intolerant	<i>Telnet, interactive games</i>	<i>e-commerce Web Browsing</i>	<i>FTP, still image paging</i>	<i>email arrival notification</i>

*Table 2.5. Application examples in terms of QoS*

## 2.3. Transport layer

Another important point is the end-to-end performance using the TCP/IP protocol, and specifically this protocol via satellite. This end-to-end performance must be affected by the significant delay of the satellite link. The great RTT of GEO satellite links greatly limits the TCP congestion window growth in time, while each packet loss, due to data corruption, is interpreted as a congestion loss by TCP.

### 2.3.1. TCP protocol: mechanisms and versions

When we study the TCP protocol one can see that it provides, basically, two functions: flow control and congestion control.

The flow control allows an adequate transmission between two nodes. The main element of this scheme is the sliding window, that basically consists in the fact that the receiver could control the amount of data that can receive, so control the data being sent by the TCP sender. This control of data is made by the size of the sliding window. This value represents the maximum amount of in-flight data. This value is being actualized every ACK that can increase the value.

The congestion control is based on two algorithms, called slow start and congestion avoidance. These algorithms are based on two variables: the congestion and receiver window (cwnd and rwnd) and the slow start threshold (ssthresh). The congestion window, managed by the sender, represents the perception of network congestion, and the receiver window represents the amount of available buffer space at the receiver. Both are actualized periodically.

The congestion control works as follows:

- if  $cwnd < ssthresh$ , the slow start algorithm is adopted and at the receipt of an ACK the cwnd increase its value in an exponential manner. So in this algorithm system tries to enlarge the cwnd in a sufficiently fast, but controlled, way.
- when the cwnd achieves the ssthresh the algorithm use is the congestion avoidance. In this case, at the receipt of an ACK the cwnd increases its value in a linear manner.

But if one ACK is not received during an RTO<sup>1</sup>, is assumed that a segment is lost due to congestion of network. This implies the reduction of the ssthresh to one-half of its current value, and the value of cwnd and rwnd is reset to its initial value.

---

<sup>1</sup> RTO is the acronym of Retransmission TimeOut

### ***TCP New Reno***

This is the most common TCP version adopted in modern operating systems. New Reno utilizes four algorithms: slow start, congestion avoidance, fast retransmit and fast recovery. New Reno resets the RTO timeout when the first partial ACK is received, i.e. impatient version. New Reno adds some bug fixes with respect to previous implementations. Also, it introduces the *recover* variable used to avoid multiple fast retransmits.

### ***TCP SACK***

The selective ACKnowledgment option allows the receiver to inform the sender about lost segments. SACK uses the TCP option field in the TCP header (the length of the TCP option permits to ask the retransmission of maximum three segments). SACK permits to recover more than one packet lost per RTT, a capability extremely useful when using large *cwnd* values. The TCP sender keeps the transmitted segments in a structure called scoreboard.

### ***TCP Westwood***

Westwood aims at limiting the consequences of the losses introduced by the wireless channel that are always erroneously ascribed to congestion by standard TCP. To this end, it introduces a modification to the fast recovery algorithm, called faster recovery: after a loss, *ssthresh* is a function of the estimated available bandwidth, thus avoiding the dramatic slow-down of the transmission rate of standard TCP standard versions. The bandwidth is estimated by averaging the rate of returning ACKs.

## **2.4. QoS Architecture for IP networks**

It's necessary to offer different services, and this implies different priorities, to the users in any network. So there should be an interface between application and transport strata. The application stratum provides the service to users and the transport supports this service guaranteeing a quality. This guarantee is based on the use of the distinct classes that are organized in priorities.

In order to achieved the necessary QoS, the user request this class by QoS signalling protocols (RSVP). this signalling is exchanged with user endpoints and the application plane. Very important possibilities for this architecture are that media path can cross several transport domains and transport domain can support different policies and QoS mechanisms. The signalling provides a description of the characteristics of the user session based on the QoS parameters like: frames per packet, frame size, jitter buffer delay, FEC, mean delay variation, packet loss.

The main assumptions are the following:

- The media path may cross several transport domains.
- Transport domains may support different QoS mechanisms and policies.
- Routing of calls between transport domains will be under the control of the application plane (one or more SPs).

Routing of calls within the transport domains will be independent of the application stratum.

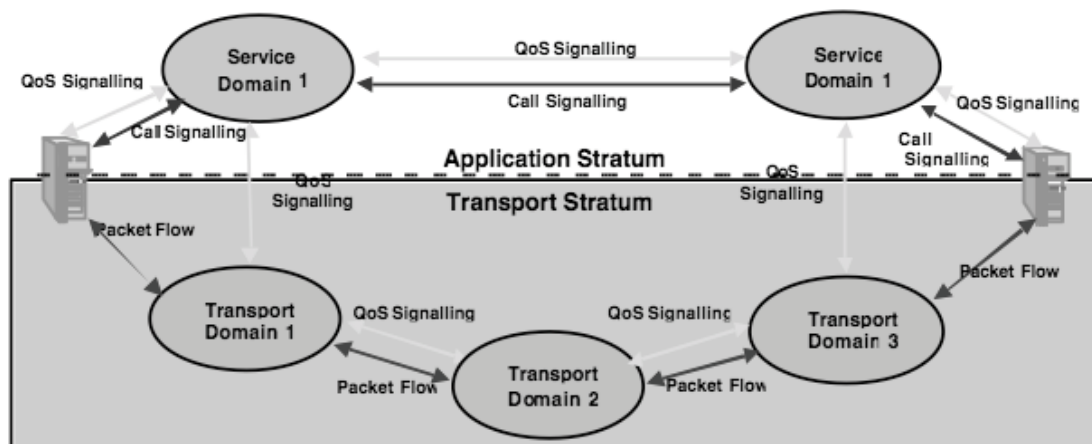


figure 2.1. Application and transport strata

### 2.4.1. QoS Network building blocks

To offer QoS services outlined above in a complete and efficient way can be complex, and can involve multiple inter-related aspects. For example, in case of network resource contention or congestion, to maintain the expected service response requires a variety of functions working at different time-scales, ranging from careful network planning based on traffic patterns over a long period (grouped in the Management Plane) to differential resource allocation and admission control based on the current network load conditions (in the Control Plane).

The range of mechanisms involved in QoS can be considered as a set of **building blocks**, or functions which can be combined in different ways to provide different overall objectives (e.g. type of network, or for guaranteed or relative QoS). These building blocks may be classified in the Management, Control and Data Planes as follows:

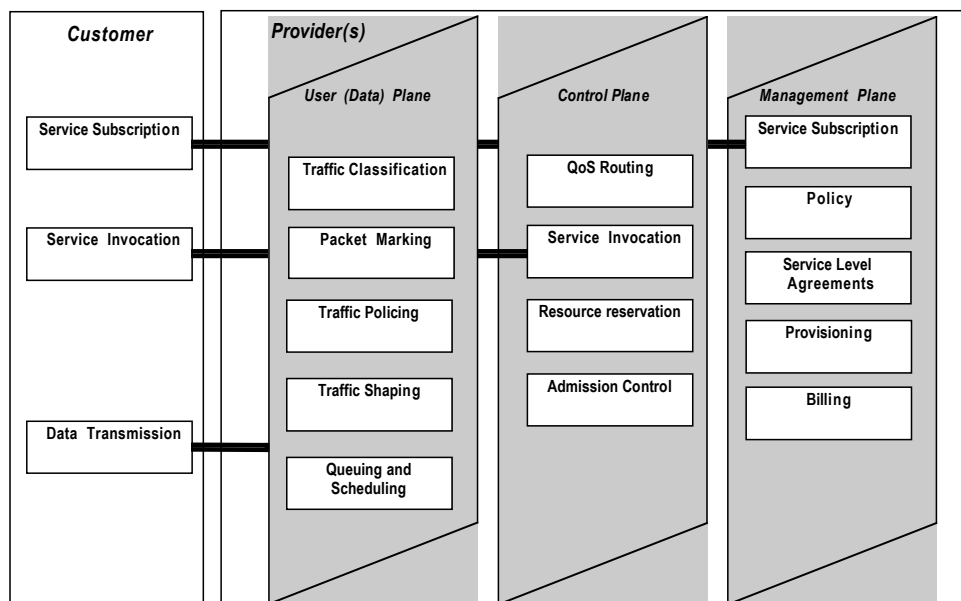


figure 2.2. QoS building blocks

We need to interact between this building blocks that allows to obtain the QoS. it's necessary that QoS parameters are exchanged. these include transaction performance at the packet level and service reliability/availability expectations in the form of traffic priority levels for specific network functions. some examples of mechanisms to convey these parameter values are signalling and database lookups and include:

- QoS signalling: signalling of QoS parameter requirements per service/flow between functional blocks
- Call Signalling: service invocation, resource reservation
- Policy control: parameters for admission control, policing, marking etc.

## **2.5. Proposed QoS support techniques on MAC layer<sup>1</sup>**

As it is well known, IETF<sup>2</sup> organization is proposing two different ways to handle QoS in IP networks, namely Integrated Services (IntServ) and Differentiated Services (DiffServ). The Intserv is based on the reservation of resources on a per-flow basis, while DiffServ is based on classification of traffic into groups and provides QoS support for aggregated traffic flows.

### ***IntServ***

In IntServ the main focus is giving services on a per-flow basis for every single, in a single manner. Flow is normally unidirectional and can have more than one recipient and is identified with source and destination IP address and port number. A flow of packets is the traffic associated with a given connection or connectionless stream having the same source host, destination host, class of service and session identification.

---

<sup>1</sup> Based on the SatLabs recommendations

<sup>2</sup> IETF is the acronym of Internet Engineering Task Force



The IntServ routers must maintain per-flow state information and must reserve end-to-end resources for individual flows in order to provide QoS. The major flaw of this approach is that it can be too complex to perform and may not scale well for large volumes of network traffic. This is due to the high control signals existing to maintain the state information at routers.

### ***DiffServ***

On the other hand, DiffServ is simple and easy to implement and supports a range of network services, which are differentiated on the basis of performance. Traffic flows are aggregated into classes: all the flows in one class will be treated in the same way. For example, when multiple voice connections are sent over the network, the same QoS provision scheme will be applied to all these connections together by using DiffServ, while IntServ will treat each voice stream separately. But the major issue to improve in this service is the fairness problem that can cause non-deterministic and non-quantifiable QoS conditions. Because it is more convenient DiffServ has been adopted as the model of choice for most satellite systems.

DiffServ replaces per-flow service with a Per-Hop Behaviour (PHB), applicable to traffic aggregates and per-flow states in network nodes are no longer required. So the end-to-end performance is now done using multiple PHB. DiffServ standards<sup>1</sup> defines five PHB classes, namely Expedited Forwarding (EF), four Assured Forwarding (AF) classes and Best Effort (BE) service of the original Internet Protocol. At each network node each packet is treated according to its class, as defined by the Type of Service (ToS) bits in the packet header.

Traffic packets will be mapped into PHB classes, i.e., packets containing voice stream will be mapped in EF class and will be transmitted before data with lower priority class (AF, BE).

---

<sup>1</sup> Defined by the IETF RFC2475

Mac layer QoS support consists in capacity request calculation/generation and scheduling/dispatching of packets from the supported queues to fill-up the assignments in Terminal Burst Time Plan (TBTP). In order to support prioritizing some PHBs over the others, DVB-RCS provides a number of different capacity requests mechanism for satellite networks there:

- *Constant Rate Assignment*: this is guaranteed capacity assignment where each PHB has guaranteed capacity. It is used for jitter-intolerant applications like VoIP or for signalling
- *Volume Based Dynamic Capacity (VBDC)* – This is dynamic capacity assignment which is based on volume. It is used for non-real time applications like FTP
- *Rate Based Dynamic Capacity (RBDC)* – This is a dynamic capacity assignment which is based on rate. It is used for near real-time jitter-tolerant such as streaming video
- *Free Capacity Assignment (FCA)* – This is a dynamic capacity assignment which is based on free capacity, if there is any left (which won't be used otherwise). It is used for non-real time applications).

Using CRA implies that the capacity is reserved and guaranteed during the whole logon time of the terminal and terminal does not have the necessity of demanding capacity every frame, so it reduces the delay. But, in the other hand even if the terminal has no traffic, capacity will be allocated.

