

HCCA- A QOS SUPPORT STANDARD IN IEEE 802.11e WIRELESS LANS

**A DISSERTATION
SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE AWARD OF THE DEGREE OF**

**MASTER OF ENGINEERING
IN
COMPUTER TECHNOLOGY & APPLICATIONS**

by

Vamshi Krishna Palakurthy

College Roll No. 16/CTA/04

Delhi University Roll No. 8518

**Under the Guidance of
Prof. Asok De**



**DEPARTMENT OF COMPUTER ENGINEERING
DELHI COLLEGE OF ENGINEERING
UNIVERSITY OF DELHI
JUNE 2006**

ACKNOWLEDGEMENT

It is a great pleasure to have the opportunity to extend my heartiest felt gratitude to everybody who helped me throughout the course of this project. I would like to express my heartiest felt regards to **Dr. Asok De**, Professor, Department of Computer Engineering for the constant motivation and support during the duration of this project. It is my privilege and honour to have worked under his supervision. His invaluable guidance and helpful discussions in every stage of this project really helped me in materializing this project.

I would like to thank **Dr. Goldie Gabrani**, Head of the department, Computer Engineering, for providing facilities for this project.

I would also like to take this opportunity to present my sincere regards to my department faculty Professor **D. Roy Chowdhary, Dr. S. K. Saxena, Mr. Rajeev Kumar** and **Mrs. Rajni Jindal** for their support and encouragement.

I am thankful to my friends and classmates for their unconditional support and motivation during this project.

Vamshi Krishna Palakurthy

M.E. (Computer Technology & Applications)

College Roll No. 16/CTA/04

Delhi University Roll No. 8518

CERTIFICATE

This is to certify that the work that is being presented in this project entitled “*HCCA – A QOS SUPPORT STANDARD IN IEEE 802.11e WIRELESS LANS*”, in partial fulfillment of the requirement for the award of the degree of Master of Engineering in Computer Technology and Application submitted by *Vamshi Krishna Palakurthy (16/CTA/04)* to the Department of Computer Engineering, Delhi College of Engineering, is an authentic record of the student’s own work carried out under the supervision and guidance of **Prof.Asok De**, in the Department of Computer Engineering.

Dr. Asok De

(Professor)

Dept. of Computer Engineering

Delhi College of Engineering

ABSTRACT

The most popularly and widely used Wireless LAN standard all over the world today is IEEE 802.11. It is likely to play a major role in the next generation wireless communication networks. In the present 802.11 Wireless LANs two basic co-ordination functions exists such as DCF (Distributed Coordination function) and PCF (Point Coordination Function). DCF is responsible for asynchronous data services and the PCF was developed for time bounded services. Unlike traditional best effort data applications, multimedia applications require quality of service (QoS) support such as guaranteed bandwidth and bounded delay/jitter. Providing such QoS support in 802.11 is challenging and thus a number of IEEE 802.11 QoS enhancement schemes have been proposed. In this present work we analyze the QoS limitations of IEEE 802.11 wireless MAC layers with present protocol PCF. And we suggest the QoS Enhancement techniques for IEEE 802.11 WLAN which is Reference Scheduler of the HCF Controlled Channel Access (HCCA) Protocol referred in IEEE 802.11e standard.

The Simulation and thus Performance of the HCCA protocol with Reference Scheduler is realized using *Network Simulator 2* (ns2). The basic structure of ns-2 and the networking protocols are realized in the programming language C++. The script language Tcl is used for easy control and assembly of new simulations. The generated trace files are to be analyzed using perl script and the graphs are plotted using xgraph/gnu plot.

Table of Contents

1. Introduction.....	1
1.1 How Wireless LANs work.....	1
1.2 Applications for Wireless LANs.....	3
1.3 Benefits of WLANs	5
1.4 Issues with Wireless LANs.....	6
2. Overview of IEEE 802.11 WLANs	8
2.1 What is IEEE 802.11?.....	8
2.2 Components of an 802.11 WLAN	8
2.3 Type of Networks.....	10
2.4 Original 802.11 MAC	13
2.5 Inter Frame Spacing.....	14
2.6 Overview of Distributed Coordination Function (DCF).....	15
2.7 Overview of Point Coordination Function (PCF).....	16
3. Point Coordination Function (PCF)	17
3.1 Reserving the medium during the contention-free period	18
3.2 The polling list	18
3.3 Transmissions from the Access Point	19
3.4 Contention-Free Period Duration.....	23
3.5 Power Management and the PCF.....	23
3.6 PCF Operation – Summary	24
3.7 Problems with PCF	25
3.8 Suggestions to improve the IEEE 802.11 PCF scheme	27
4. HCCA- QoS standard for WLans.....	32
4.1 Hybrid Control Function (HCF)	33
4.2 Enhancements of 802.11e over 802.11	34
4.3 Overview of Enhanced Distributed Coordination Access (EDCA).....	37
4.4 Overview of HCF Controlled Channel Access (HCCA)	38
4.5 HCF Controlled Channel Access HCCA).....	38
4.5.1 The Hybrid Controller (HC).....	38
4.5.2 Relationship between CFP, CP, CAP & CC.....	39
4.5.3 Understanding the HCF superframe.....	41
4.6 Traffic Scheduling in IEEE 802.11e.....	43
4.7 Simple scheduler	45
4.8 HCCA Reference Scheduler.....	47

5. HCCA Network Simulator	52
5.1 Introduction.....	52
5.2 General Architecture	52
5.3 Example Simulation Script.....	55
5.4 MAC Layer.....	58
5.5 NS2 HCCA Class Diagram.....	59
5.6 Events used in HCCA.....	61
6. Simulation Results	63
Conclusion and Future Scope	67
Appendix A	68
References	72
List of Abbreviations	74

List of Figures

FIGURE NAME	PAGE NO.
Figure 1 : Components of an 802.11 wireless network	8
Figure 2 : Independent BSS	10
Figure 3 : Infrastructure BSS	12
Figure 4 : Infrastructure BSS (distributed)	13
Figure 5 : 802.11 MAC architecture	14
Figure 6 : 802.11 Superframe	17
Figure 7 : PCF Operation	18
Figure 8 : Example of Data + CF-Poll, Data + CF-ACK, CF-ACK + CF-Poll	21
Figure 9 : Example of Data + CF-ACK + CF-Poll	22
Figure 10 : Unpredictable Beacon Delay	26
Figure 11 : Typical Scenario without using More Data Bit	30
Figure 12 : After using More Data Bit	30
Figure 13 : MAC architecture in 802.11e	34
Figure 14 : Mapping User priorities (UP) to Access Categories (AC)	35
Figure 15 : Traffic Differentiation	36
Figure 16 : Relationship of CFP, CP, CAP within one Superframe	40
Figure 17 : Control Contention Interval & the CC frame	41
Figure 18 : 802.11e Superframe	42
Figure 19 : TSPEC elements as defined in IEEE 802.11e	43
Figure 20 : Simple Schedule for a single stream i	46
Figure 21 : Simple Schedule for two streams i and j	46
Figure 22 : Beacon interval for HCF scheduling algorithm	48
Figure 23 : Classification of 802.11 and 802.11e protocols	51
Figure 24 : C++/OTCL duality	53

Figure 25 : Simplified View of NS	54
Figure 26 : Directory Structure of NS	54
Figure 27 : Scenario for simple simulation script	55
Figure 28 : NS2HCCA class diagram	59
Figure 29 : Number of flows vs throughput (for 5 to 8 stations)	64
Figure 30 : Number of flows vs throughput (for 8 to 11 stations)	65
Figure 31 :Number of flows vs throughput (for 5 to 11 stations)	66

Chapter 1

Introduction

A wireless LAN (WLAN) is a flexible data communication system implemented as an extension to or as an alternative for, a wired LAN within a building or campus. Using electromagnetic waves, WLANs transmit and receive data over the air, minimizing the need for wired connections. Thus, WLANs combine data connectivity with user mobility, and through simplified configuration, enable movable LANs.

Over the last ten years, WLANs have gained strong popularity in a number of vertical markets, including the health-care, retail, manufacturing, warehousing, and academic arenas. These industries have profited from the productivity gains of using hand-held terminals and notebook computers to transmit real-time information to centralized hosts for processing. Today WLANs are becoming more widely recognized as a general-purpose connectivity alternative for a broad range of business customers.

1.1 How Wireless LANs work

Wireless LANs use electromagnetic airwaves (radio and infrared) to communicate information from one point to another without relying on any physical connection. Radio waves are often referred to as radio carriers because they simply perform the function of delivering energy to a remote receiver. The data being transmitted is superimposed on the radio carrier so that it can be accurately extracted at the receiving end. This is generally referred to as modulation of the carrier. Once data is superimposed (modulated) onto the radio carrier, the radio signal occupies more than a single frequency.

Multiple radio carriers can exist in the same space at the same time without interfering with each other if the radio waves are transmitted on different radio frequencies. To extract data, a radio receiver tunes in (or selects) one radio frequency while rejecting all other radio signals on different frequencies.

In a typical WLAN configuration, a transmitter/receiver (transceiver) device, called an access point, connects to the wired network from a fixed location using standard Ethernet cable. At a minimum, the access point receives, buffers, and transmits data between the WLAN and the wired network infrastructure. A single access point can support a small group of users and can function within a range of less than one hundred to several hundred feet. The access point (or the antenna attached to the access point) is usually mounted high but may be mounted essentially anywhere that is practical as long as the desired radio coverage is obtained.

End users access the WLAN through wireless LAN adapters, which are implemented as PC cards in notebook computers, or use ISA or PCI adapters in desktop computers, or fully integrated devices within handheld computers. WLAN adapters provide an interface between the client network operating system (NOS) and the airwaves (via an antenna). The nature of the wireless connection is transparent to the NOS.

1.2 Applications for Wireless LANs

Wireless LANs frequently augment rather than replace wired LAN networks providing the final few meters of connectivity between a backbone network and the in-building or on campus mobile user.

The following list describes some of the many applications made possible through the power and flexibility of wireless LANs:

- ❖ Doctors and nurses in hospitals are more productive because hand-held or notebook computers with wireless LAN capability deliver patient information instantly.

- ❖ Consulting or accounting audit teams or small workgroups increase productivity with quick network setup.

- ❖ Network managers in dynamic environments minimize the overhead of moves, and changes with wireless LANs, thereby reducing the cost of LAN ownership.

- ❖ Training sites at corporations and students at universities use wireless connectivity to facilitate access to information, information exchanges, and learning.

- ❖ Network managers installing networked computers in older buildings find that wireless LANs are a cost-effective network infrastructure solution.

- ❖ Warehouse workers use wireless LANs to exchange information with central databases and increase their productivity.

- ❖ Network managers implement wireless LANs to provide backup for mission-critical applications running on wired networks.

- ❖ Restaurant waitresses and car rental service representatives provide faster service with real-time customer information input and retrieval.

- ❖ Senior executives in conference rooms make quicker decisions because they have real-time information at their fingertips.

1.3 Benefits of WLANs

The widespread strategic reliance on networking among competitive businesses and the meteoric growth of the Internet and online services are strong testimonies to the benefits of shared data and shared resources. With wireless LANs, users can access shared information without looking for a place to plug in, and network managers can set up or augment networks without installing or moving wires. Wireless LANs offer the following productivity, service, convenience, and cost advantages over traditional wired networks:

❖ **Mobility improves productivity and service**

Wireless LAN systems can provide LAN users with access to real-time information anywhere in their organization. This mobility supports productivity and service opportunities not possible with wired networks.

❖ **Installation Speed and Simplicity**

Installing a wireless LAN system can be fast and easy and can eliminate the need to pull cable through walls and ceilings.