RBDC is used when we have high-variable rate traffic which can tolerate MAC scheduler dynamic response time latency, capacity is also reserved and guaranteed, but a terminal has to ask dynamically for slots depending of the traffic sending rate.

When VBDC is used, capacity is not guaranteed or reserved and terminal has to ask dynamically for VBDC slots depending on the amount of packets to transmit. It is assigned after satisfying total CRS and RBDC capacity components.

The only capacity requests mechanism that is not necessary to requests is the FCA. If they are unused slots they will be assigned to terminals.

An important parameter in the QoS serving is the Request Class (RC), which is a representation of a PHB at the MAC layer, in the DAMA controller. RC is a concept similar to the PHB, but applied at the MAC layer instead in IP layer. The Return Channel Satellite Terminal (RCST) maps each PHB onto an RC and it shall be possible to map any PHB to any RC.

## **2.6. Relation between QoS Traffic Classes and Capacity Categories**

Forwarding of packets on IP layer is in reality realized by the mechanisms that are applied on MAC layer. Capacity Request strategies allow for efficient combinations of DVB-RCS capacity categories in order to maximize the QoS and return link efficiency. So we need to map the QoS traffic classes to capacity categories supported by the scheduler on the MAC layer.

On one hand one has three main QoS traffic types for DiffServ, called: EF (Expedited Forwarding), AF (Assured Forwarding) and BE (Best Effort). On the other hand, one has Capacity Request strategies, known as: HG (High Grade), MG (Medium Grade), LG (Low Grade), RB (Rate-Based), VB (Volume-Based), PERF (Performance), and UTIL (Utilization). One can define the following DiffServ mapping:

*QoS class*  $\longrightarrow$  *Capacity Request Strategy*  $\longrightarrow$  *Capacity Category*

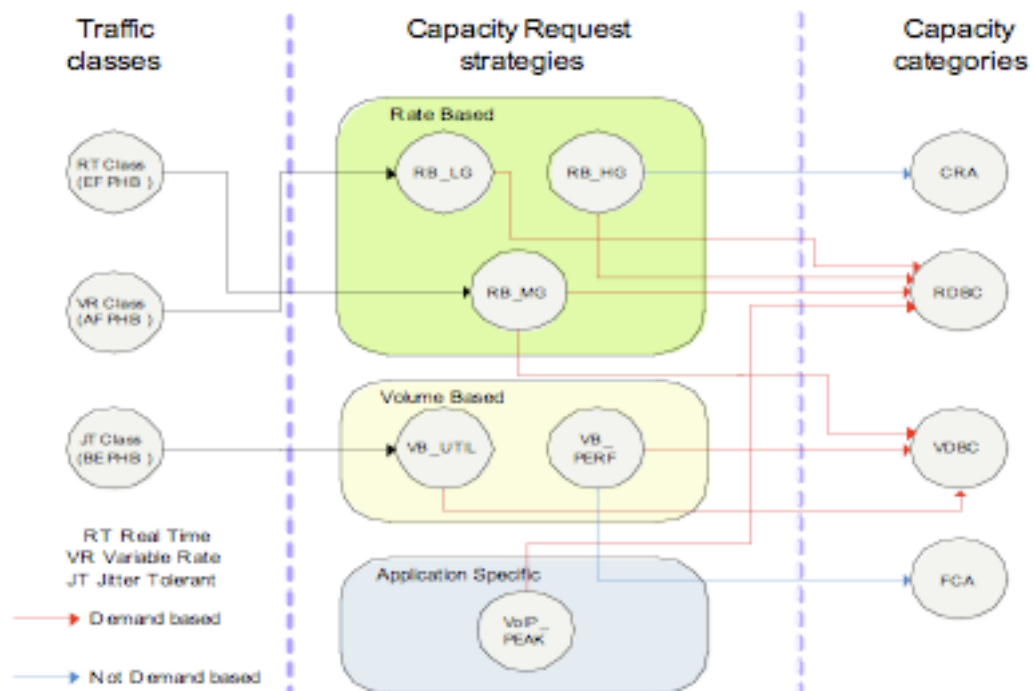


figure 2.3. Example mapping of QoS traffic classes to Capacity Strategies



## 3. WiFi part

The first standard used in a wireless environment is the IEEE 802.11. This standard fix the needing in the creation of a wireless network. But with time it's not only necessary to offer a network without wires. Nowadays the users have many applications that requires some specific resources in order to function correctly. For that reason appears the 802.11e. This specification belongs to the distinct specifications of the standard 802.11.

With the 802.11e we are allowed to offer a wireless network able to give Quality of Service to its users. This protocol is an emerging standard that modifies the medium access control, where the differentiation of services are made.

### 3.1. Introduction to 802.11

The IEEE 802.11 Wireless Local Area Network is being deployed widely and rapidly for many different environments. Its main characteristics are simplicity and robustness against failures due to the distributed approach of its *medium access control (MAC)* protocol. There is a diverse set of versions existing in market that apply different transmission schemes and operate in different frequency bands.

Today, 802.11 WLAN can be interpreted as a wireless version of Ethernet supporting best effort service. But, nowadays, the real interest in wireless networks is making them able to support Quality of Service. The 802.11 working group initiated an activity to enhance the current 802.11 MAC protocol to enable support of applications requiring QoS.

We want to describe the limitations of the 802.11 in terms of QoS. Considering a *basic service station (BSS)*, which is composed of an *access point (AP)* and a number of stations associated with the AP.

The basic 802.11 MAC protocol is referred to as *distributed coordination function (DCF)* and operates as a listen-before-talk scheme. This is based on *carrier sense multiple access (CSMA)*. Applying the DCF, a station determines individually when to access the medium. The decision of accessing to the medium is distributed among all stations. Stations deliver MSDU's<sup>1</sup> ("talk") after detecting that there are no other transmissions in progress on the wireless medium ("listen"). If two or more stations detect the medium idle at the same time, and inevitably collision occurs.

To reduce the probability of collisions, the DCF applies a *collision avoidance (CA)* mechanism, where stations perform a so-called *backoff procedure* before initiating a

---

<sup>1</sup> MSDU is the acronym of MAC Service Data Unit

transmission. After detecting the medium idle for a minimum duration called *DCF interframe space (DIFS)*, station keep sensing the medium for an additional random time called *backoff time*. A station initiates its transmission only if the medium remains idle for this additional time. The duration of this random time is determined by each station individually, as a multiple of a slot time. A new independent random value is selected for each new transmission attempt.

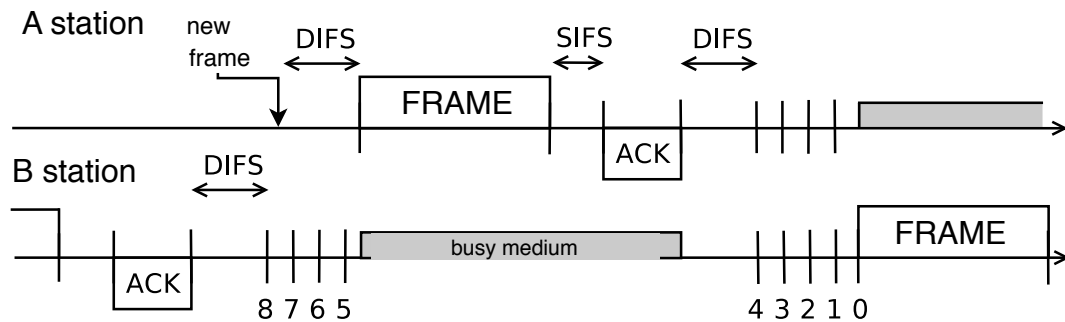


figure 3.1. Backoff procedure in the DCF mode

This random time is selected in a random manner. This number of slots is compressed between 0 and  $CW^1$ , which is initially set to a minimum value called  $CW_{min}$ .

All the stations select the same  $CW_{min}$ , but they selected its random backoff time individually. The use of the same minimum value makes that users have no priority one among the others, so we haven't any mechanism to differentiate between stations and their traffic and therefore, no QoS support in the DCF.

During the downcounting of its backoff counter if the medium is idle, the station stops downcounting and defers from medium access until the medium becomes idle for a DIFS again.

<sup>1</sup> CW is the acronym of Congestion Window



Since collisions may occur due to CSMA/CA<sup>1</sup> protocol, the stations have to be informed of the performance of transmission, using an ACK<sup>2</sup> for each MPDU<sup>3</sup> transmitted. If this ACK is not received the station needs to retransmit the MPDU.

Every failed transmission makes increase the CW value. The CW value is doubled after any unsuccessful transmission up to a maximum value called  $CW_{max}$ . This increase allows to reduce a future collision between stations, so, higher value of CW lower probability of a collision.

But not only a backoff counter is done before a transmission. After each successful transmission the station, who performs as the sender, initiates a backoff called post-backoff. This guarantees that there is always one random backoff time between two consecutive exchange, and therefore is an important mechanism to guarantee DCF functionality.

Parameter	Value
<i>SIFS</i>	<i>10 <math>\mu s</math></i>
<i>Te</i>	<i>20 <math>\mu s</math></i>
<i>DIFS</i>	<i>50 <math>\mu s</math></i>
<i>CW<sub>min</sub></i>	<i>32</i>
<i>CW<sub>max</sub></i>	<i>1024</i>

*table 3.1. 802.11b parameters*

---

<sup>1</sup> CSMA/CA is the acronym of Carrier Sense Multiple Access/Collision Avoidance

<sup>2</sup> ACK is the acronym of ACKnowledgment

<sup>3</sup> MPDU is the acronym of MAC Protocol Data Unit

Another important point that is performed by the 802.11 allows to fragment the frames into shorter MPDU's. This causes a lower loss of information in case of a collision occurs. Only before the first fragment is transmitted, the sender initiates a backoff counter. The rest of MPDU fragments are sent subsequently. With this we can solve the retransmission of a long MPDU, but in cons we have an increase of the overhead of fragmentation.

Another problem that appears in the wireless networks is the *hidden stations*. This refers to the stations that have a very bad channel condition and it is not able to detect other ongoing transmissions. If this kind of stations initiates transmission during another transmission that causes unacceptable levels of interference at the receiving station. In order to solve this problem, 802.11 uses a RTS/CTS<sup>1</sup> mechanism. Before transmitting an MPDU, the station transmits a short RTS control frame, followed by the CTS control frame that is transmitted by the receiving station. These frames include information on how long it will take to transmit the next data frame or fragment, and the corresponding ACK. At the moment that the stations received the RTS or the CTS all stations in transmission set their timer, called *Network Allocation Vector*, with the duration announced in the RTS/CTS frames. So the stations that receive these frames will not start any transmission until this timer expires.

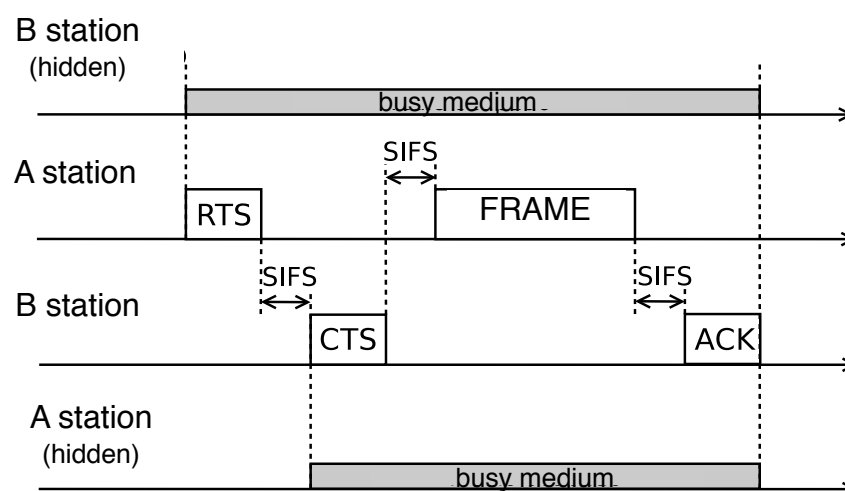


figure 3.2. RTS/CTS performance

<sup>1</sup> RTS/CTS is the acronym of Request-To-Send/Clear-To-Send

Also, between two consecutive frames in the sequence of RTS, CTS, MPDU and ACK frames, a *Short InterFrame Space (SIFS)* gives transceivers time to turn around. We have to note that DIFS is longer than SIFS which gives CTS and ACK always the highest priority to access to the wireless medium.