❖ **Installation Flexibility**

Wireless technology allows the network to go where wire cannot go.

❖ **Reduced Cost-of-Ownership**

While the initial investment required for wireless LAN hardware can be higher than the cost of wired LAN hardware, overall installation expenses and life-cycle costs can be significantly lower. Long-term cost benefits are greatest in dynamic environments requiring frequent moves and changes.

❖ **Scalability**

Wireless LAN systems can be configured in a variety of topologies to meet the needs of specific applications and installations. Configurations are easily

changed and range from independent networks suitable for a small number of users to full infrastructure networks of thousands of users that allow roaming over a broad area.

1.4 Issues with Wireless LANs

One of the drawbacks of wireless networks in comparison to wired networks is that they are generally less efficient and unpredictable.

The Main issues with wireless LANs are listed below:

- Throughput
- Security
- Best effort service provided.
- No built in Quality of Service (QoS)
- Medium Access Control

The main causes for these issues could be traced down to the following factors:

- Wireless LANs have limited bandwidth, high packet overheads, and is more prone to environmental factors such as obstructions, interference, weather and so on.
- The wireless medium (air) is much harder to control than a physical wire.
- The WLAN medium is also unlicensed and is therefore subject to interference from other devices.

- Wireless devices are generally constrained by size, weight and battery size, limiting the processing power and the battery life.
- Wireless networks do not have a “*collision-detect*” feature unlike Ethernet.

These factors further limit the capability of the network to provide an optimal solution. The main objective of WLAN QoS is to optimize use of limited bandwidth offered by a WLAN to address the issues noted above.

Thesis Outline

In the next sections i.e in the second chapter we discuss more about IEEE 802.11 WLans standard (infrastructure, components, basic protocols etc), In the next chapter we give an elaborate knowledge about the basic existing function protocols in the MAC layer and their summary along with the limitations of such protocols and in the later chapter we discuss about how to overcome such shortcomings with the enhanced set of protocols in IEEE 802.11e (basically HCCA) and we perform some of the simulations to prove that the enhanced protocols aid in providing QOS services, We also mention in brief about the network simulator under which all the simulations were done and Finally the conclusion and future scope.

Chapter 2

Overview of IEEE 802.11 WLANs

2.1 What is IEEE 802.11?

IEEE 802.11 is a standard developed for Wireless LANs. The standard specifies a common medium access control (MAC) Layer, which provides a variety of functions that support the operation of 802.11-based wireless LANs.

2.2 Components of an 802.11 WLAN

IEEE 802.11 is a standard developed for Wireless LANs. 802.11 WLANs consist of four major physical components. The components are:

1. Distribution System
2. Access Point
3. Wireless Medium
4. Stations

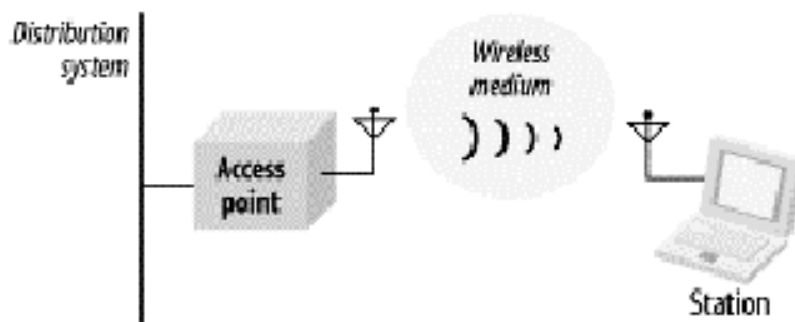


Figure 1 : Components of an 802.11 wireless network

Distribution system

When several access points are connected to form a large coverage area, they must communicate with each other to track the movements of mobile stations. The distribution system is the logical component of 802.11 used to forward frames to their destination. 802.11 does not specify any particular technology for the distribution system. In most commercial products, the distribution system is implemented as a combination of a bridging engine and a distribution system medium, which is the backbone network used to relay frames between access points; it is often called simply the backbone network. In nearly all commercially successful products, Ethernet is used as the backbone network technology.

Access points

Frames on an 802.11 network must be converted to another type of frame for delivery to the rest of the world. Devices called access points perform the wireless-to-wired bridging function. It provides access to a distribution system for associated stations.

Access points perform a number of other functions, but bridging is by far the most important.

Wireless medium

To move frames from station to station, the standard uses a wireless medium. Several different physical layers are defined; the architecture allows multiple physical layers to be developed to support the 802.11 MAC. Initially, two radio frequency (RF) physical layers and one infrared physical layer were standardized, though the RF layers have proven far more popular.

Stations

Networks are built to transfer data between stations. Stations are computing devices with wireless network interfaces. Typically, stations are battery-operated laptop or handheld computers. There is no reason why stations must be portable computing devices, though. In some environments, wireless networking is used to avoid pulling new cable, and desktops are connected by wireless LANs.

2.3 Type of Networks

The basic building block of an 802.11 network is the *basic service set* (BSS), which is simply a group of stations that communicate with each other. Communications take place within a somewhat fuzzy area, called the *basic service area*, defined by the propagation characteristics of the wireless medium. When a station is in the basic service area, it can communicate with the other members of the BSS.

Independent networks

Stations in an *independent BSS* (IBSS) communicate directly with each other and thus must be within direct communication range. The smallest possible 802.11 network is an IBSS with two stations. Typically, IBSSs are composed of a small number of stations set up for a specific purpose and for a short period of time. One common use is to create a short-lived network to support a single meeting in a conference room. As the meeting begins, the participants create an IBSS to share data. When the meeting ends, the IBSS is dissolved. Due to their short duration, small size, and focused purpose, IBSSs are sometimes referred to as *ad hoc BSSs* or *ad hoc networks*.

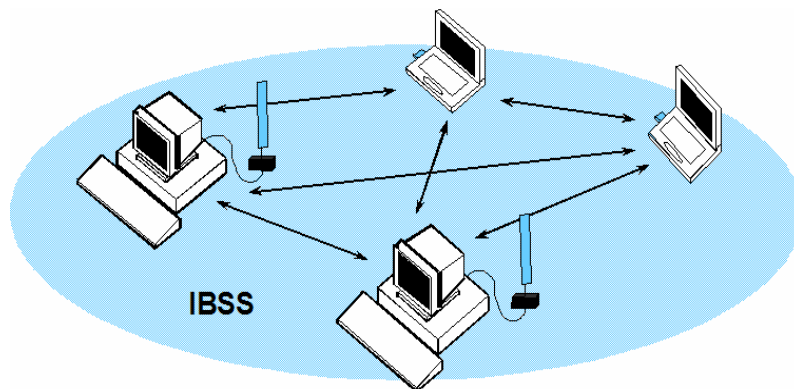


Figure 2 : Independent BSS

Infrastructure networks

Infrastructure BSS is distinguished by the use of an access point. Access points are used for all communications in infrastructure networks, including communication between mobile nodes in the same service area. If one mobile station in an infrastructure BSS needs to communicate with a second mobile station, the communication must take two hops. First, the originating mobile station transfers the frame to the access point. Second, the access point transfers the frame to the destination station. With all communications relayed through an access point, the basic service area corresponding to an infrastructure BSS is defined by the points in which transmissions from the access point can be received. Although the multihop transmission takes more transmission capacity than a directed frame from the sender to the receiver, it has two major advantages:

An infrastructure BSS is defined by the distance from the access point. All mobile stations are required to be within reach of the access point, but no restriction is placed on the distance between mobile stations themselves. Allowing direct communication between mobile stations would save transmission capacity but at the cost of increased physical layer complexity because mobile stations would need to maintain neighbor relationships with all other mobile stations within the service area.

Access points in infrastructure networks are in a position to assist with stations, attempting to save power. Access points can note when a station enters a power-saving mode and buffer frames for it. Battery-operated stations can turn the wireless transceiver off and power it up only to transmit and retrieve buffered frames from the access point.

In an infrastructure network, stations must *associate* with an access point to obtain network services. Association is the process by which mobile station joins an 802.11 network; it is logically equivalent to plugging in the network cable on an Ethernet. It

is not a symmetric process. Mobile stations always initiate the association process, and access points may choose to grant or deny access based on the contents of an association request. Associations are also exclusive on the part of the mobile station: a mobile station can be associated with only one access point. The 802.11 standard places no limit on the number of mobile stations that an access point may serve.

Implementation considerations may, of course, limit the number of mobile stations an access point may serve. In practice, however, the relatively low throughput of wireless networks is far more likely to limit the number of stations placed on a wireless network.

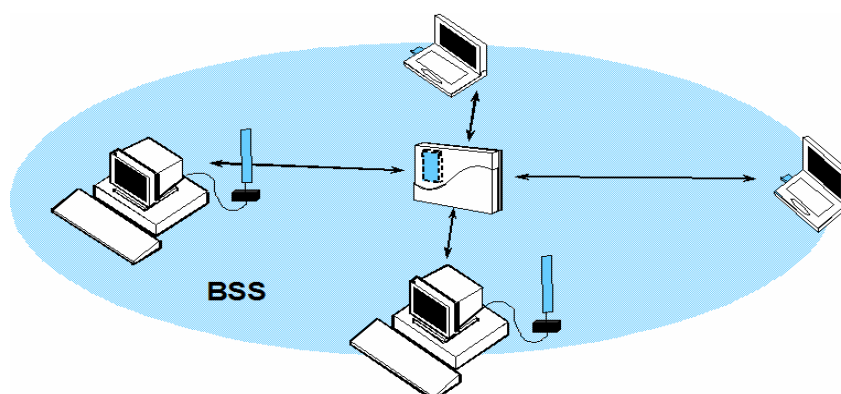


Figure 3:Infrastructure BSS

Extended service areas

BSSs can create coverage in small offices and homes, but they cannot provide network coverage to larger areas. 802.11 allows wireless networks of arbitrarily large size to be created by linking BSSs into an *extended service set* (ESS). An ESS is created by chaining BSSs together with a backbone network. 802.11 does not specify

a particular backbone technology; it requires only that the backbone provide a specified set of services.

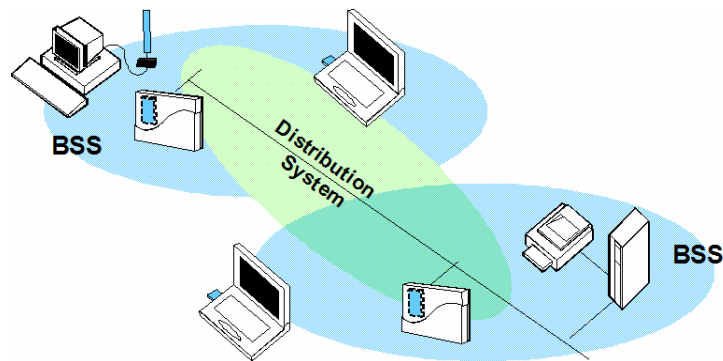


Figure 4 : Infrastructure BSS (distributed)

In the above Figure, the ESS is the union of the two BSSs (provided that both the access points are configured to be part of the same ESS).

2.4 Original 802.11 MAC

The original 802.11 MAC does not provide differentiated services based on traffic type. However, as wireless networks provide multimedia services involving voice and video, high packet overhead with limited bandwidth in a WLAN can become a major stumbling block for delivering delay-sensitive packets. In the original 802.11 standard, as much as one third of the data rate can be consumed by packet fragmentation, inter-frame spacing and acknowledgments. Furthermore, under heavy traffic load conditions, collisions and backoffs can severely deteriorate the quality of voice and video applications.

The 802.11 standard specifies two channel access mechanisms:

- **Distributed Coordination Function (DCF)**
- **Point Coordination Function (PCF)**

DCF is a *contention based* mechanism i.e. each station participates in deciding which station will get access to the communication medium. PCF is a *polling based (contention free)* mechanism. Point Coordinator (PC) at AP decides who accesses the communication medium by polling that station. DCF implementation is mandatory. PCF implementation is optional and it co-exists with DCF. PCF is implemented on top of DCF.

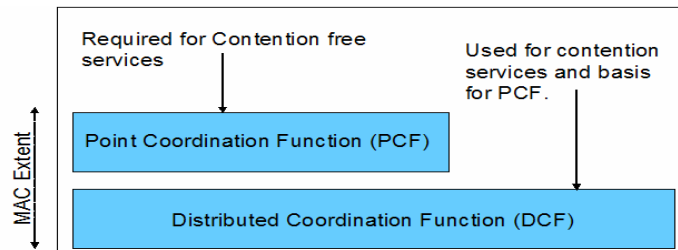


Figure 5: 802.11 MAC architecture

Our project deals with Contention Free Access. So, we shall focus more into contention free access method i.e. PCF.

Before we start discussing the access mechanisms, we need to understand the Inter Frame Space defined in 802.11.

2.5 Inter Frame Spacing

IEEE 802.11 defined four Inter Frame Spaces:

- Short Interframe Space
- PCF Interframe Space
- DCF Interframe Space
- Extended Interframe Space

Short interframe space (SIFS)

The SIFS is used for the highest-priority transmissions, such as RTS/CTS frames and positive acknowledgments. High-priority transmissions can begin once the SIFS has elapsed. Once these high-priority transmissions begin, the medium becomes busy, so frames transmitted after the SIFS has elapsed have priority over frames that can be transmitted only after longer intervals.

PCF interframe space (PIFS)

The PIFS, sometimes erroneously called the priority interframe space, is used by the PCF during contention-free operation. Stations with data to transmit in the contention-free period can transmit after the PIFS has elapsed and preempt any contention-based traffic.

DCF interframe space (DIFS)

The DIFS is the minimum medium idle time for contention-based services. Stations may have immediate access to the medium if it has been free for a period longer than the DIFS.

Extended interframe space (EIFS)

The EIFS is not a fixed interval. It is used only when there is an error in frame transmission.

2.6 Overview of Distributed Coordination Function (DCF)

DCF allows sharing of the wireless medium between the Stations (STAs) and the AP using Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA). DCF provides best-effort service and does not provide either medium access priority or support delay and bandwidth requirements of different applications.

DCF mode operates as follows. Each station checks whether the medium is idle before transmitting. If the medium is detected to be idle for Distributed Inter-frame

Space (DIFS) interval of time, the station begins transmission. In the case where the medium is determined to be busy, the station defers transmission until the medium is idle for DIFS time. The station then selects a random backoff interval (using a backoff algorithm) and decrements a backoff counter while the medium is idle. The backoff mechanism is used to prevent two or more stations from transmitting simultaneously. Once the backoff interval has expired, the station begins its transmission. The range from which the random backoff interval is selected is called the Contention Window (CW) and depends on the number of previous retransmission attempts. Once the MAC Service Data Unit (MSDU) has been transmitted, the station waits for Short Inter-Frame Space (SIFS) duration for the Acknowledgment (ACK) from the recipient.

2.7 Overview of Point Coordination Function (PCF)

To support time-bounded services, the IEEE 802.11 standard defines the Point Coordination Function (PCF) to let stations have priority access to the wireless medium, coordinated by a station called Point Coordinator (PC). The PCF has higher priority than the DCF, because it may start transmissions after a shorter duration than DIFS; this time space is called PCF Interframe Space (PIFS). The PC polls the stations asking them to transmit a pending frame. With PCF, a Contention Free Period (CFP) and a Contention Period (CP) alternate over time, in which a CFP and the following CP form a superframe. A superframe starts with a so-called beacon frame, regardless whether PCF is active or not. The PC, which is typically collocated with the AP, generates beacon frames at regular intervals. Note that the beacon frame is required in pure DCF even if there is only contending traffic.

We shall provide a detailed insight into the Point Coordination Function (PCF) mechanism in the following chapter.

Chapter 3

Point Coordination Function (PCF)

PCF is an optional channel-access mechanism in the 802.11 standard that is not commonly implemented due to lack of market demand. PCF provides contention-free access to the medium. It was designed to support time-sensitive applications. The Point Coordinator (PC) residing in the AP provides a contention-free wireless medium access. A polling method is used to provide access, with the PC acting as the polling master. This eliminates collisions and the time spent on backoff and contention as described previously for DCF.

Contention-free access to the medium is not provided at all times. When PCF is used, time on the medium is divided into a Contention-Free Period (CFP) and a Contention Period (CP) by the PC. During CFP and CP, PCF and DCF are used to access the medium, respectively.

The wireless channel is divided into **superframes**. Superframe consists of a contention free period (CFP) and a contention period (CP). CFP and CP times alternate. Alternating periods of contention-free service and contention-based service repeat at regular intervals, which are called the contention-free repetition interval.

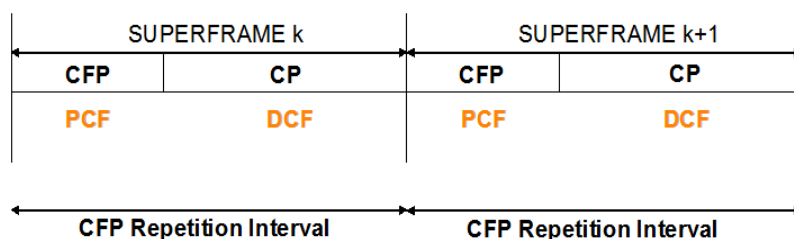


Figure 6 : 802.11 Superframe

3.1 Reserving the medium during the contention-free period

At the beginning of the contention-free period, the access point transmits a Beacon frame. One component of the beacon announcement is the maximum duration of the contention-free period, *CFPMaxDuration*. All stations receiving the Beacon set the NAV (Network Allocation Vector) to the maximum duration to lock out DCF-based access to the wireless medium. The PC, which is typically collocated with the AP, generates beacon frames at regular beacon frame intervals, thus every station knows when the next beacon frame will arrive; this time is called *target beacon transition time (TBTT)* and is announced in every beacon frame.

As an additional safeguard to prevent interference, all contention-free transmissions are separated only by the short interframe space and the PCF interframe space. Both are shorter than the DCF interframe space, so no DCF-based stations can gain access to the medium using the DCF.

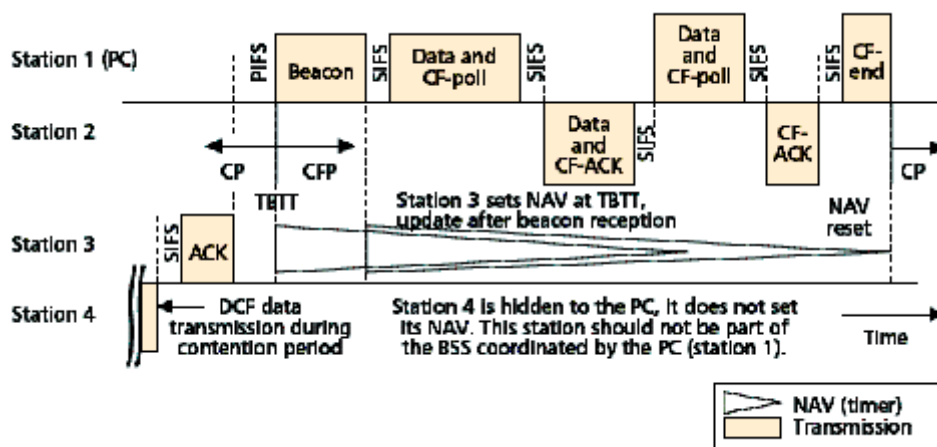


Figure 7 : PCF Operation

3.2 The polling list

After the access point has gained control of the wireless medium, it polls any associated stations on a *polling list* for data transmissions. During the contention-free

period, stations may transmit only if the access point solicits the transmission with a polling frame. Contention-free polling frames are often abbreviated CF-Poll. *Each CF-Poll is a license to transmit one frame.* Multiple frames can be transmitted only if the access point sends multiple poll requests.

The polling list is the list of privileged stations solicited for frames during the contention-free period. Stations get on the polling list when they associate with the access point. The Association Request includes a field that indicates whether the station is capable of responding to polls during the contention-free period.

3.3 Transmissions from the Access Point

Generally, all transmissions during the contention-free period are separated by only the short interframe space. To ensure that the point coordinator retains control of the medium, it may send to the next station on its polling list if no response is received after an elapsed PCF interframe space. The access point polled the second station on its list but received no response. After waiting one PCF interframe space, the access point moves to the third station on the list. By using the PCF interframe space, the access point ensures that it retains access to the medium.

The access point may use several different types of frames during the contention-free period. During this period, the point coordinator has four major tasks. In addition to the "normal" tasks of sending buffered frames and acknowledging frames from the stations, the point coordinator can poll stations on the polling list to enable them to send frames; it may also need to transmit management frames.

Time in the contention-free period is precious, so acknowledgments, polling, and data transfer may be combined to improve efficiency. When any subsets of these functions are combined into a single frame, the result is a bit strange. A single frame

could, for example, acknowledge the receipt of the previous frame, poll a different station for buffered data, and send its own data to the station on the polling list.

Several different frame types can be used in the contention free period:

1. Data

The standard data frame is used when the access point is sending a frame to a station and does not need to acknowledge a previous transmission. The standard Data frame does not poll the recipient and thus does not allow the recipient to transmit any data in return. The Data-Only frame used in the contention-free period is identical to the Data frame used in contention-based periods.

2. CF-Ack

This frame is used by stations to acknowledge the receipt of a frame when no data needs to be transmitted. Contention-free acknowledgments are longer than the standard control frame acknowledgment, so this frame may not be used in actual implementations.

3. CF-Poll

CF-Poll frames are sent by the access point to a mobile station to give the mobile station the right to transmit a single buffered frame. It is used when the access point does not have any data for the mobile station. When a frame for the mobile station is available, the access point uses the Data+CF-Poll frame type.

4. Data + CF-Ack

This frame combines data transmission with an acknowledgment. Data is directed to the frame recipient; the acknowledgment is for the previous frame transmitted and usually is not for the recipient of the data.

5. Data + CF-Poll

This frame is used by access points to transmit data to a mobile station and request one pending frame from the mobile station. The Data+CF-Poll can only be sent by the access point during the contention-free period.

6. CF-ACK + CF-Poll

This frame acknowledges the last frame from one of the access point's clients and requests a buffered frame from the next station on the polling list. It is directed to the next station on the polling list, though the acknowledgment may be intended for any mobile station associated with the access point.

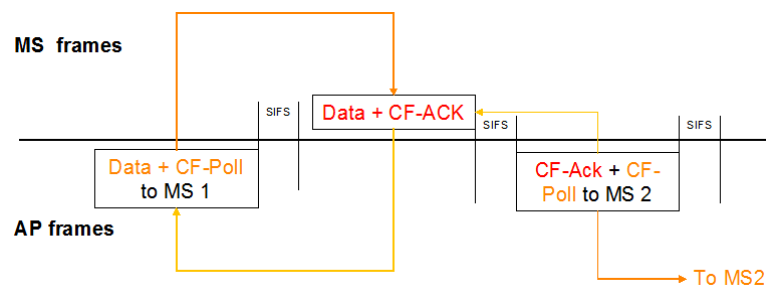


Figure 8: Example of Data + CF-Poll. Data + CF-ACK. CF-ACK + CF-Poll

7. Data + CF-ACK + CF-Poll

This frame brings together the data transmission, polling feature, and acknowledgment into one frame for maximum efficiency.