But the main focus that this work is focused is on the support QoS in the wireless medium. The 802.11 uses the *Point Coordination Function (PCF)*. This provide mechanisms for prioritized access to the wireless medium, and is centrally coordinated by a station called the *Point Coordinator (PC)*, that typically is the AP.

With the PCF, we find two different times of transmission: the *Contention Free Period (CFP)* and the *Contention Period (CP)*, that alternate periodically over time. This times formed together a superframe. The CFP is used to access during the PCF while the CP is used to access with the DCF. But this superframe must include a CP time, that, at least, allows to transmit one MSDU of maximum size, and also at the slowest transmission rate under the DCF.

A superframe starts with a beacon frame, send by the AP, that are used to maintain synchronization of the local timers in the stations and to deliver protocol-related parameters. These are sent periodically and the stations knows these interval. These are referred as *Target Beacon transmission Time (TBTT)* and are announced in the previous beacon frame.

During the CPF the contention among stations does not exist. Instead of contention, stations are polled. So the PC polls a station for a frame pending of transmission. Whenever the PC itself has a pending frame to a station, it uses a combined data and poll frame by piggybacking the CF-Poll frame onto the data frame. The station has to respond sending an ACK for the frame sent, and piggyback an MPDU. If the CP doesn't received

any response of the station, during a PIFS<sup>1</sup>, it polls another station or finish the CFP. So the maximum idle time during a CFP is a PIFS. In order to finish the CFP, a specific control frame, named CF-End, is transmitted by the PC as the last frame to indicate the end of the CFP.

But this solution to give QoS services in a wireless medium is not sufficient and many problems have to be solved. Some of them are unpredictable beacon delays and unknown transmission durations of the polled stations.

### **3.1.1. 802.11 lacks**

The DCF mechanism gives fairness at long term, understanding that all nodes have the same probability of occupy a time slot. But, this fairness should be desirable at first moment because it has a consequence: in a network with many modulation schemes and, for that reason, many transmission bit rates, if one station transmits at minimum rate the rest of stations will receive the same bandwidth. This phenomenon is known as performance anomaly, and it becomes in a severe degradation of performance.

On the other hand the main reasons of not giving QoS, not in the DCF mode and also in the PCF, in 802.11 are:

- The DCF mode does not offer a traffic differentiation mechanism, so all stations receiving the same kind of service, best effort.
- The PCF mode, that is not very extended, also does not differentiate different flows. The beginning of the polling period by the AP is not immediate, so it is difficult to determinate the transmission time of every station.

---

<sup>1</sup> PIFS is the acronym of PCF InterFrame Space

### 3.2. The 802.11e, QoS support mechanisms

The enhancements introduced in the 802.11e are under development. In this extension the IEEE introduces the *Hybrid Coordination Function (HCF)* for QoS support. The HCF also defines two medium access mechanisms, the contention-based channel access and the controlled channel access.

The contention-based channel access is referred to as *Enhanced Distributed Channel Access (EDCA)* and the controlled as *HCF Controlled Channel Access (HCCA)*, in which our simulations are based.

This two mechanisms also uses the two times defined previously, the CP and the CFP. While the EDCA only operates at CP period, the HCCA can works at both.

An important improvement introduced is that the stations have a limited time to transmit and use the radio resources. This time is the known as the *transmission opportunity (TXOP)*. This TXOP is a limit of time in which the station is allowed to deliver MSDU's. It is defined by its starting time and duration. Also we differentiate two types of TXOP. If it is obtained via contention-based medium access, this are referred to as EDCA-TXOP. Alternatively, a TXOP obtained by the HC<sup>1</sup> via controlled medium access is referred to as HCCA-TXOP or polled TXOP. The duration of the EDCA-TXOP is limited by a QBSS<sup>2</sup>-wide parameter referred to as *TXOP-limit*. This TXOP-limit is distributed regularly by the HC within an information field of the beacon. With this time we can control the station time delivering and the MSDU delivery delay.

---

<sup>1</sup> HC is the acronym of Hybrid Coordinator

<sup>2</sup> QBSS is the acronym of QoS Basic Service Set

Another enhancement is that stations no transmit across the TBTT. Only a station can transmit if it has sufficient time before the upcoming TBTT. With this, the expected beacon delay is reduced, and allows to the HC a better control over the medium, especially if the optional CFP is used after beacon transmission.

Additionally, a 802.11e station is allowed to transmit frames directly to another station in a QBSS, without involving communication with the AP, contrarily in the 802.11 all data frames are sent or received by the AP. But to achieve this direct connection between stations they have to use the *Direct Link Protocol (DLP)* before initiating direct frame transmissions.

### **3.2.1. HCF Contention-Based Medium Access (EDCA)**

In this medium access the QoS is supported introducing the concept of *access categories (ACs)*. This categories allows to divide the different frames in function of their priority. Also every station has multiple independent backoff entities. So the MSDUs are treated by different of this entities in parallel in one 802.11e station. In order to prioritized the MSDUs every backoff entity used the AC-specific contention parameters, called EDCA parameter set. In this access exists four ACs that exists in every station. Every AC are labeled corresponding for its kind of traffic that carries, i.e. AC\_VO (voice), AC\_VI (video), AC\_BE (best effort) and AC\_BK (background). The EDCA parameter set fix parameters like the interframe spaces, contention windows and many other, that defines the priorities of every AC.

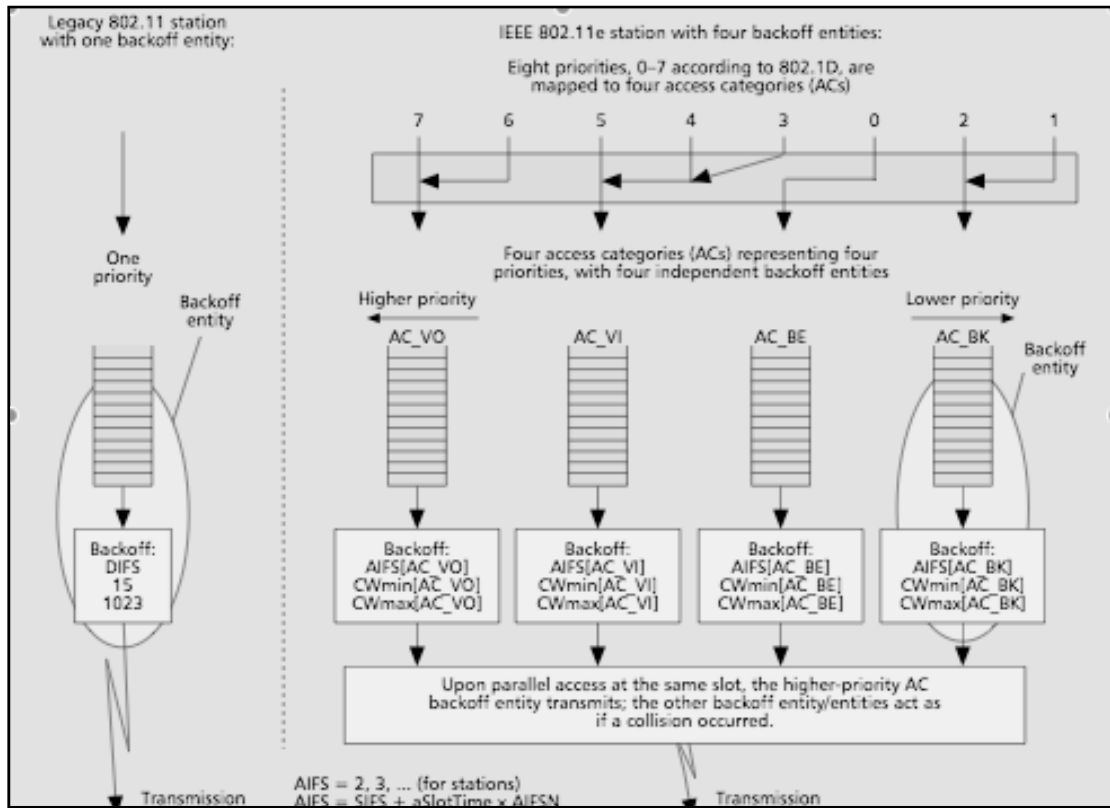


figure 3.3. 802.11 station and 802.11e station with four ACs within one station

We have to fix the EDCA parameter set for every backoff entity of the stations. The value that are used is defined by the HC, and their value can vary in time using the information fields in the beacon frames. The same EDCA parameter set is used by the backoff entities of the same AC in different stations. It is important that all equal AC of every station has the same set of parameters, because they have to give the same priority for all the stations.

Each backoff entity within a station independently contends for a TXOP. It starts dncounting the backoff-counter after detecting the medium being idle for a duration defined by the *Arbitrary Interframe Space* ( $AIFS[AC]$ ) instead of DIFS, which is used in 802.11. This  $AIFS[AC]$  is like the DIFS but can be enlarged using the  $AIFSN[AC]$ <sup>1</sup>. This number must be selected by the HC such that the earliest access time of EDCA stations is DIFS.

<sup>1</sup>  $AIFSN[AC]$  is the acronym of Arbitration Interframe Space Number

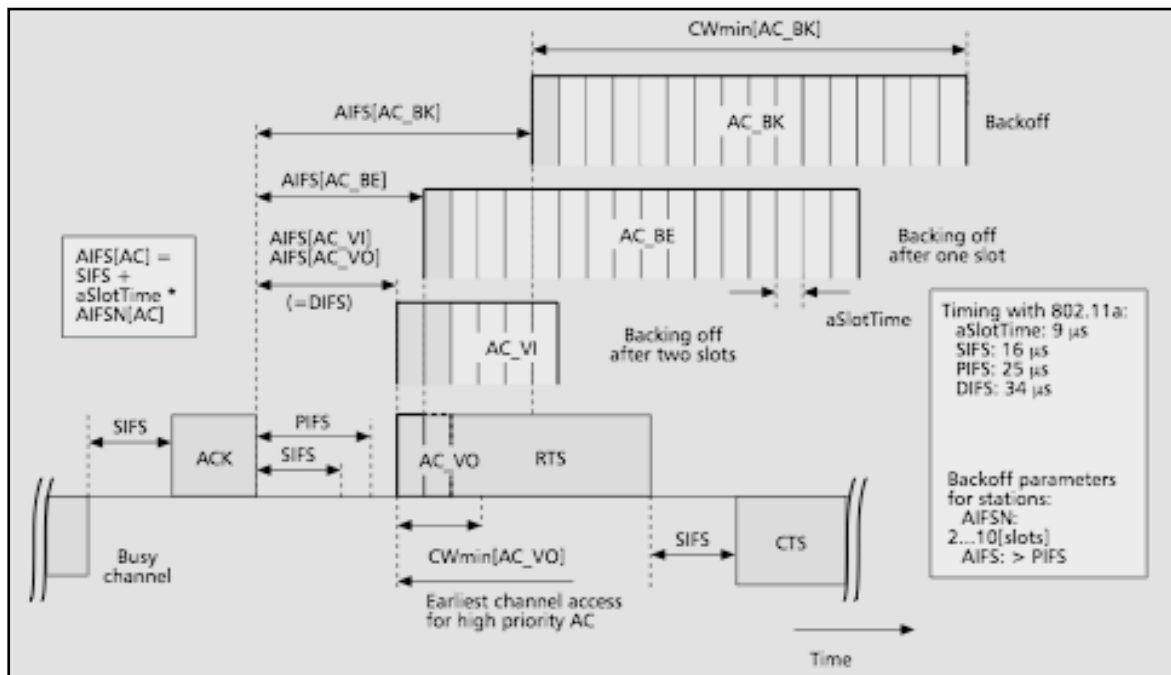


figure 3.4. In EDCA multiple backoffs entities contends for medium access with different priorities

Another important parameter, depending on the AC, is the value of the minimum size of the CW, the  $CW_{min}[AC]$ . Similar to the 802.11, the initial value of the backoff counter, use to access to the medium, is set in an interval defined by the CW. But the major difference between DCF and the 802.11e EDCA in terms of backoff countdown rule is that the first backoff countdown occurs at the end of the  $AIFSN[AC]$  interval, and a frame transmission is initiated after a slot from the moment when the backoff counter becomes zero.