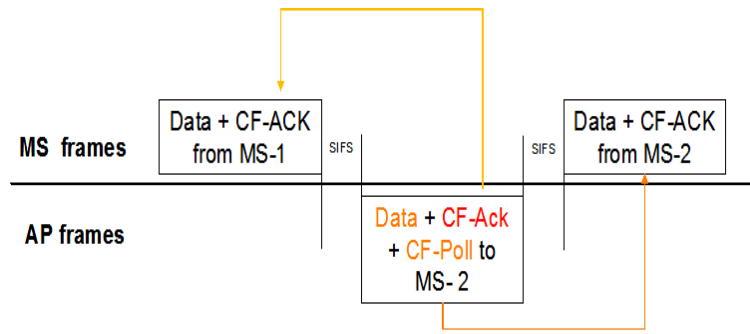


Figure 9: Example of Data + CF-ACK + CF-Poll

8. CF-End

This frame ends the contention-free period and returns control of the medium to the contention-based mechanisms of the DCF.

9. CF-End + CF-Ack

This is the same as the CF-End frame but also acknowledges the previously transmitted Data frame.

10. Any Management

No restriction is placed by the standard on which management frames can be transmitted during the contention-free period. If the rules applying to a particular frame type allow its transmission, the access point may transmit it.

11. Null Frame

It is transmitted by a station in response to a poll from the AP, when it has no data to send.

3.4 Contention-Free Period Duration

The contention period must be long enough for the transfer of at least one maximum-size frame and its associated acknowledgment. The minimum length of the contention period is the time required to transmit and acknowledge one maximum-size frame. It is possible for contention-based service to overrun the end of the contention period, however. When contention-based service runs past the expected beginning of the contention-free period, the contention-free period is *foreshortened*.

When the contention-free period is foreshortened, the existing frame exchange is allowed to complete before the beacon announcing the start of contention-free operation is transmitted.

The point coordinator may also terminate the contention-free period prior to its maximum duration by transmitting a CF-End frame. It can base this decision on the size of the polling list, the traffic load, or any other factor that the access point considers important.

3.5 Power Management and the PCF

Power conservation during the contention-free period is similar to power conservation during the contention-based period, with a few minor exceptions. The basic distinction between the two is that frame delivery must obey the PCF rules, so buffered frames can be delivered only to CF-Pollable stations. Stations that do not support PCF operations must wait until contention-based service resumes before retrieving buffered frames.

Stations on the polling list are not allowed to sleep during the contention-free period. When the access point is performing its point coordination functions, it may poll any station on the polling list at any time. Frames destined for stations on the polling list

do not need to be buffered during the contention-free period because those stations do not sleep.

Frame buffering is identical under contention-free and contention-based service. By maintaining power-saving status for each station, the access point can buffer frames for any station in a low-power mode. Broadcast and multicast frames are buffered whenever an associated station is in a low-power mode.

If a station switches from a low-power mode to the active mode, any frames buffered for it are transferred to the point coordination function for delivery during the contention-free period. The transfer does not result in immediate delivery, but the access point can place the frames into a queue for transmission as soon as the point coordination function permits.

3.6 PCF Operation – Summary

- PC senses the channel.
- If channel is idle for PIFS interval, PC sends the beacon frame and thereby announcing the start of CFP.
- Stations using DCF set their NAV to CFP_Max_Duration (from the beacon frame).
- PC polls each station on its polling list in a round robin fashion.
- PC can actively terminate CFP by sending a CF-End frame announcing the start of CP.
- On reaching CFP_Max_Duration, CFP ends and CP starts.
- Station using DCF reset their NAV to 0.

3.7 Problems with PCF

Neither DCF nor PCF has sufficient functionality to provide QoS demanded by multimedia applications. DCF treats all traffic the same—with all stations contending for the medium with the same priority. PCF also has several inadequacies in the support of QoS:

- Round Robin Polling and wastage of polls and bandwidth.
- All the communications have to go through the AP. This mechanism degrades the bandwidth performance of WLAN.
- Unpredictable Beacon delays.
- Transmission times of the polled stations are unknown.
- No concept of differentiating frames with different priorities.
- No mechanisms for the stations to communicate their QoS requirements to the AP.

Round Robin Polling

The PC polls each station in the polling list in a round robin fashion. However, this can lead to a lot of wastage of polls or bandwidth under certain cases. E.g. If a station has no data to send during CFP (wastage of bandwidth), a station receives data just after it was polled and it replied with a null etc.

Unpredictable Beacon Delays

At TBTT, a PC schedules the beacon as the next frame to be transmitted, and the beacon can be transmitted when the medium has been determined to be idle for at

least PIFS. Depending on the wireless medium at this point of time, i.e., whether it is idle or busy around the TBTT, a delay of the beacon frame may occur. The time the beacon frame is delayed, i.e., the duration it is sent after the TBTT, delays the transmission of time-bounded MSDUs that have to be delivered in CFP. From the legacy 802.11 standard, stations can start their transmissions even if the MSDU Delivery cannot finish before the upcoming TBTT.

This may severely affect the QoS as this introduces unpredictable time delays in each CFP.

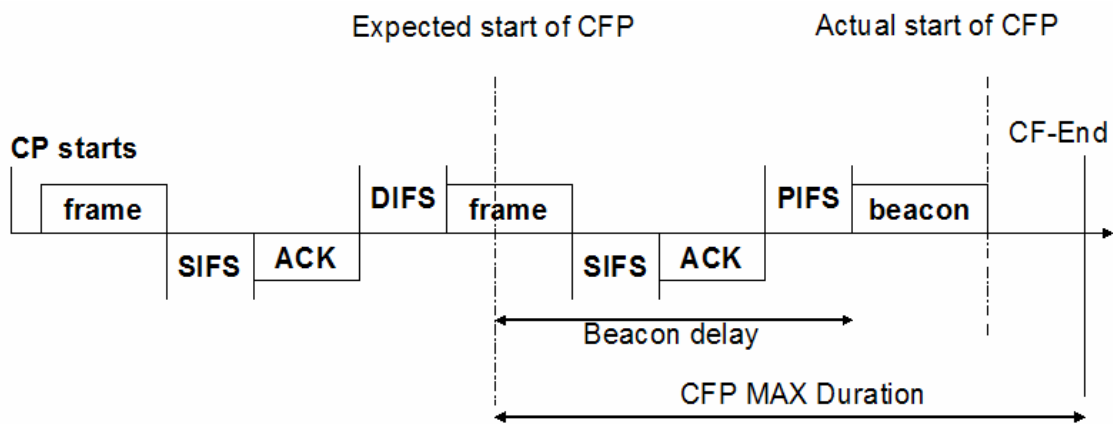


Figure 10 : Unpredictable Beacon Delay

Transmission times of the polled stations are unknown

There is another problem with the PCF, the unknown transmission time of polled stations. A station that has been polled by the PC is allowed to send a single frame that may be fragmented and of arbitrary length, up to the maximum of *2304 bytes* (*2312 bytes* with encryption). Further, different modulation and coding schemes are specified in 802.11a, thus the duration of the MSDU Delivery that happens after polling is not under the control of the PC. This destroys any attempt to provide QoS to other stations that are polled during the rest of the CFP.

To overcome the first shortcoming i.e. round robin polling and wastage of polls and bandwidth, we suggest the following schemes, which may not eliminate the poll wastage but, they will definitely reduce them.

3.8 Suggestions to overcome the difficulties of the IEEE 802.11 PCF scheme

The following techniques may be applied to 802.11 PCF to improve its performance:

1. Dynamic Polling List
2. Multipolling
3. Dynamic CFP Interval and More-Data field
4. Avoiding CFP-to-CP migration

1. Dynamic Polling List

The biggest disadvantage of PCF is that a lot of bandwidth is wasted by sending CF-Polls and Null packets when STAs have no packets to send. Usually, in normal conversations, when one person is talking, the other one listens without talking. So, one STA is sending packets and the other STA is not sending but just receiving packets (if silence suppression is used) and a lot of bandwidth is wasted with a lot of unnecessary CF-Polls and Null packets. This waste of bandwidth significantly reduces the capacity of VoIP in wireless network. To minimize the waste of bandwidth, We introduce a dynamic polling list. The dynamic polling list maintains the active nodes, i.e., those sending data. The PC can avoid polling non-active nodes which are not sending any data.

Removing an STA from the Polling List:

The PC needs to remove an STA when it stops sending packets. When the AP gets a Null packet from an STA, the PC removes the STA from the polling list. However, we should not remove the STA immediately because a packet can be lost or delayed; as a heuristic we remove the STA after the AP gets three consecutive Null packets from the same STA.

Adding an STA to the Polling List:

When an STA starts to talk, it sends the first VoIP packet in CP. When the AP gets a VoIP packet in CP, it adds the STA into the polling list and the STA is polled starting from the next CFP. The problem of this method is that if CP is very congested, the first packet of a talk-spurt could be delayed until the next CP. However, even if the first packet is delayed, the next packets are sent without significant delay because of the More Data field.

2. Multipolling

The AP sends a single poll (Multipoll) for a group of stations. Stations of a particular group are arranged in some order (which is known to everyone) and they transmit turn by turn.

If a station comes at position k according to the order, then he has to wait for $k-1$ idle PIFS intervals to get his turn to transmit. The AP waits for $n-1$ idle PIFS intervals where n is the no of stations in the polled group. After that it can poll the next group.

3. Dynamic CFP Interval and the More Data Field

In PCF, the CFP interval should be less than or equal to the packetization interval to deliver packets without delay. However, VoIP clients differ in their packetization intervals. When more than one packetization interval is used in a wireless network, the choice of the CFP interval affects the capacity and quality of VoIP. For example, let's say STA A uses 10 ms and STA B uses 20 ms packetization intervals. In this case, when 10 ms is used as CFP interval, STA B is polled twice per packetization interval so one CF-Poll is wasted every 20 ms. When 20 ms is used as CFP interval, STA A generates 2 packets and only one packet can be sent in a CFP interval (20 ms). The other packet should be sent in CP or in the next CFP interval. This creates significant delay and such a delay is accumulated.

To solve this problem, we use the *More Data field* in the Frame Control field. This field is defined in IEEE 802.11 standard and used in power-save mode to indicate that at least one additional buffered MSDU is present for the same STA. We use the largest packetization interval as CFP interval.

If an STA needs to send more than one packet, it sets the More Data field in the first packet. When the AP detects the More Data field in a data packet, the AP polls the STA again until the More Data field is not set any more. In the above example, STA A can send two packets per 20 ms instead of sending only one packet. The first packet is delayed 10 ms but the delay is never accumulated and CF-Polls are not wasted.

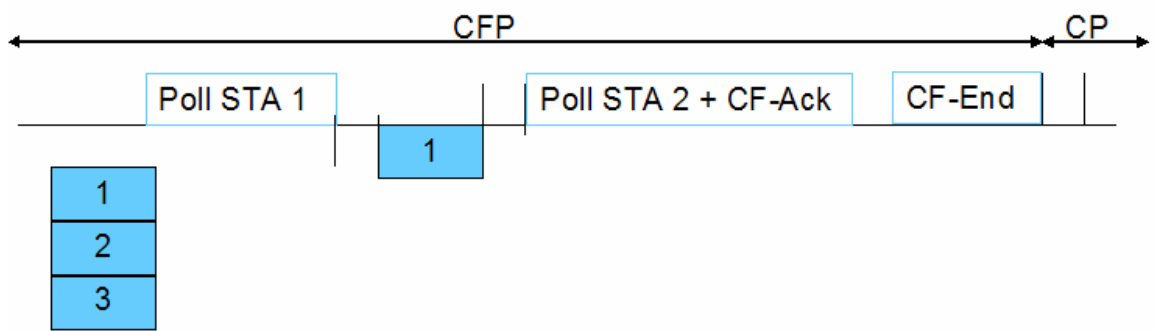


Figure 11 : Typical Scenario without using More Data Bit

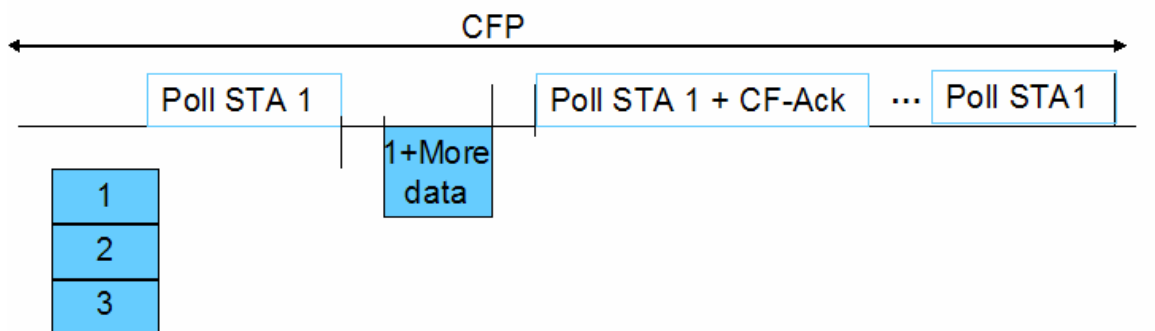


Figure 12 : After using More Data Bit

4. Avoid CFP-to-CP Migration

We cannot eliminate wasted CF-Polls and Null packets completely even if we use the dynamic polling list, a dynamic CFP interval and the More Data field. PCF has the following critical disadvantage.

If the STA is polled before a VoIP packet is created, the CF-Poll is wasted and the data packet will be sent in CP or in the next CFP interval. If CP is not very

congested, the packet will be sent in CP, the next CF-Poll will be wasted and the same problem will occur over and over. Eventually, most of the CF-polls are wasted and most of the packets are sent in CP.

This is a synchronization problem between CF-Polls and data. If this synchronization problem happens in a lot of STAs, CFP is shortened and CP is increased. During CP there is enough time to send the packets missed in CFP, and the vicious cycle is repeated. The ideal solution for this problem is synchronizing CF-Polls and data. In reality, it is very difficult to synchronize CF-Polls and data because the PC cannot poll an STA with the exact same interval, and at the application layer, packets cannot be created at exactly the same rate.

We propose a practical solution, which prevents STAs from sending VoIP packets in CP when there is only one VoIP packet in their queue so that the packet can be sent in the next CFP. That is, VoIP packets can be sent in CP only when there is more than one packet in the queue. In this way, we can avoid wasting CF-Polls, and the number of packets sent in CP is decreased, optimizing the CP for the non-VoIP packets.

Chapter 4

HCCA- QoS Standard for Wireless Lans

To optimize the best use of the resources and fulfill the resource requirements of different applications, QoS provides mechanisms to control access and usage of the medium based on the application. Each application has different needs in terms of latency, bandwidth and packet-error rate and, therefore, QoS must cater to each of these needs. Applications requiring low latency (e.g. voice) may be given higher priority to use the medium, whereas applications requiring higher bandwidth may be assigned longer transmit times (e.g. video). Other traffic may require high reliability (e.g. email and data) and must be delivered with low packet-error rate.

The original 802.11 standard was not designed to provide differentiation and prioritization based on the traffic type, thus providing less than optimal user experience for voice and video over WLAN applications. Voice applications require no dropped calls or bad connections. Video/audio applications require enough bandwidth to maintain high quality video/audio streams. Email and file-sharing applications require ensuring delivery of error-free files.

To fulfill these requirements, the IEEE 802.11e has added several QoS features and enhancements to WLAN. The key benefits of the 802.11e standard are:

- ❖ Reduces latency by prioritizing wireless packets based on traffic type.
- ❖ Enables Access Point (AP) to schedule resources based on client/station data rate and latency needs.
- ❖ Improves wireless bandwidth efficiency and packet overheads.

The 802.11e standard provides a comprehensive QoS solution and contains many optional features. Here we discuss the key features of the 802.11e standard, their benefits and also provide our suggestions on how to improve the standard mechanisms.

While QoS issues in the Ethernet are not been considered as a high priority topic due to the huge improvements of physical layer bandwidths, the IEEE 802.11e group is developing MAC enhancements to support QoS sensitive applications (e.g., multimedia applications). These enhancements will enable a better mobile user experience and will make more efficient use of the wireless channel.

The IEEE 802.11e is currently defining enhancements to the 802.11 MAC access methods (DCF and PCF), providing the classes of service, enhanced security and authentication mechanism. These enhancements are defined in 802.11e which introduces a new access method called **Hybrid Coordination Function (HCF)**. It includes two new schemes namely the **Enhanced Distributed Coordination Access (EDCA)** and **HCF Controlled Channel Access (HCCA)**.

In the following pages of this section, we will describe firstly this new access method of IEEE 802.11e draft D2.0, secondly we will present in detail one of its basic method, the HCF controlled channel access method.

4.1 Hybrid Control Function (HCF)

In order to support both integrated service and differentiated service QoS approaches in 802.11, TGe has defined a new mechanism called HCF. This mechanism is backwardly compatible with legacy DCF and PCF. *It has both polling based and contention based channel access mechanisms in a single channel access protocol.* HCF consists of two access methods, **Enhanced Distributed Coordination Access (EDCA)** - a distributed channel access scheme; and called **HCF Controlled Channel Access (HCCA)** - a polling-based scheme.

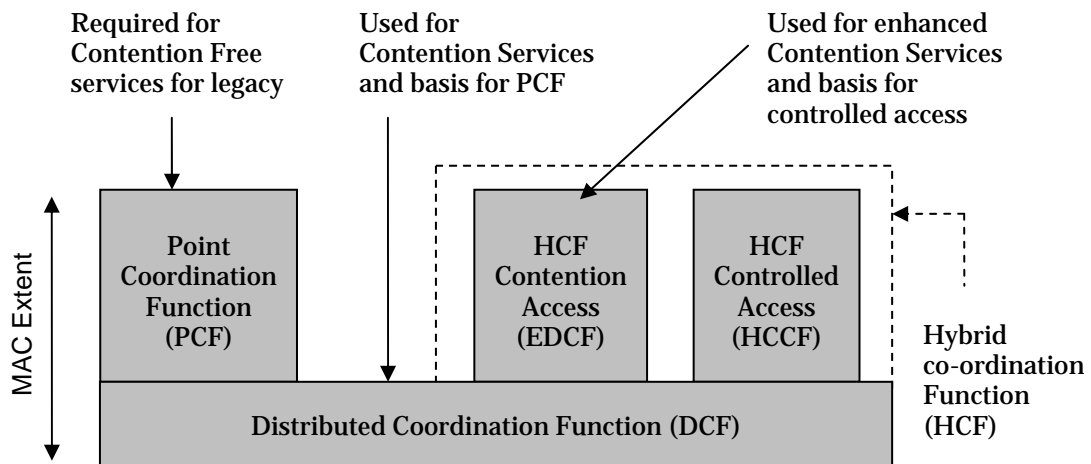


Figure 13 : MAC architecture in 802.11e

In HCF there may still be two phases of operations within the superframes, the CFP and CP, which alternate over time. The EDCF is used in the CP only, while the HCF controlled channel access method is use in both phases, which makes this new coordination function hybrid.

4.2 Enhancements of 802.11e over 802.11

The IEEE 802.11e has defined enhancements to the original 802.11 MAC (Medium Access Control). The major enhancements in 802.11e are:-

1. Traffic differentiation
2. Concept of Transmission Opportunity (TXOP)
3. Enhanced DCF or EDCF(contention-based access scheme)
4. HC controlled channel access or HCCA (contention free access scheme)

In this section, we will explain only traffic differentiation and transmission opportunity; and briefly talk about EDCA. The next section will deal with HCCA in full details.

Traffic Differentiation

One main feature of HCF is to introduce four access category (AC) queues and eight traffic stream (TS) queues at MAC layer. When a frame arrives at MAC layer, it is tagged with a traffic priority identifier (TID) according to its QoS requirement, which can take the values from 0 to 15. The frames with TID values from 0 to 7 are mapped into four AC queues using EDCF access rule. On the other hand, frames with TID values from 8 to 15 are mapped into eight TS queues using HCF controlled channel access rule. *The reason of separating TS queues from AC queues is to support strict parameterized QoS at TS queues while prioritized QoS is supported at AC queues.*

UP, User Priority (Same as 802.1D)	802.1D Designation	802.11e AC (Access Category)	Service type
2	Not defined	0	Best Effort
1	Background (BK)	0	Best Effort
0	Best Effort (BE)	0	Best Effort
3	Excellent Effort (EE)	1	Video Probe
4	Controlled Load (CL)	2	Video
5	VI (Video <100ms latency and jitter)	2	Video
6	VO (Video <10ms latency and jitter)	3	Voice
7	Network Control (NC)	3	Voice

Figure 14 : Mapping User priorities (UP) to Access Categories (AC)

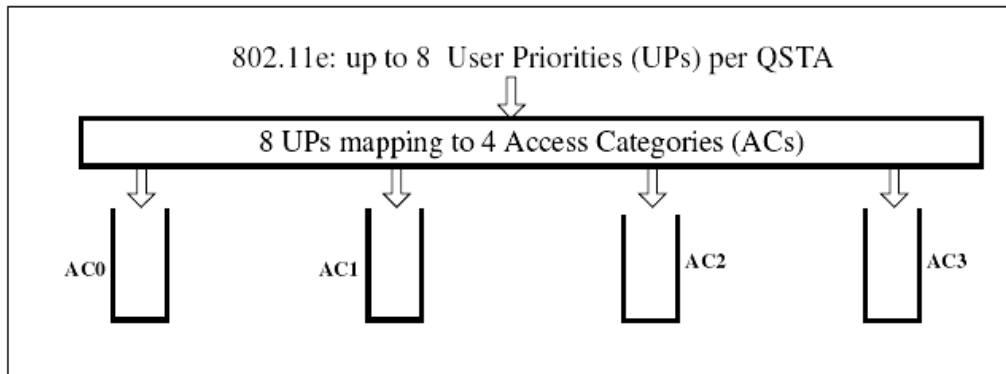


Figure 15 : Traffic Differentiation

TXOP (Transmission Opportunity)

Another crucial feature of 802.11e MAC is the TXOP. A TXOP is defined as the interval of time when a particular QSTA (QoS station) has the right to initiate transmissions onto the wireless medium. A TXOP is denoted by the set $\langle \textit{Starting time}, \textit{Maximum Duration} \rangle$. During a TXOP, there can be a series of frames transmitted by a QSTA separated by SIFS.

During the contention period (CP), each TXOP begins either when the medium is determined to be available under the EDCF rules (*EDCF-TXOP*) or when the QSTA receives a special frame from the HC (*polled-TXOP*). The duration of an EDCF-TXOP is limited by a QBSS-wide TXOP limit distributed in the beacon frames, while the duration of a polled-TXOP is specified in the header of the QoS CF-Poll frame. During the contention-free period (CFP) the starting time and maximum duration of each TXOP is specified by the HC using the QoS CF-Poll function. Within the limits of the duration of a TXOP, decisions regarding which MPDU (MAC Protocol Data Unit) that can be transmitted are made locally by each QSTA. By this way, internal collisions with one QSTA are avoided.