The positions and sizes of the contention windows relative to each other, as defined per AC by the EDCA parameter set, are important factors to define relative priority in medium access per AC. Every unsuccessful transmission becomes in an increasing of the contention window, always without exceeding the  $CW_{max}[AC]$ . Also this maximum value is defined in the EDCA parameter set. A smaller  $CW_{max}[AC]$  becomes in a higher priority to access to the medium. But reducing excessively this value becomes, also, in an increasing of the collision probability. And it is important to remark that exists retry counters that limit the number of retransmissions. The 802.11e protocol also defines the maximum MSDU lifetime per AC, that defines the maximum time that a frame can remain in the MAC. If a frame exceed this parameter the frame is dropped without being transmitted. This



parameter is useful in transmissions that are delay sensitive, because is not useful transmit a frame of voice in retard.

Another backoff parameter is the  $TXOP_{limit}[AC]$ . This is defined per AC as part of the EDCA parameter set. As this parameter is larger, the capacity of share is larger for this AC. Once a backoff entity obtained a TXOP, this deliver more than one MSDU consecutively during the same TXOP. This important concept in 802.11e is referred to as continuation of an EDCA-TXOP.

As we said before, every station has four different backoff entities, each of them having their own EDCA parameter set. During contention, when the counters of two ore more backoffs entities in the same station reach zero at the same time, a virtual collision occurs. Upon access to the same slot by more than one backoff entity with their higher priority will transmit, whereas all other backoff entities act as if a collision occurred on the medium. Also it can occur that the transmission of a backoff entity with the higher priority can collide with another transmission initiated by other stations.

Parameter	AC_VO	AC_VI	AC_BK	AC_BE
$CW_{min}$	8	16	32	32
$CW_{max}$	16	32	1024	1024
$TXOP$	3 ms	6 ms	3 ms	0

*table 3.2. Standard configuration for EDCA in 802.11b*

### 3.2.2. Hybrid Coordination Function, Controlled Medium Access (HCCA)

The controlled medium access of the HCF, referred to as *HCF Controlled Channel Access (HCCA)* extends the EDCA access rules by allowing the highest priority medium access to the HC during the CFP and CP.

A TXOP can be obtained by the HC via the controlled medium access. The HC may allocate TXOPs to itself to initiate MSDU deliveries whenever it requires, after detecting the medium as being idle for PIFS, and without backoff. In order for that the HC can give higher priority over the 802.11 DCF and EDCA access, AIFSN[AC] must be selected such that the earliest medium access for EDCA stations is DIFS for any AC.

During the contention period, each TXOP of an 802.11e station begins either when the medium is determined to be available under the EDCA rules, i.e., after AIFS[AC] plus the random backoff time, or when backoff entity receives a polling frame, the QoS CF-Poll, from the HC. This polling frame can be transmitted after a PIFS idle period, without the use of any backoff. During the CFP, the starting time and the maximum duration of each TXOP is determined by the HC using the QoS CF-Poll frames. During the contention free period the 802.11e backoff entities will not attempt to access to the medium without being explicitly polled, so, only the HC can allocate TXOPs by transmitting QoS CF-Poll frames or by immediately transmitting downlink data. During a polled TXOP, the station can transmit the frames selected by its scheduling algorithm, separated by a SIFS time gap, always under the TXOP<sub>limit</sub>.

Polled TXOP allocations may be delayed by the duration of an EDCA-TXOP. The HC controls the maximum duration of EDCA-TXOPs within its QBSS by announcing the TXOP<sub>limit</sub>[AC] for every AC via the beacon. But it is possible to allocate polled TXOPs at any time during the CP, and the optional CFP.

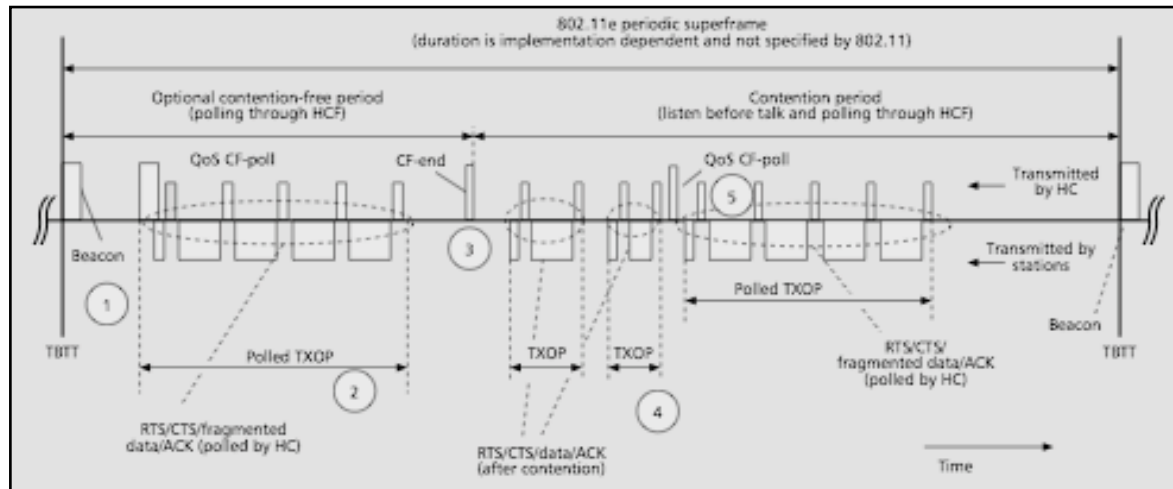


figure 3.5. An example of an 802.11e superframe where the HC grants TXOPs in contention free period.

The figure 3.5 shows an example of a superframe that includes a CFP and a CP. The superframe starts with a beacon transmitted by the HC (on the image 1). During the CFP, the backoff entities only transmit upon being polled by the HC. (On the image 2) Here we remark the transmission of an MSDU fragmented within the CFP. In order to finalize the CFP the HC transmit a CF-end frame (on the image 3). In the next part of the frame starts the contention free period, so the backoff entities attempt to transmit using the contention-based medium access. The EDCA-TXOPs are obtained through contention. Two of them are remark in the figure (on the image 4). Also during the CP, the HC can poll a station. Following the EDCA-TXOPs the HC allocates a station with a polling-based manner (on the image 5), that is a fragmented MSDU. One can observe that in every transmission is present the RTS/CTS system.

This method allows to improve the performance of EDCA, that is not sufficient to provide effective traffic protection and QoS guarantees, especially under high traffic loads. In order to be included in the polling list of the HC, a QSTA must send a QoS reservation request using the special QoS management frame, and each individual flow needs one particular reservation request. Figure 3.6 shows a common frame format for carrying traffic specification (TSPEC) parameters. Normally TSPEC parameters include:

- **Mean data rate ( $\rho$ ):** the average bit rate for packet transmission, in bits per second
- **Delay bound ( $D$ ):** the maximum delay allowed to transport a packet across the wireless interface in milliseconds
- **Maximum service interval ( $SI_{max}$ ):** the maximum time allowed between neighbor TXOPs allocated to the same station, in microseconds
- **Nominal MSDU size ( $L$ ):** the nominal packet size in octets
- **Minimum PHY rate ( $R$ ):** the minimum physical bit rate assumed by the scheduler for calculating transmission time in bps

Element ID	Length	TS Info	Nominal MSDU Size	Maximum MSDU Size	Minimum Service Interval	Maximum Service Interval	Inactivity Interval	Suspension Interval
Service Start Time	Minimum Data Rate	Mean Data rate	Peak Data Rate	Maximum Burst Size	Delay Bound	Minimum PHY Rate	Surplus Bandwidth Allowance	Medium Time

*figure 3.6. common management frame format for traffic specification*

One can see that, although the QAP decision is based on the characteristics of individual flows, the HCCA TXOPs are actually assigned on a per QSTA basis instead of per flow, so it is each QSTA the responsible of allocating the TXOPs to its individual flows.

### **3.2.3. Another improvements**

Also, a part from the EDCA and HCCA access medium protocol, we can talk about some other schemes that allow to improve the efficiency of the MAC protocol, and also are part of the 802.11e.

#### **3.2.3.1. Block Acknowledgment**

With the optional block acknowledgment, the throughput efficiency of the protocol is improved. This allow a backoff entity to deliver a number of MSDUs being delivered consecutively during one TXOP and transmitted without individual ACK frames. The MPDUs transmitted during the TXOP are referred to as a block of MPDUs. At the end of the block, or in a later TXOP, all MPDUs are acknowledged by a bit pattern transmitted in the block acknowledgment frame, thus reducing the overhead of control exchange sequences to a minimum of one acknowledgment frame per number of MPDUs delivered in a block.

#### **3.2.3.2. Direct Link Protocol**

Any backoff entity can directly communicate with any other backoff entity in a QBSS, without communicating via the AP. In the legacy 802.11 protocol, within an infrastructure-based BSS (which is denoted as BSS), all data frames are sent to the AP, and received from the AP. This, however, consumes at least twice the channel capacity compared to the direct communication. Only in an independent BSS (which is denoted IBSS), station-to-station communication is allowed in the legacy protocol, due to the absence of the AP. The direct communication in 802.11e is referred to as direct link(DiL). A setup procedure, the Direct Link Protocol (DLP), is defined to establish a DiL between 802.11e backoff entities.

## 4. The *Network Simulator*

The network simulator is a simulator of discrete events and its simulations are at packet level. Its main characteristics are that it is open source, it is capable of doing scheduling, routing and congestion control, and it is also capable to simulate wired networks, like P2P links and LAN, and wireless networks, like ad-hoc, GPRS, UMTS, WLAN, Bluetooth and satellite ones.

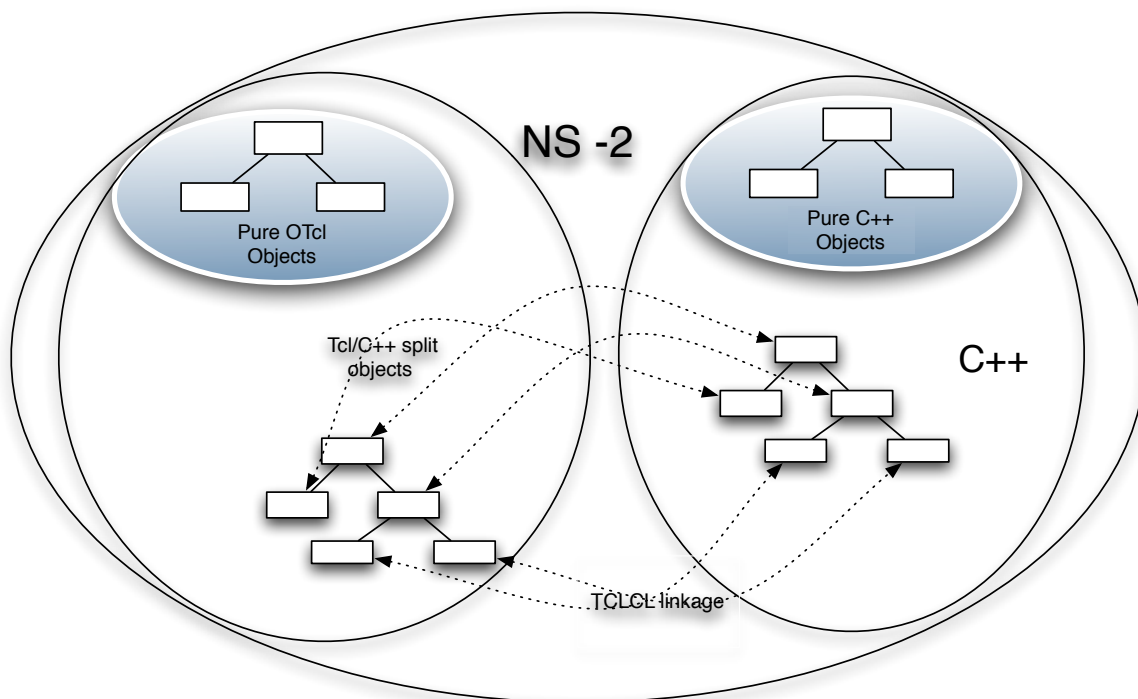
The ns simulator covers a very large number of applications, of protocols, of network types, of network elements and of traffic models. We call these “simulated objects”.

It was born 1989 by Cornell and it was known as REAL network simulator. In the beginning it was made to evaluate control strategy of traffic and congestion in a packet-based network. It was written in C language. In 1995, it developed to ns-1 and it was introduced the Tcl / C++ architecture. Finally, in 1996 appeared ns-2 which the object-oriented Tcl (Otc1). After some work groups have developed some wireless extensions.

NS simulator is based on two languages, an object oriented simulator, written in C++, and a OTcl (an object oriented extension of Tcl) interpreter, used to execute user's command scripts. Also ns has a rich library of network and protocol objects. There are two class hierarchies: the compiled C++ hierarchy and the interpreted OTcl one, with one to one correspondence between them.

The compiled C++ hierarchy allows us to achieve efficiency in the simulation and faster execution times. This is in particular useful for the detailed definition and operation of protocols. this allows one to reduce packet and event processing time.

Then in the OTcl script provided by the user, we can define a particular network topology, the specific protocols and applications that we wish to simulate and the form of the output that we wish to obtain from the simulator. The OTcl can make use of the objects compiled in C++ through an OTcl linkage that creates a matching of OTcl object for each of the C++.



*figure 4.1. Structure of languages of ns-2*

NS is a discrete event simulator, where the advance of time depends on the timing of events which are maintained by a scheduler. An event is an object in the C++ hierarchy with a unique ID, a scheduled time and the pointer to an object that handles the event. The scheduler keeps an ordered data structure with the events to be executed and fires them one by one, invoking the handler of the event.

The simulator is structured in some directory that are constructed following a hierarchy that is showed in the following figure.

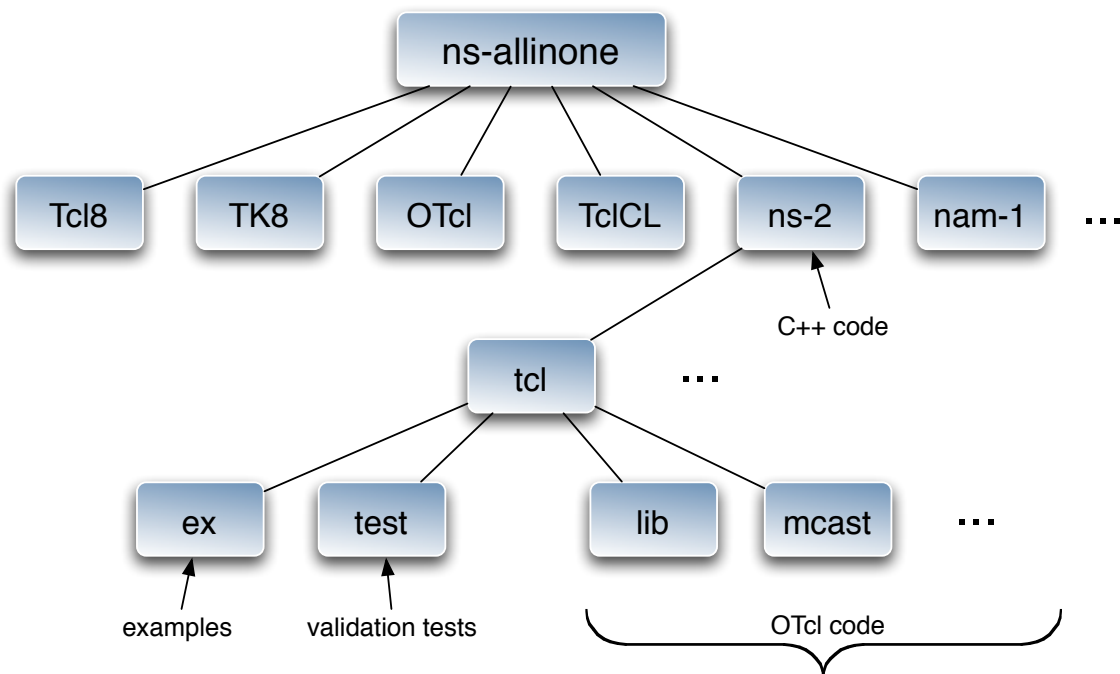


figure 4.2. Directory hierarchy



## **4.1. Satellite networks in ns2**

Network simulator can model, thanks to some extensions, traditional geostationary bent-pipe satellites with multiple users per uplink/downlink and asymmetric links, geostationary satellites with processing payloads, and polar orbiting LEO constellations such as Iridium and Teledesic. With these models one can use ns to study networking aspects of satellite systems, in particular, MAC, link layer, routing, and transport protocols.

### **4.1.1. Geostationary satellites**

Geostationary satellites orbit the Earth at an altitude of 35888 km above the equator. The position of the satellites is specified in terms of the longitude of the nadir point (subsatellite point on the Earth's surface).

Two kinds of geostationary satellites can be modeled. Traditional bent-pipe geostationary satellites are merely repeaters in orbit, so all packets received by such satellites on an uplink channel are piped through at RF frequencies to a corresponding downlink, and the satellite node is not visible to routing protocols. Newer satellites will increasingly use baseband processing both to regenerate the digital signal and to perform fast packet switching on-board the spacecraft. In the simulations, these satellites can be modeled more like traditional ns nodes with classifiers and routing agents.

### 4.1.2. Satellite nodes

There are two basic kinds of satellite nodes, the geostationary and the non-geostationary satellite node. Also exists the figure of terminal nodes, with the possibility to place them in the Earth's surface. These three terminals are actually implemented with the same class SatNode object, but with different position, handoff manager, and link objects attached.

### 4.1.3. Satellite links

Satellite links resemble wireless links. Each satellite node has one or more satellite network interface stacks, to which channels are connected to the physical layer object in the stack. the figure below shows the major components. But satellite links differ from the wireless links in two aspects: the transmit and receive interfaces must be connected to different channels, and there is no ARP implementation.

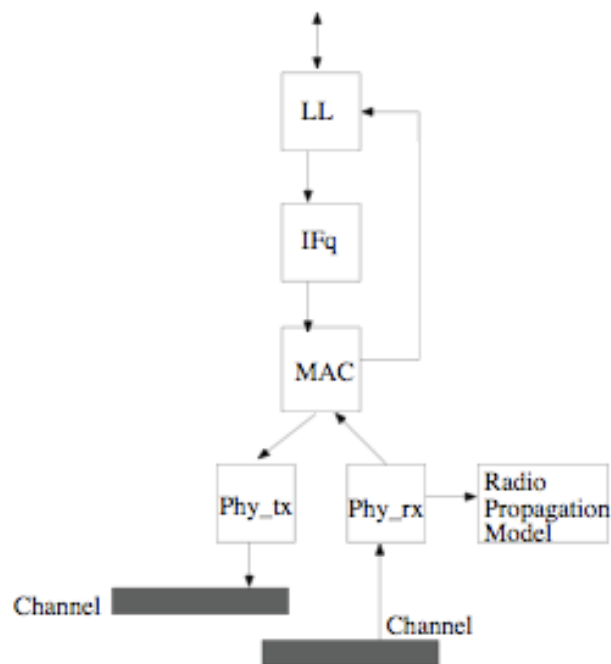


figure 4.3. Main components of a satellite network interface

The interface of satellite node must contain the following parameters:

- *type*: the following link types can be indicated: geo or polar for links from a terminal to a geo or polar satellite, respectively, gsl and gsl-repeater for links from a satellite to a terminal, and intraplane, interplane and crossseam ISLs. the type field is used internally in the simulator to identify the different types of links, but structurally they are all very similar.
- *ll*: the link layer type, that currently the class Ll/Sat is the only defined.
- *qtype*: the queue type (normally DropTail). Any queue type may be used however, if additional parameters beyond the length of the queue are needed, then *instproc* may need to be modified to include more arguments.
- *qlim*: the length of the interface queue in packets.
- *mac*: The MAC type. Currently, two types are defined: class Mac/Sat, that is a basic MAC for links with only one receiver, i.e. without collision detection, and class Sat/Mac/UnslottedAloha.
- *mac\_bw*: The bandwidth of the link is set by this parameter, which controls the transmission time how fast the MAC sends.
- *phy*: the physical layer, currently exists two: class Phy/Sat and class Phy/Repeater are defined. The first just pass the information up and down the stack as in the wireless code, so a radio propagation model could be attached at this point. The other class pipes any packets received on a receive interface straight through to a transmit interface

#### **4.1.4. Interconnection of wired and wireless networks**

It is important also to interconnect a wired node with the satellites nodes in order to increase the scenarios of study.

The satellite code normally performs all routing in C++, while the traditional ns code uses a mix of OTcl and C++ code. For backward compatibility reasons, it is difficult to fully integrate both wired and wireless code. The strategy for integrating wireless and wired code has been to define a special gateway node, to use hierarchical routing, and to locate a single base station node in the wireless network with the network stack located in both the wireless and the wired subnet.

### **4.2. Mobile networks in ns2**

The initial wireless model of ns2 was ported as CMU's Monarch group's mobility extension to ns. In a first moment this extension covered the internals of a mobile node, routing mechanisms and network components. The components that were introduced are Channel, Network interface, radio propagation model, MAC protocols, interface queue, link layer and address resolution protocol model (ARP). But this initial extension only covers the simulation of pure wireless networks. Further extensions make possible the union between wired and wireless networks.

The hierarchy of an object that forms part of wireless network is the shown in the following figure.

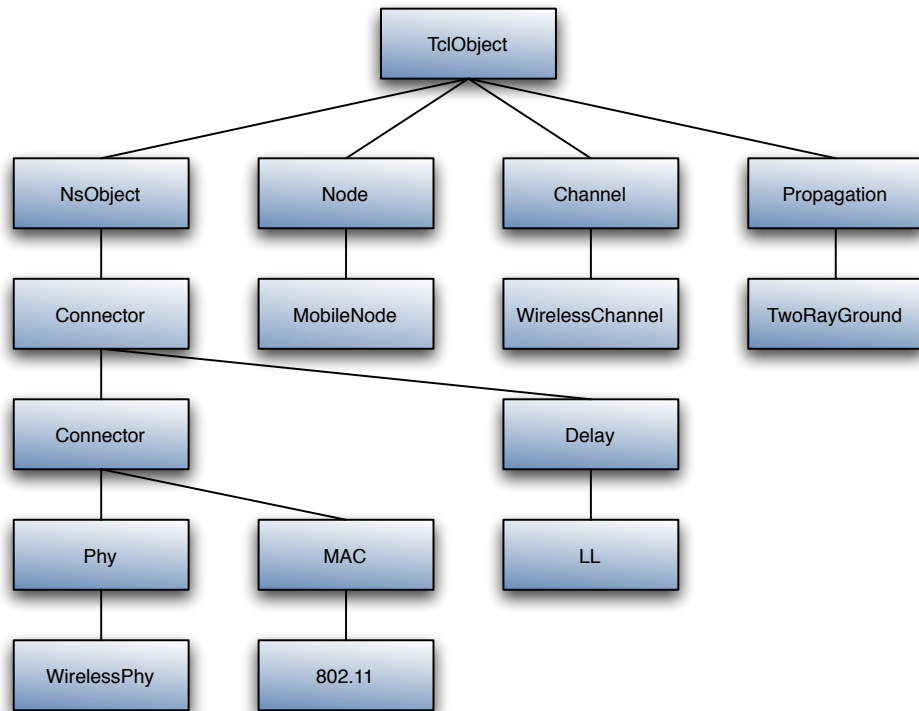


figure 4.3. wireless hierarchy

The components that forms, the main of them, are described below:

- *Link layer*: The LL used by mobile node has an ARP module connected to it which resolves all IP to hardware (Mac) address conversions. Normally for all outgoing (into the channel) packets, the packets are handed down to the LL by the Routing Agent. The LL hands down packets to the interface queue. For all incoming packets the mac layer hands up packets to the LL which is then handed off at the node\_entry\_point.
- *ARP*: The Address Resolution Protocol module receives queries from Link layer. If ARP has the hardware address for destination, it writes it into the mac header of the packet. Otherwise it broadcasts an ARP query, and caches the packet temporarily. For each unknown destination hardware address, there is a buffer for a single packet. In case additional packets to the same destination is sent to ARP, the earlier buffered packet is dropped. Once the hardware address of a packet's next hop is known, the packet is inserted into the interface queue.