A TXOP can end in one of the following situations:-

1. The QoS transmitting which has been allocated the current TXOP has no more data to transmit
2. The current TXOP time expires
3. Currently transmitting station leaves medium free for DIFS time, after which, EDCA traffic gains control of channel

4.3 Overview of Enhanced Distributed Coordination Access (EDCA)

The EDCA is designed for the contention-based prioritized QoS support. Each QoS-enhanced STA (QSTA) has 4 queues (ACs), to support 8 user priorities (UPs) as defined in IEEE 802.1D. Each AC queue works as an independent DCF STA and uses its own backoff parameters.

In EDCA, relative priorities are provisioned by configuring the time to access the channel once it is sensed idle and by changing the size of the contention window. EDCA uses the contention window to assign priority to each traffic category. Indeed, assigning a short contention window to a high priority TC ensures that in most cases, high priority TC is able to transmit ahead of low priority one. Thus, the CWmin and CWmax parameters can be set differently for different traffic categories, such as, a high priority TC with small values of CWmin and CWmax.

For further differentiation, various interframe spaces (IFS) can be used by different traffic categories. Instead of using a DIFS, as a minimum specified idle duration time as defined in DCF, a new kind of interframe space called **Arbitration Interframe Space (AIFS)** is used. Thus, a traffic category (TC) with a small AIFS has a high priority.

4.4 Overview of HCF Controlled Channel Access (HCCA)

While EDCF prioritization is an important enhancement of legacy DCF, it is not enough to provide effective traffic protection and QoS guarantees. These can only be achieved with polling-based HCCA and centralized scheduling.

The HCF controlled channel access uses a hybrid coordinator (HC) which manages the allocation of the wireless medium data transfer bandwidth. The HC is typically collocated with the QoS enhanced access point (QAP) of a QoS basic service set (QBSS).

4.5 HCF Controlled Channel Access (HCCA)

The HCF controlled channel access mechanism is designed for the parameterized QoS support, which combines the advantages of PCF and DCF. HCF can start the controlled channel access mechanism in both CFP and CP intervals, whereas PCF is only allowed in CFP.

4.5.1 The Hybrid Controller (HC)

The HCF controlled channel access uses a hybrid coordinator (HC) which manages the allocation of the wireless medium data transfer bandwidth. The HC is typically collocated with the QoS enhanced access point (QAP) of a QoS basic service set (QBSS).

The HC has higher priority access than wireless stations (WSTAs) to be able to allocate transmission opportunities (TXOPs) to these WSTAs (i.e., the HC can access the wireless medium only after detecting the channel is idle for PIFS, which is shorter than DIFS, thus priority over DCF traffic, and also shorter than any AIFS, priority over EDCF traffic).

HC traffic delivery and TXOP allocation may be scheduled during both the CP and CFP in order to meet the QoS requirements of particular traffic categories (TCs) or traffic streams (TSs). Moreover, the HC transfers data to QSTAs during the CFP based on the amounts of queued traffic belonging to different TCs or TSs.

Although HC may seem similar to a PC (point controller from 802.11 standard), there are some remarkable differences between them as listed below:-

1. HC has the ability to poll a QSTA during both contention period & contention free period, unlike in 802.11, where the PC can only poll in contention period.
2. In 802.11, the PC has no control over the time interval for which a station has control over the wireless medium. But, in 802.11e, HC grants a polled-TXOP to a QSTA, which limits the duration of QSTA's access to the medium.

4.5.2 Relationship between CFP, CP, CAP & CC

In HCF controlled channel access mode, *Controlled Access Period (CAPs)* are defined as several intervals within one CP when short bursts of frames are transmitted using polling-based controlled channel access mechanisms. But, during the rest of the CP, frames are transmitted using EDCF contention-based rules. A CAP starts when the medium remains free for PIFS time and ends when medium remains idle for DIFS interval. Following figure shows the relationship between CFP, CP, and CAPs within one 802.11e superframe.

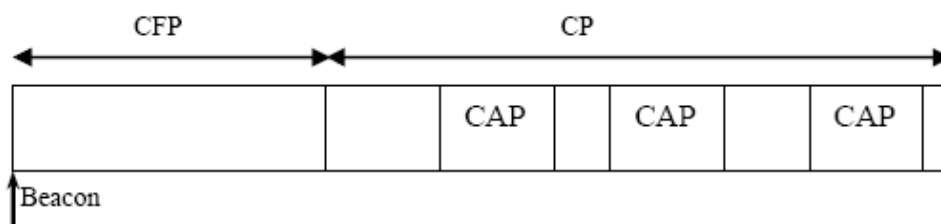


Figure 16 : Relationship of CFP, CP, CAP within one superframe

CAPs may also include *controlled contention (CC)* periods, which allow wireless stations to request the allocation of polled TXOPs without having to contend with EDCF traffic. These requests serve to initiate periodic polled TXOPs to handle periodic traffic under a particular Traffic Specification (TSPEC), or to handle a traffic burst or to create an initial TXOP for a new QSTA (or newly active-TS).

Controlled contention is a way for the HC to learn which stations needs to be polled, at which time and for which duration. *By introducing controlled contention (CC) in CAP, the HCF channel mode access can provide Guaranteed Services with a much higher probability than EDCF, especially under heavy load.*

Each instance of CC occurs during the controlled contention interval (CCI) which is started when the HC sends a specific *CC control frame*. The CC control frame forces legacy stations to set their NAVs until the end of the CCI, thus improving protection of the CCI duration. The CC frame generated by the HC includes a priority mask, the duration of each Controlled Contention Opportunity (CCOP), and the number of CCOPs within the CCI (NCCOP). The priority mask is a filtering mask containing the Traffic Categories (TC) in which Reservation Requests (RR) may be placed. The following figure shows the CC frame and the controlled contention interval (CCI) as well as the RR placed in each CCOP.

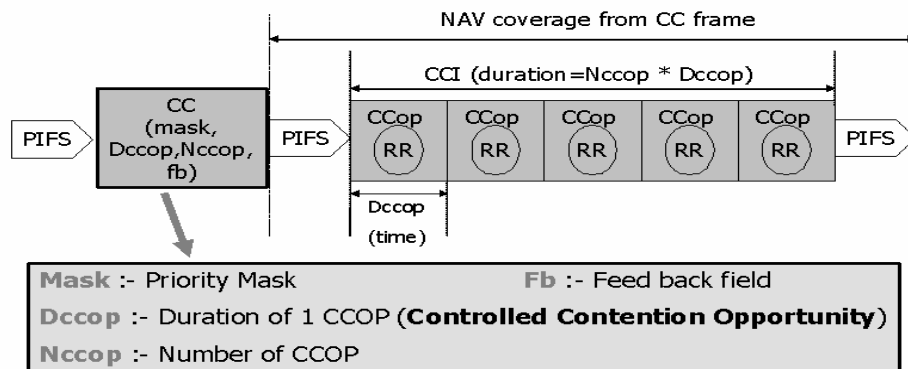


Figure 17 : Control Contention Interval & the CC frame

Upon receipt of a control frame CC, each QSTA which has an RR to send for a traffic category (TC) matching the priority mask, selects a special CCOP by using a random access protocol (CSMA/CA-like) to send its RR (after entering a special backoff time parameterized by N_{ccop} and D_{ccop}). Note that, since a random access protocol is used for RR transmission, RR frames may collide; in this case no retransmission procedure is performed. Successful reception of the request reservation frames are acknowledged by the HC to the different QSTAs in the next CCI sequence by using the feedback field (fb), i.e., fb field in the CC frame as shown in the previous figure. This mechanism enables fast collision resolution, so that the requesting QSTAs can detect collisions during the next CCI.

4.5.3 Understanding the HCF Superframe

Now that the various components of the HCF have been considered separately, we will have a look at the HCF superframe and the working of HCCA as a whole. The following figure illustrates an example of 802.11e superframe (also called beacon interval).

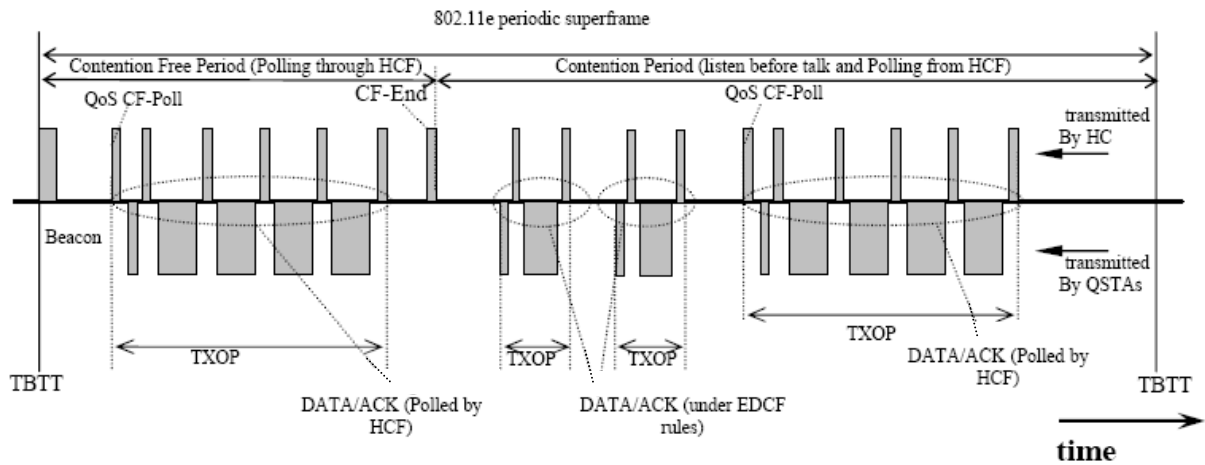


Figure 18 : 802.11e superframe

A superframe consists of two periods, the contention free period (CFP) and the contention period (CP). During the CFP, the starting time and maximum duration of each TXOP is specified by the HC using QoS CF-Poll frames. QSTAs will not attempt to get access to the wireless medium on its own, so only the HC can grant TXOP to these stations. The CFP ends after the time announced in the beacon frame or by an explicit frame, CFEnd, from the HC. During CP, each TXOP begins either when the medium is determined to be available under the EDCF rules, i.e., after AIFS plus backoff time, or when the QSTA receives a QoS CF-Poll from the HC. The QoS CF-Poll from the HC can be sent after a PIFS idle period without any backoff (high priority WM access for the HC). Therefore the HC can issue TXOPs in the CP using its prioritized medium access.

In order to achieve a better medium utilization, a **Contention Free Burst (CFB)** can take place during the CFP. CFB allow a sequence of MPDUs (more than one MPDU) to be transmitted during a TXOP, separated by a SIFS period. The MPDU within a burst is acknowledged by a BurstAck MPDU, requested by a BurstAck request sent from the originating WSTAs. This “request”/ “response” mechanism gives the recipient time to perform any necessary FEC decoding.

The burst is started by receiving a QoS CF-Poll frame from the HC. The duration values of the burst data MPDUs and any burst ACK exchange should fit within a single TXOP (i.e., polled TXOP duration specified in the QoS CF-Poll frame). The main purpose of CFB is that enables the HC to give up medium control (to possibly another QBSS) and defer control until new frames of other CFB are available. By this way the HC can relinquish medium control to other QBSSs in the same area to solve the problem of BSS overlap.