- *Interface queue:* The class PriQueue is implemented as a priority queue which gives priority to routing protocol packets, inserting them at the head of the queue. It supports running a filter over all packets in the queue and removes those with a specified destination address.
- *Mac layer:* The IEEE 802.11 distributed coordination function (DCF) Mac protocol has been implemented by the CMU. It uses a RTS/CTS/DATA/ACK pattern for all unicast packets and simply sends out DATA for all broadcast packets. The implementation uses both physical and virtual carrier sense.
- *Network interfaces:* The Network Interphase layer serves as a hardware interface which is used by mobile node to access the channel. The wireless shared media interface is implemented as class Phy/WirelessPhy. This interface subject to collisions and the radio propagation model receives packets transmitted by other node interfaces to the channel. The interface stamps each transmitted packet with the meta-data related to the transmitting interface like the transmission power, wavelength, .. This meta-data in pkt header is used by the propagation model receiving network interface to determine if the packet has minimum power to be received and/or captured and/or detected (carrier sense) by the receiving node.
- *Radio propagation model:* It uses Friss-space attenuation at near distances and an approximation two-ray ground at far distances. the approximation assumes specular reflection off a flat ground plane.
- *Antenna:* An omni-directional antenna having unity gain is used by mobile nodes.

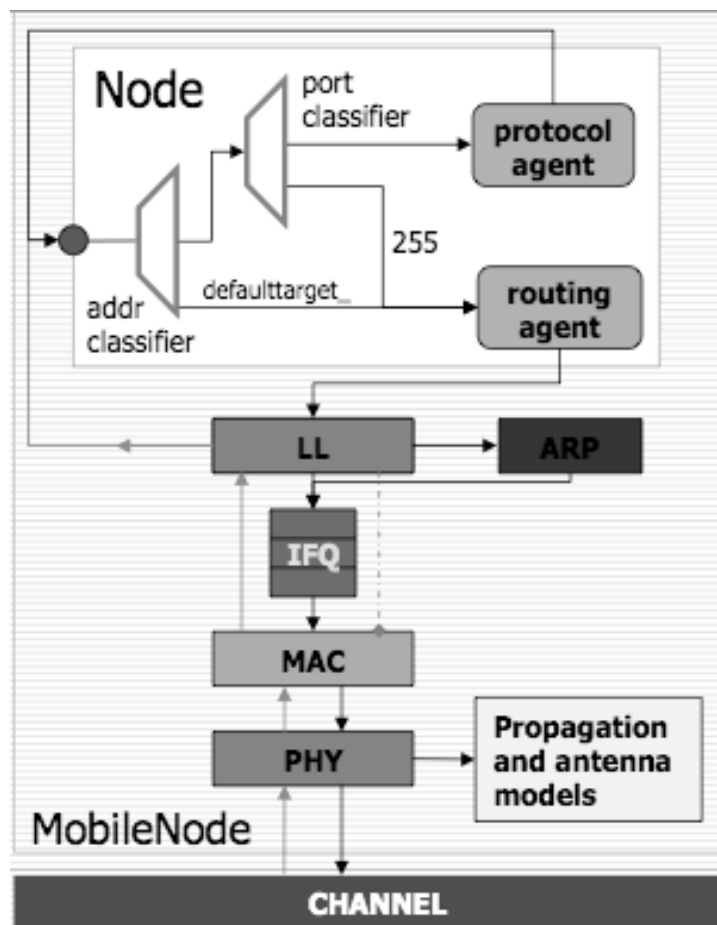


figure 4.4. block scheme of a mobile node in ns-2

# 5. Simulation results

In this chapter we want to show the simulation obtained with the network simulator. In a first moment the objective was to implement a scenario involving the satellite part and the wireless part.

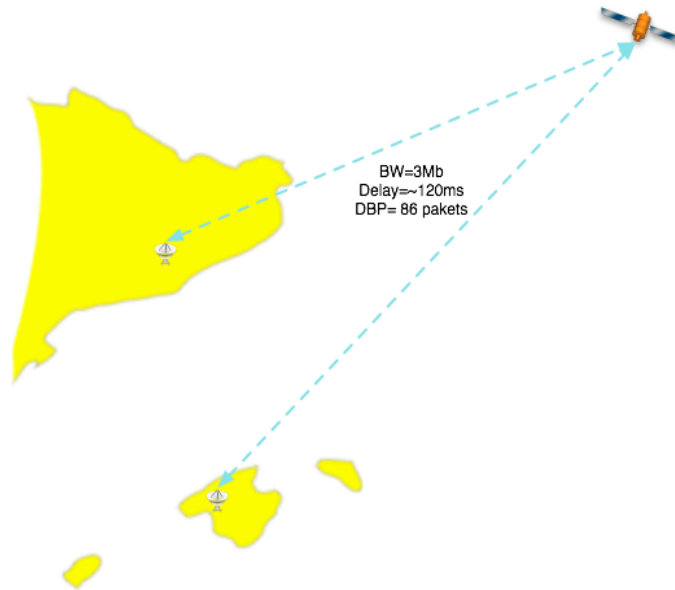
Being the first time using this simulator, a major part of the work was to achieve a minimum knowledge of the simulator performance. So the results shown in this chapter reflects the improves achieved during the realization of this research project.

The first part being simulated was the satellite. Using a geostationary satellite and introducing two satellite terminals, the study done was the difference between a wired node and the satellite network. How affects the use of satellite in the throughput link.



## 5.1. Satellite scenario

In this first part of simulations the scenario on we move is the showed below. One can see a geostationary satellite connecting two earth terminals. The parameters used in the simulations are shown in the images following.



*figure 5.1. Scenario used (global vision)*



*figure 5.2. Detail of the scenario used (satellite terminals vision)*

In the satellite part we are going to study the different behavior of three types of TCP, and also the importance of the buffer queue in a bottleneck satellite network. This would be reflected in the throughput achieve in the system and also in the congestion window of the satellite link.

We modify the queue parameter in order to see the effect in this kind of networks the values used are 50, 85 and 125<sup>1</sup>. Always being under the value of the sum of the receiver window and the delay bandwidth product for make the simulation independent of the receiver window value.

In the congestion window we see the following results:

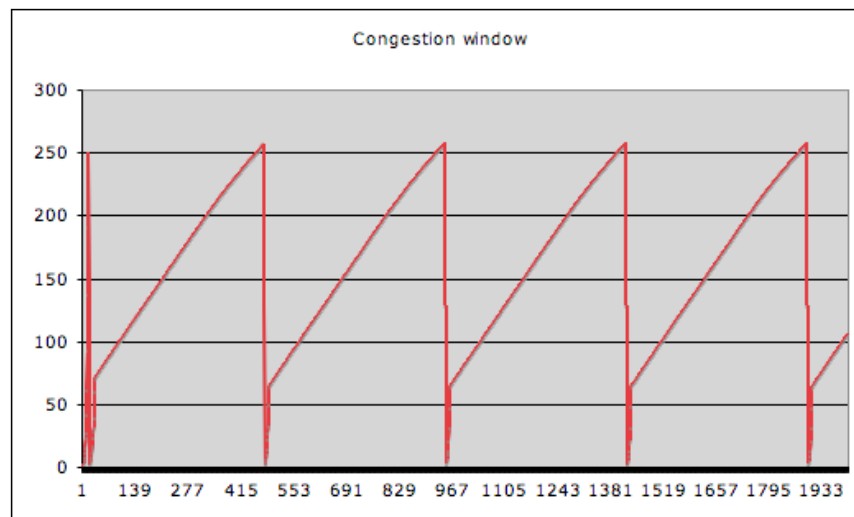
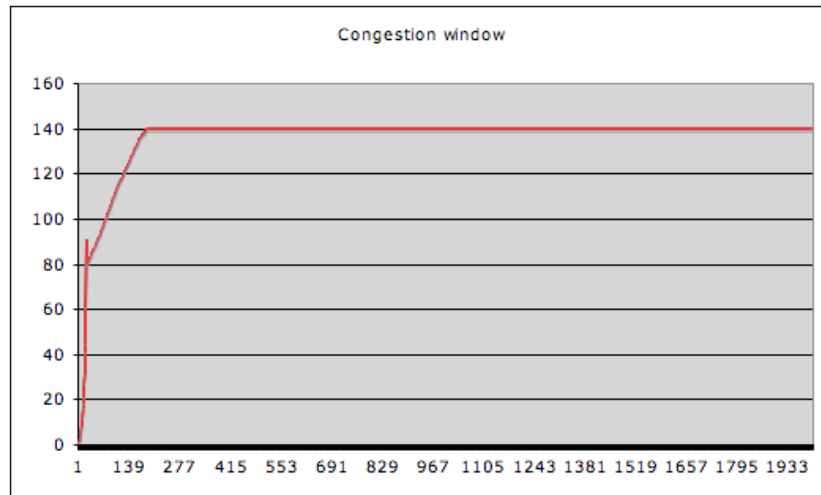
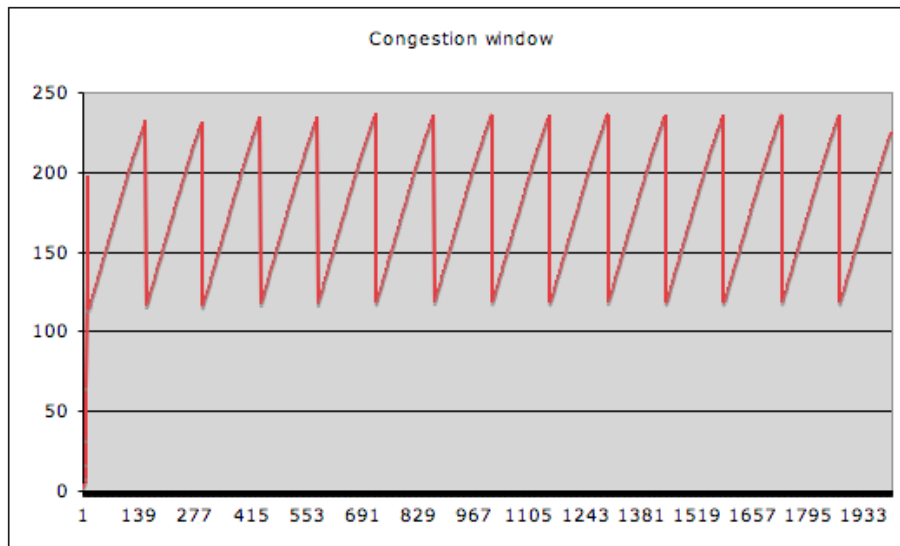


figure 5.3. cwnd of TCP New Reno

<sup>1</sup> these values are the same used in an EDCA simulation perform in Siena university



*figure 5.4. cwnd of TCP Vegas*



*figure 5.5. cwnd of TCP Sack*

If we observe the congestion window (cwnd) of New Reno and Sack we can see the two phases defined by the TCP congestion avoidance. In New Reno, first one can see an exponential increase, followed by a linear increase until the ssthresh value. When a loss is performed the cwnd returns to its initial value. In TCP Vegas the cwnd remains constant, with a previously oscillation of its value, achieving a top value of 140. And for TCP Sack we can see that a loss is being deployed returning to the size of the cwnd, making only the linear increase.

For the different values for the queue buffer the results show very low differences between them. Only we can say that the best result is achieved by the TCP Sack, thanks that it works in a better form with a great value of the receiver window. In this case is of 500 packets. So as we want the TCP Sack obtain a better performance, as we can see in the following figures.