4.6 Traffic Scheduling in IEEE 802.11e

One of the main components of HCCA is the Traffic Scheduler. It decides the allocation of TXOPs to various Traffic Streams in QSTAs on the basis of Traffic Specification (TSPEC).

A QSTA can request parameterized services using the TSPEC elements. The TSPEC element contains the set of parameters that characterize the traffic stream that the QSTA wishes to establish with the HC.

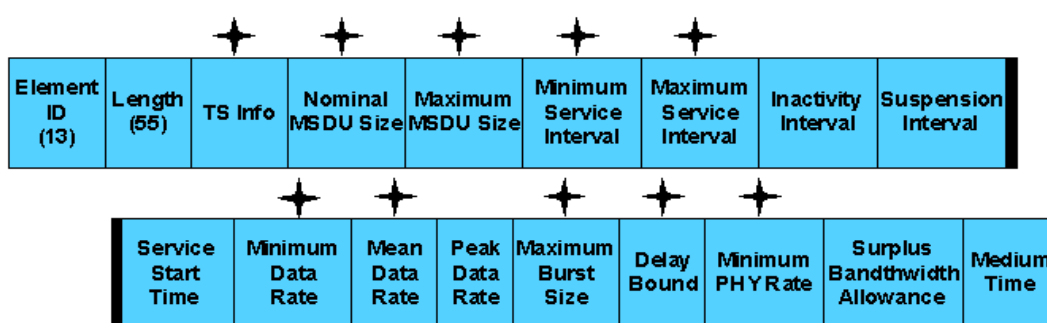


Fig 19 Tspec parameters described in 802.11e

Some of these parameters that are important to our discussion of traffic scheduling are explained in the paragraphs to follow.

1. **TS Information**

This parameter specifies the Traffic Stream. Its values ranges from 0 (lowest) to 7 (highest).

2. **Nominal MSDU size (L)**

Average size of the packets belonging to this TS

3. **Maximum MSDU size (M)**

Maximum size of the packets belonging to this TS

4. **Minimum Service Interval (mSI)**

Minimum interval between the start of two successive Service Periods (or start of a TXOP)

5. **Maximum Service Interval(MSI)**

Minimum interval between the start of two successive Service Periods

6. **Minimum Data Rate**

Lowest Data Rate required to transfer the packets of this TS

7. **Mean Data Rate (ρ)**

Average rate of traffic arrival for this TS

8. **Peak Data Rate**

Peak rate of traffic arrival for this TS

9. **Maximum Burst Size (MBS)**

Maximum size of the data burst that can be transmitted at the peak data rate

10. **Delay Bound**

Maximum delay allowed to transport a packet across the wireless interface (including queuing delay)

11. Minimum PHY rate (R)

Desired minimum Physical transmission rate specified for this TS

The following TSPEC parameter set express QoS expectation of a QSTA <User Priority, Minimum Data Rate, Mean Data Rate, Peak Data Rate, Maximum Burst Size, Minimum PHY Rate, Delay Bound>. Unspecified parameters are marked as zero to indicate the non-AP QSTA does not have a specific requirement for these parameters if the TSPEC request was issued.

Once the TSPEC request is received by the HC, it analyzes the TSPEC parameters and decides whether to admit the stream into network or not. This is known as the **Admission Control** process. If the stream is admitted, the HC schedules the delivery of downlink traffic and/or QoS CF-polls in order to satisfy the QoS requirements of the stream as specified in the TSPEC. Several scheduling algorithms can be used in the HC. This process is known as **Scheduling**. Several scheduling principals can be used in the HC. The performance of the HCCA is dependant on the choice of the admission control and the scheduling algorithm. We use Reference Scheduler for Constant Bit Rate traffic. The scheduled service interval for an admitted stream is calculated in two steps:-

1. Calculation of the scheduled Service Interval(SI), the time between any two consecutive TXOPs
2. Calculation of TXOP duration (TD) for a given SI

The Simple scheduler algorithm is discussed below

4.7 Simple Scheduler

The simple scheduler was included in the 802.11e D7.0 draft. Owing to its operation to assign static TXOPs at static intervals for all QSTAs, it performs well in scenarios with uniform *constant bit rate traffic* (CBR), where the QoS requirements are similar for all transmitted traffic. The schedule generated by any scheduler should meet

normative behavior. The normative behavior states that the HC shall grant every flow negotiated TXOP in SI.

Calculating SI

1. Find $m = \min\{MSI_1, MSI_2, \dots, MSI_n\}$

This is the minimum of all MST for all admitted streams.

2. Chose a number smaller than m that is a factor of the beacon interval.
3. This value is the scheduled SI (Service Interval) for all QSTA.

Example of a Simple Scheduler Operation

Consider a stream i is admitted to a network. The HC will allocate a TXOP for this stream. The beacon interval is 100ms and the scheduler uses a service interval of 50ms.

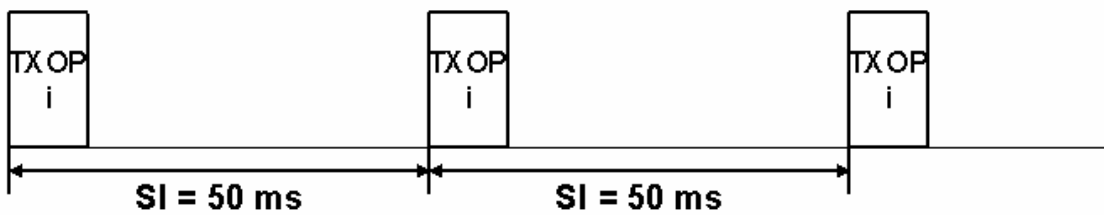


Figure 20 : Simple Schedule for a single stream i

The same process is repeated continuously if more than one stream is in network. Each stream is polled in round robin, and granted a specified TXOP duration according to the requirements of the stream. For example, if a new stream j is admitted to the network, the scheduling will now look like:

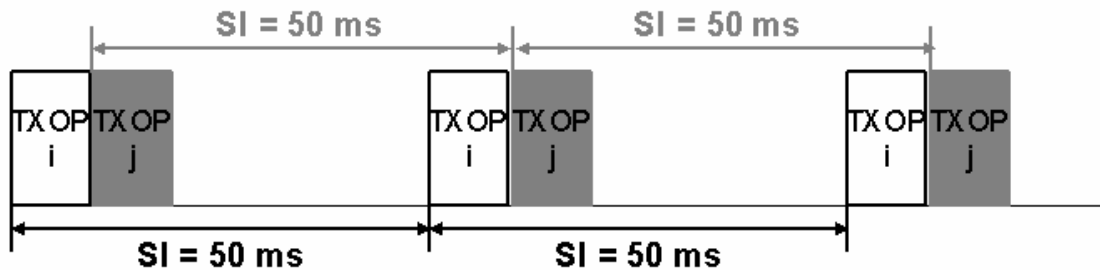


Figure 21 : Simple Schedule for two streams i & j

We discuss more about Simple Scheduler which is also called as HCCA Reference Scheduler.

4.8 HCCA Reference Scheduler

The HCCA scheme: In order to provide strict and parameterized QoS support regardless the traffic conditions, the HCF controlled channel access (HCCA) mechanism has been proposed by the 802.11e working group. HCCA uses a poll-and-response mechanism similar to PCF, but there are many differences between the two mechanisms. For example, HCCA is more flexible than PCF, i.e., QAP can start HCCA during both CFP and CP where PCF is only allowed in CFP. In addition, HCCA solves the three main problems of PCF.

(1) A direct link between peer stations is allowed in 802.11e, where stations can communicate each other without going through AP in HCCA.

(2) An 802.11e QSTA is not allowed to transmit a packet if the frame transmission cannot be finished before the next beacon, which solves the beacon delay problem with PCF.

(3) A *TXOPLimit* is used to bound the transmission time of a polled station, figure below shows example of an 802.11e beacon interval (the duration between two consecutive beacons), composed of alternated modes of optional CFP and mandatory CP. During CP, QAP is allowed to start several contention-free bursts, called controlled access period (*CAP*), at anytime after detecting channel as being idle for a time interval of PIFS. As shown in Figure 4.2, PIFS is shorter than DIFS and AIFSs, which gives a QAP a higher probability to start HCCA at any instant during a CP than other contending QSTAs. HCCA is more flexible than PCF because the latter must occur periodically after a beacon frame, while a QAP can initiate an HCCA whenever it wishes. Even if PCF is still allowed in 802.11e the flexibility of HCCA makes PCF useless. Thus, PCF is defined as optional in the 802.11e draft. After an optional period of CFP, the mechanisms of EDCA and HCCA which is used in CAP durations, alternate in a beacon interval. Although HCCA can provide more strict QoS support than EDCA, the latter is still mandatory in 802.11e for supporting QoS specification

exchange between QTSAs and QAP. For this purpose, the maximum duration of HCCA in an 802.11e beacon interval is bounded by a variable, *TCAPLimit*.

A simple HCF scheduling algorithm is suggested as a reference design in the 802.11e specification providing a parameterized QoS support based on the contract between QAP and corresponding QSTA(s). Before any data transmission,

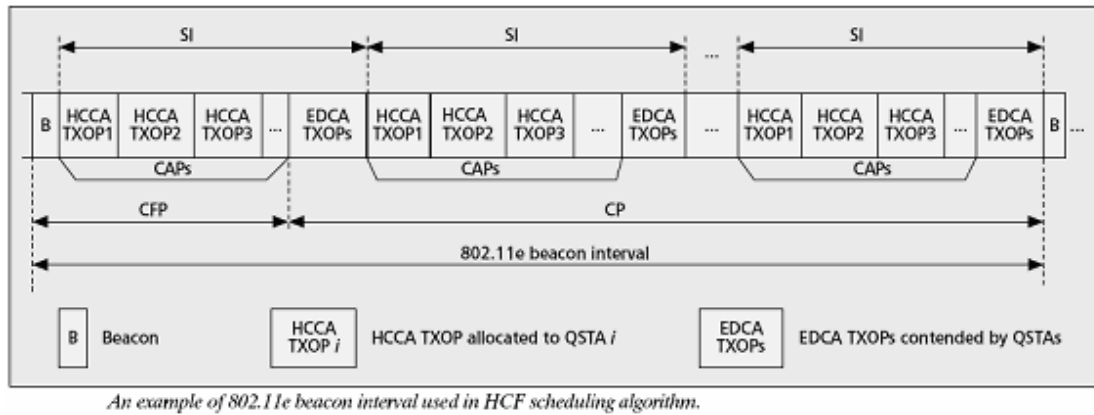


Figure 22: Beacon Interval in HCF scheduling algorithm

a traffic stream has first to be established and each QSTA is allowed to have no more than eight traffic streams with different priorities. Note that traffic streams and ACs are separated and use different MAC queues. In order to setup a traffic stream connection, a QSTA must send a QoS request frame containing the corresponding traffic specification (TSPEC) to the QAP1. A TSPEC describes the QoS parameter requirement of a traffic stream such as mean data rate, the maximum MAC service data unit (MSDU) size, the delay bound and the maximum Required Service Interval (RSI).