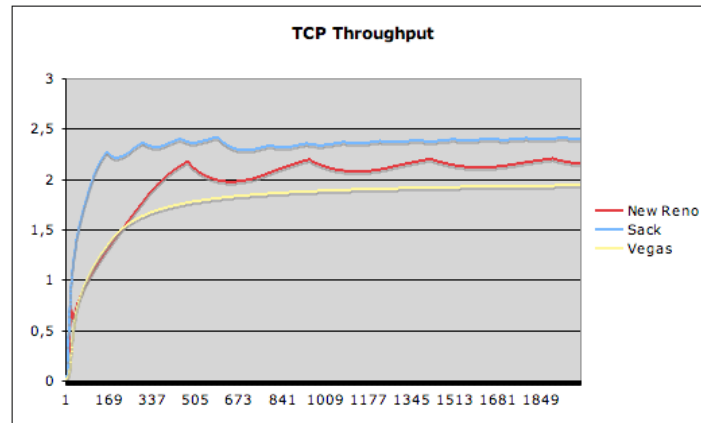


figure 5.6. Queue buffer of 50 packets

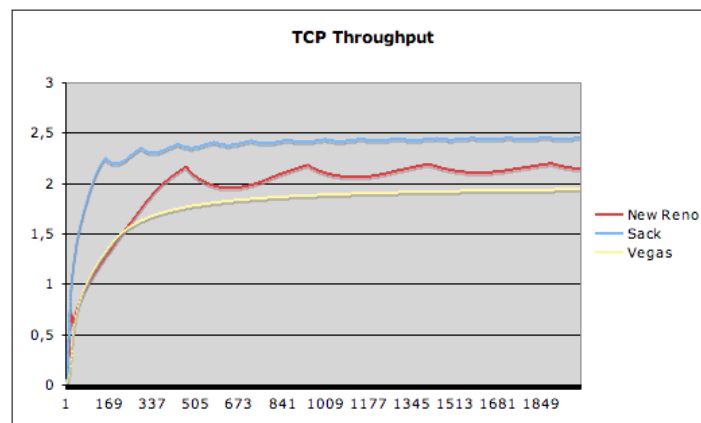


figure 5.7. Queue buffer of 85 packets

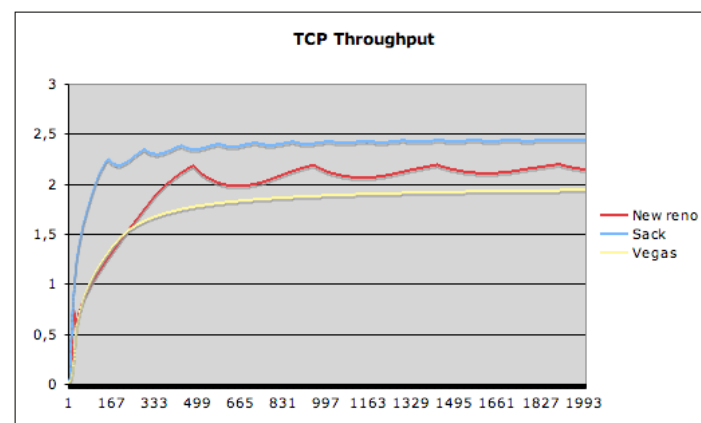


figure 5.7. Queue buffer of 125 packets

The last part of simulations is using a double connection between the satellite link. As is shown in the following image we have defined two TCP senders, and two TCP receivers, using TCP New Reno, using the same satellite link.

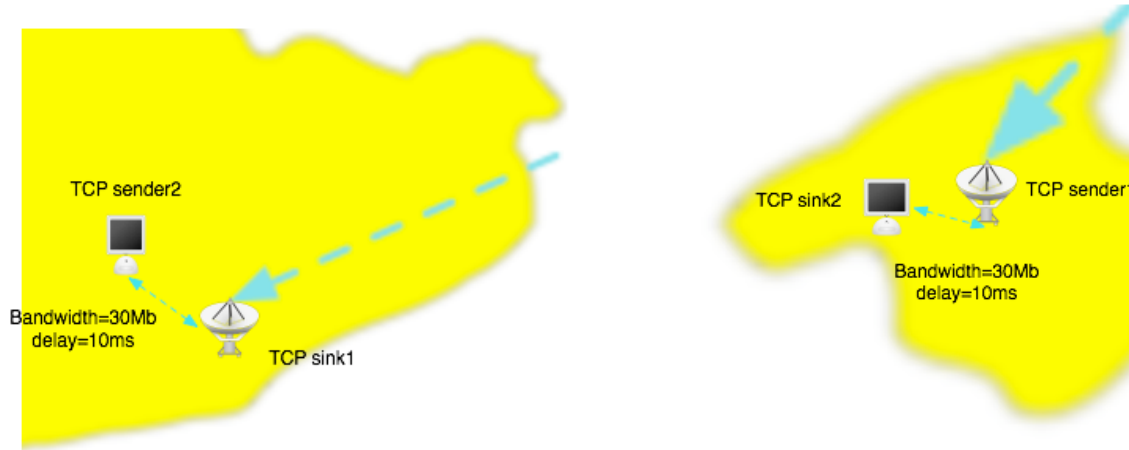


figure 5.9. Detail on the scenario

We have two traffic links, the one that involve the satellite terminals and the one that involves the wired nodes.

In this case we will see the effect of having different receiving windows. Their values are set to: 50, 100, 150 and 200 packets. The results obtained are shown below:

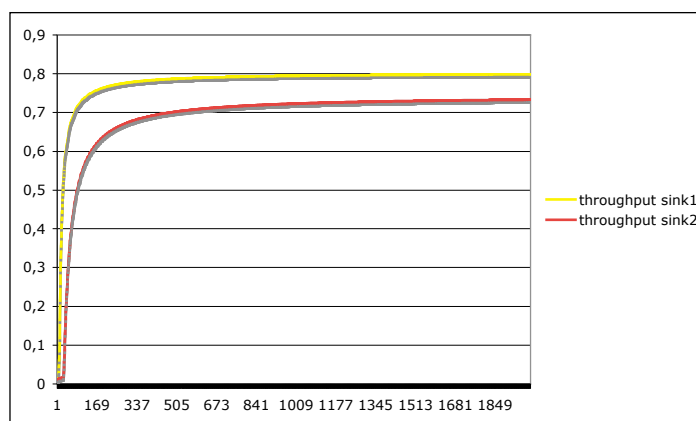


figure 5.10. rwnd of 50 packets

The fact that we achieve a constant throughput is due to the use of a rwnd lower than the Delay Bandwidth Product.

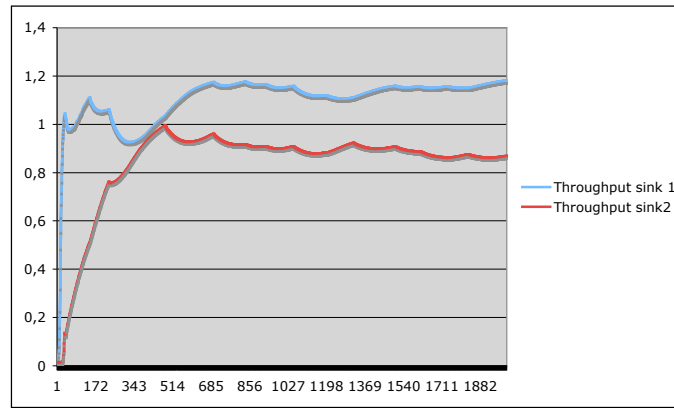


figure 5.11. *rwnd 100 packets*

The maximum Throughput achieved is about the 1,2 Mbps. The goodput normalized achieved by the TCP of sink1 (received in the satellite terminal) has a value of 39,4%, and the achieve by the other TCP connection is 29,4%. So the parameters used are insufficient to giving a good performance.

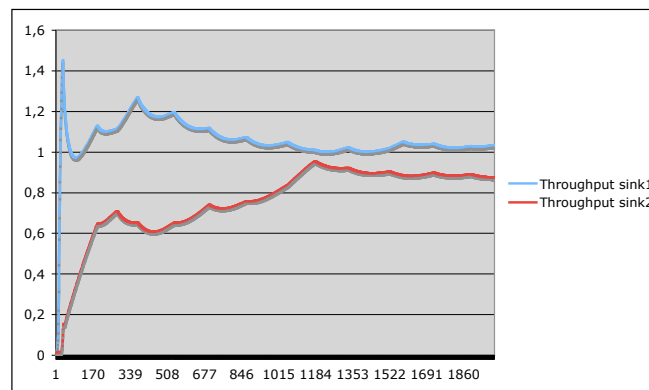
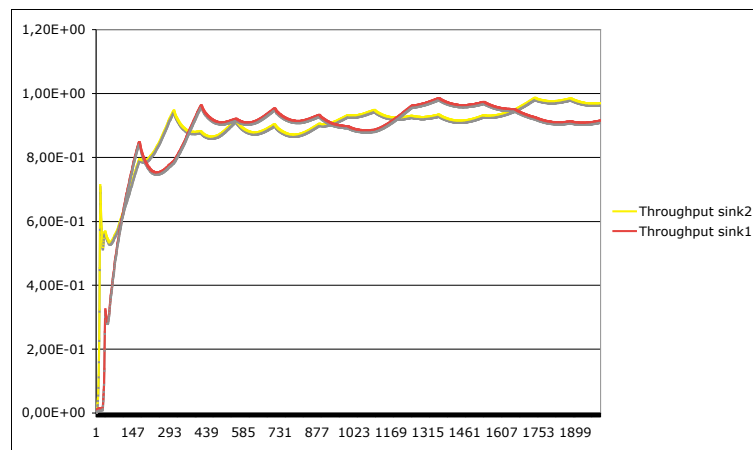


figure 5.12. *rwnd 150 packets*

In this case the goodput achieve by the connections are: 34,5% and 29,2%. As one can see the improvement is not as good as one could want.



*figure 5.13. rwnd 200 packets*

Now we achieved a similar throughput in both connections although its values are below the 1Mbps. the goodput values are: 32,2%, for the sink1, and 30,5% for the sink2. We continue seeing that these chosen values are not as good as one want. So it is necessary to find a better set of parameters to increase and improve the satellite performance with satellite TCP connections.

## 5.2. Wireless simulations

The wireless part is focus on the HCCA patch provided by the Pisa University. In a first moment we started to study the effect of the increasing users that use the VoIP service. So if the number of users that use the real time service what will pass in the network performance. This scenario also have users that transmit FTP packets using TCP protocol, while the CBR packets are transported via UDP protocol (VoIP service).

This last packets are prioritized over the other packets of the TCP. The patch allow to transmit together 9 users. We tried to discover what makes this limitation. we try to modify the C++ codes but we couldn't find the form to improve that.

Another point is the change of physical parameters of the simulation. This is another fact that didn't allow us to perform more kinds of simulations. The different codes used to implement the HCCA patch are complex and their study is complex.

The HCCA can give priorities to the stations via the TSPEC contained in the frames. In order to achieve some different it could be important to change them. But changing from the initial value makes impossible to achieve a correct simulation. For that reason the simulations present here are done varying the number of users that transmit real-time packets.

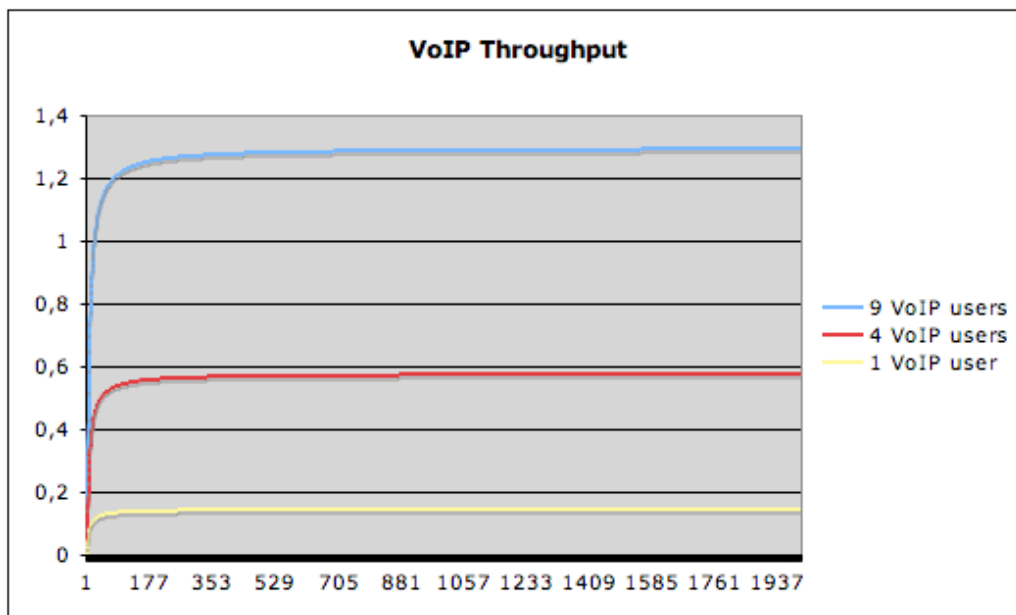


figure 5.2. VoIP throughput with different number of users

In this simulation we can observe that increasing the number of users allow to achieve a higher throughput of the system. As we expected the throughput achieve is constant during the time simulation due to the service required is a constant bit rate, as the user expect.

On the other hand the increase of real-time stations makes decrease the system throughput in the TCP users, as we can see in the following figure. In the set of figures we can see the differences between three kind of TCP: New Reno, Vegas and Sack. The differences between this TCP protocols are explained in a previous section of this work.



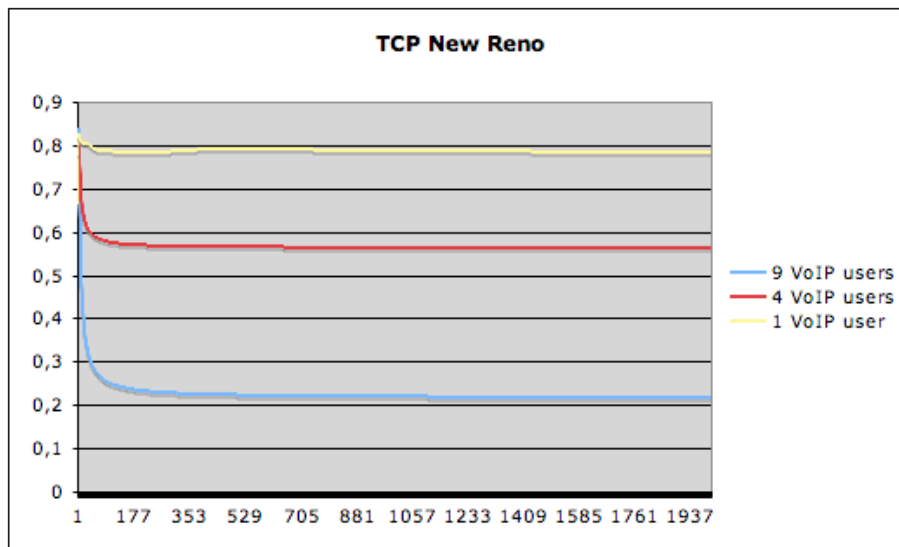


figure 5.3. Throughput of TCP New Reno

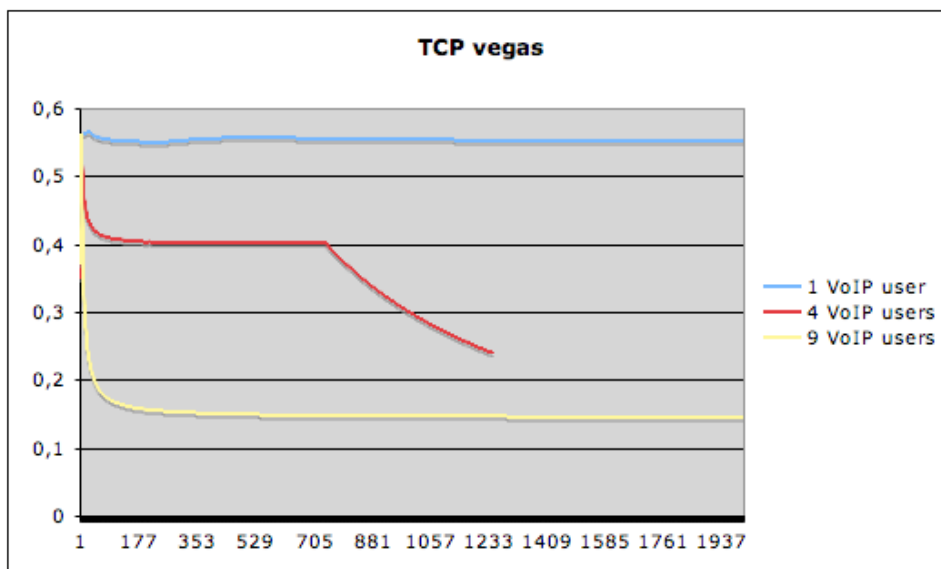
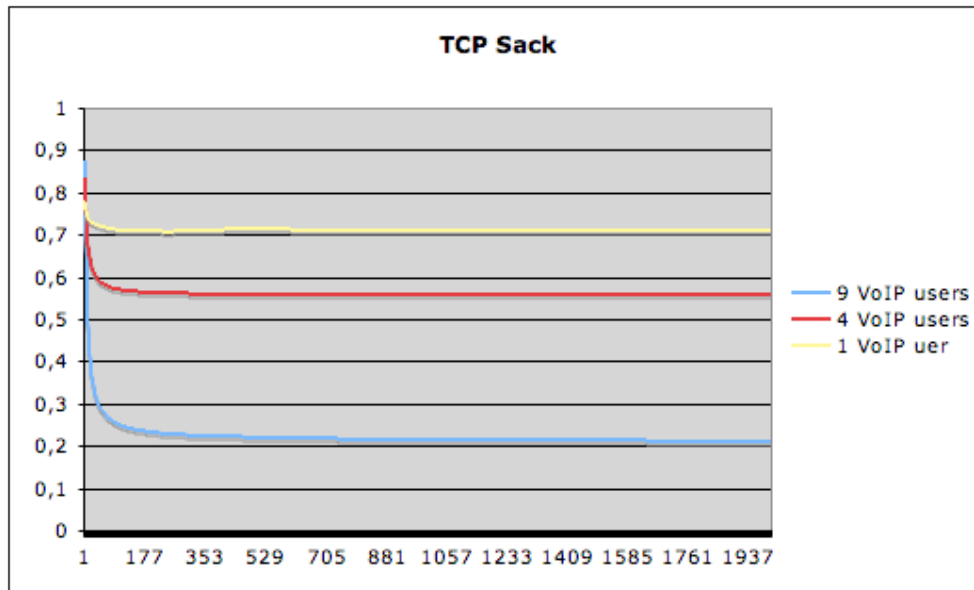


figure 5.4. Throughput of TCP Vegas



*figure 5.5. throughput of TCP Sack*

As one can see in the figures, the effect is clear. Increase the number that transmit in a CBR way makes decrease the throughput performance of the TCP packets. We can remark that the best TCP option is the New Reno as we said before, this TCP is the best option and the most common.

After the simulation of this scenario, the main issue for us is to achieve the interconnection of a satellite network with a wireless network based on HCCA. But unfortunately it can not be possible. In a first moment with the investigation equip of Siena we try to achieve this interconnection with the EDCA 802.11e enhancement. This is also an issue, although they could achieve some result is also a problem to solve.



## 6. Conclusions

As we have remark during all this work, the satellite and the wireless networks are the nowadays networks that makes to increase the market in communications. First for it is capacity to give a large communication without the needed of having a terrestrial architecture, that in some cases it is impossible. The second network is capable to offer service to a group of users without involving any cable. Giving mobility to its users.

But this is not sufficient nowadays. Many applications are appearing in market and these applications requires some parameters that the initials satellite and wireless networks couldn't give.

For that reason, techniques like the cross layer, allows to increase and improve the performance of the satellite link. Giving QoS using the DiffServ scheme allows the actual satellite networks the possibility of allocate users that requires some specifications on their traffic and not only giving a simple transmission.

In the other hand the 802.11e appears, finally in 2005, to achieve the same intention as we described in the above paragraph. Giving QoS in the wireless networks. It introduces improvements to the existent resources giving the opportunity of allowing the users to

transmit any kind of application. All of those that requires traffic specifications and priority respect another applications.

All this research work developed in this document has been focus on simulating this QoS achievements using the open simulator ns-2. During the research on papers for achieving a better acknowledgment on the QoS giving in both networks, also it was necessary achieving acknowledges in this simulator.

Thanks to the patch of the university of Pisa we can initiate the firsts simulations on the HCCA part. We can do some simulations on this standard but not all that was thought.

So in order to remark future investigations in these fields we can say that it can be possible to continue the research on the behavior of the HCCA and satellite network working together. Under the ns-2 simulator the work is initiated by the university of Pisa and also the University of Siena that are realizing a research using the same simulator on interconnection the satellite network with the EDCA wireless network.

## 7. Bibliography

- (I) S. Mangold *et al.*, “Analysis of IEEE 802.11e for QoS support in Wireless LANs”. IEEE Wireless Communications, December 2003 pp 40-50
- (II) - “IEEE 802.11e Wireless LAN for Quality of Service”.
- (III) Yang Xiao, “An Analysis for Differentiated Services in IEEE 802.11 and 802.11e Wireless LANs”. Proc of the 24th International Conference on Distributed Computing System (ICDCS’04)
- (IV) Qiang Ni, *et al.* “Journal of Wireless Communications and Mobile Computing”, Wiley. 2004: Volume 4, Issue 5: pp.547-566
- (V) Giovanni Giambene *et al.* “Optimization of IEEE 802.11e Access Class Parameters with Cross-layer Approach”.
- (VI) “SatLab System recommendation v2 QoS”, <http://satlabs.org/content/view/87/79>. November 2006
- (VII) Paolo Barsocchi *et al.* “Radio Resource Management Across Multiple Protocol layers in Satellite Networks: a Tutorial Overview” Satnexus NoE Project
- (VIII) Alberto Gotta *et al.* “Simulating dynamic bandwidth allocation on satellite links” 5th Research framework Programme (SatNexus IST-507052)
- (IX) Ns2 simulator, available online- <http://www.isi.edu/nsnam/ns>
- (X) Harald Skinnemoen *et al.* “VoIP with QoS and bandwidth-on-demand for DVB-RCS”
- (XI) R.J. Mort *et al.* “ QoS architecture for Broadband Satellite Multimedia (BSM) networks”.
- (XII) Claudio Cicconetti *et al.* “A software architecture for simulating IEEE 802.11e HCCA”. 3rd Internet Performance, Simulation, Monitoring and Measurement, IPS-MoMe 2005, March 14-15 2005
- (XIII) IEEE Computer Society LAN/MAN standards committee. “IEEE 802.11e: Wireless LAN Medium Access Control and Physical Layer specifications. November 2005.



## **Abstract**

This project is based on the study of offering quality of service in satellite and wireless networks. For that reason the study of the techniques of cross-layer and the IEEE 802.11e has been the key point for the theoretical development of this study.

Using the software *network simulator*, in the part of simulations three situations consider: the study of the satellite network, the study of the access method HCCA and the interconnection of the satellite network with the wireless. Although this last point, incomplete in this work, must be the continuation for future investigations.

## **Resum**

Aquest projecte es basa en l'estudi de l'oferiment de qualitat de servei en xarxes wireless i satelital. Per això l'estudi de les tècniques de cross-layer i del IEEE 802.11e ha sigut el punt clau per al desenvolupament teòric d'aquest estudi.

Usant el simulador de xarxes *network simulator*, a la part de simulacions es plantegen tres situacions: l'estudi de la red satelital, l'estudi del mètode d'accés HCCA i la interconnexió de la red satelital amb la wireless. Encara que aquest últim punt, incomplet en aquest projecte, ha de ser la continuació per a futures investigacions.

## **Resumen**

Este proyecto se basa en el estudio del ofrecimiento de calidad de servicio en redes wireless y satelitales. Por eso el estudio de las técnicas de cross-layer y del IEEE 802.11e ha sido el punto clave para el desarrollo teórico de este estudio.

Usando el simulador de redes *network simulator*, en la parte de simulaciones se plantean tres situaciones: el estudio de la red satelital, el estudio del método de acceso HCCA y la interconexión de la red satelital con la wireless. Aunque este último punto, incompleto en este proyecto, tiene que ser la continuación para futuras investigaciones.