The maximum RSI refers to the maximum time duration between the start of successive *TXOPs* that can be tolerated by the application. Intuitively, there is a link between maximum RSI and the delay bound for a given traffic stream. Consequently, the IEEE 802.11e draft suggests that if both the maximum RSI and the delay bound are specified by a QSTA, the HCF simple scheduler only uses the maximum RSI for the calculation of *TXOP* schedule. The simple 802.11e HCF scheduling algorithm is summarized as follows:

On receiving all these QoS requests, the QAP scheduler first determines the minimum value of all the maximum RSIs required by the different traffic streams. Second, it chooses the highest sub-multiple value of the 802.11e beacon interval duration as the selected service interval (SI), which is less than the minimum of all the maximum RSIs. Third, the 802.11e beacon interval is cut into several SIs and QSTAs are polled accordingly during each selected SI. The selected SI refers to the time between the start of successive *TXOPs* allocated to a QSTA, which is the same for all the stations. As soon as the SI is determined, the QAP scheduler computes the different *TXOP* values allocated to the different traffic streams for different QSTAs, which are *TXOP1*, *TXOP2*, etc.,. Suppose the mean data rate request of the applications from traffic stream j in the QSTA i is $\rho_{i,j}$ and the nominal MSDU size for this queue is $M_{i,j}$ then the number of packets arriving in the traffic stream during the selected SI can be approximately computed as follows:

$$N_{i,j} = \left\lceil \frac{\bar{\rho}_{i,j} SI}{M_{i,j}} \right\rceil$$

Thus the QAP scheduler computes the allocated *TXOP*, $T_{i,j}$ for the traffic stream j in QSTA i as follows:

$$T_{i,j} = \max \left(\frac{N_{i,j} M_{i,j}}{R} + O, \frac{M_{\max}}{R} + O \right)$$

where R is the PHY layer transmission rate and M_{\max} is the maximum MSDU size (i.e. 2304 bytes). O refers to the transmission overheads due to PHY/MAC layer frame headers, IFSs, ACKs and poll frames. O is in time units and is computed as $2SIFS + TACK$ in this section.

Fourth, the QAP scheduler sums all the *TXOP* values of different traffic streams in a QSTA i as: $TXOP_i = \sum_{j=1}^{J_i} T_{i,j}$, where J_i is the number of active traffic streams in QSTA i . Then, the QAP scheduler allocates the time interval of $TXOP_i$ to QSTA i and allows the QSTA to transmit multiple frames during this time interval. In this way, the

QAP scheduler is supposed to allocate the corresponding $TXOP$ for transmitting all the arriving frames during the selected SI. Thus, the QAP scheduler is expected to control the delays.

An admission control algorithm is also suggested in the simple HCF scheduler: Using the above scheduling algorithm, the total fraction of transmission time reserved for HCCA of all K QSTAs in an 802.11e beacon interval can be computed as:

$$\sum_{i=1}^k \frac{TXOP_i}{SI}$$

. In order to decide whether or not a new request from a new traffic flow can be accepted in HCCA , the QAP scheduler only needs to check if the new request of $TXOP_{K+1}$ plus all the current $TXOP$ allocations are lower than or equal to the maximum fraction of time that can be used by HCCA:

$$\frac{TXOP_{K+1}}{SI} + \sum_{i=1}^k \frac{TXOP_i}{SI} \leq \frac{T_{CAPLimit}}{T_{Beacon}}$$

where $T_{CAPLimit}$ is the maximum duration bound of HCCA and T_{Beacon} represents the length of a beacon interval. Each QSTA is polled once per SI according to the HCF scheduling algorithm. This scheduling algorithm assumes that all types of traffic are CBR, so the queue length increases linearly according to the constant application data rate.

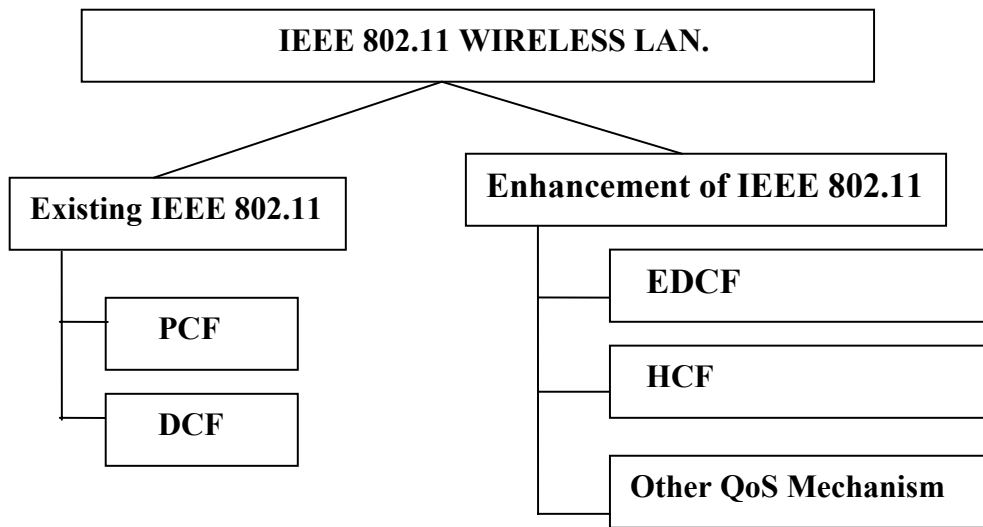


Figure 23: Classification of 802.11 and 802.11e protocols

Chapter 5

HCCA NETWORK SIMULATOR

5.1 INTRODUCTION

NS (version 2) is an object-oriented, discrete event driven network simulator developed at UC Berkeley written in C++ and OTcl. NS is primarily useful for simulating local and wide area networks. It implements network protocols such as TCP and UDP, traffic source behavior such as FTP, Telnet and CBR, router queue management mechanism such as Drop Tail, RED and CBQ, routing algorithms such as AODV, DSR, and more. NS also implements multicasting and some of the MAC layer protocols for LAN simulations. The purpose of this section is to give some basic idea of how the simulator works and how to setup a simulation network.

5.2 GENERAL ARCHITECTURE OF NS

NS is a discrete event simulator written in C++, with an OTcl interpreter as a front-end. The simulator supports a class hierarchy in C++ (we also call it the compiled hierarchy), and a similar class hierarchy within the OTcl interpreter (we also call it the interpreted hierarchy). The two hierarchies are closely related to each other. From the user's perspective, there is a one-to-one correspondence between a class in the interpreted hierarchy and one in the compiled hierarchy. The root of this hierarchy is the class TclObject. Users create new simulator objects through the interpreter; these objects are instantiated within the interpreter, and are closely mirrored by a corresponding object in the compiled hierarchy. The interpreted class hierarchy is automatically established through methods defined in the class TclClass. User instantiated objects are mirrored through methods defined in the class TclObject.

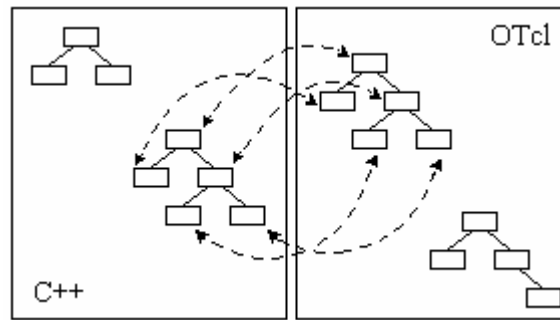


Figure 24: C++/OTcl Duality

NS uses two languages because the simulator has two different kinds of things it needs to do. On the one hand, detailed simulations of protocols require a systems programming language which can efficiently manipulate bytes, packet headers, and implement algorithms that run over large data sets. For these tasks, the run-time speed is important and the turn-around time is less important. On the other hand, a large part of network research involves slightly varying parameters or configurations, or quickly exploring a number of scenarios. In these cases, the iteration time is more important. Since configuration runs once (at the beginning of the simulation), the run-time of this part of the task is less important. C++ is fast to run but slower to change, making it suitable for detailed protocol implementation. OTcl runs much slower but can be changed very quickly (and interactively), making it ideal for simulation configuration.

When a simulation is finished, NS produces one or more text-based output files that contain detailed simulation data, if specified to do so in the input OTcl script. The data can be used for simulation analysis or as an input to a graphical simulation display tool called Network Animator (NAM). NAM has a nice graphical user interface that can graphically present information such as throughput and number of packet drops at each link, although the graphical information cannot be used for accurate simulation analysis.

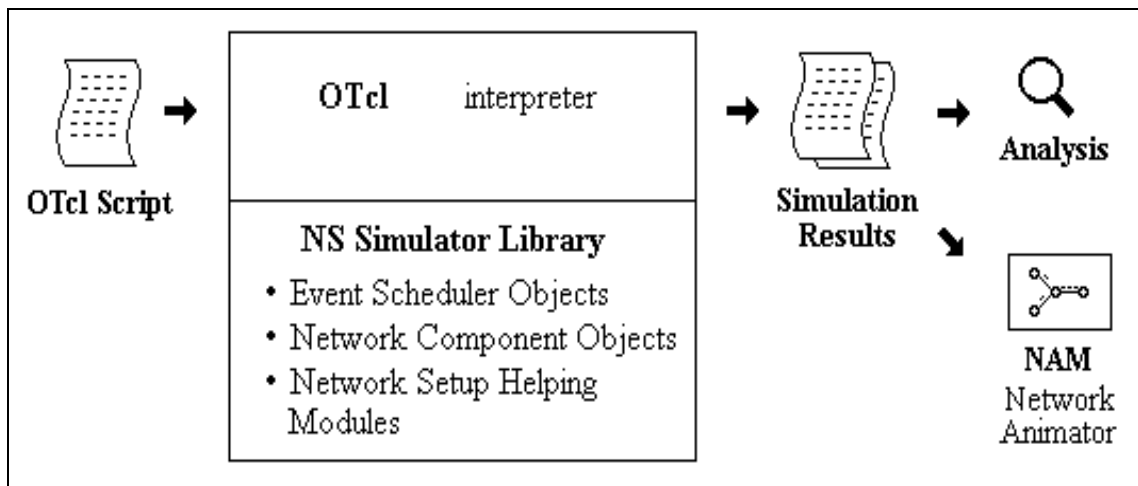


Figure 25: Simplified View of NS

We now briefly examine what information is stored in which directory or file in ns-2.

ns Directory Structure

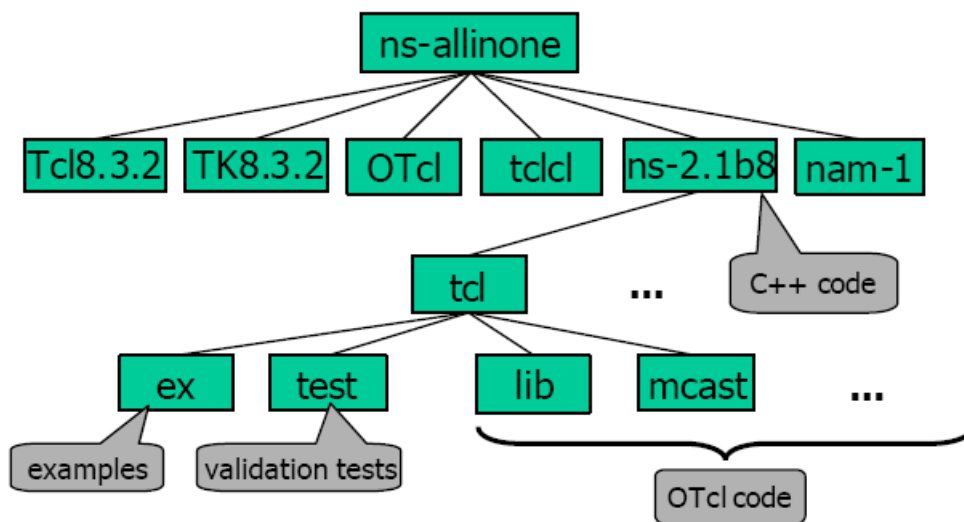


Figure 26: Directory Structure of NS

Among the sub-directories of ns-allinone-2.27, ns-2 is the place that has all of the simulator implementations (either in C++ or in OTcl), validation test OTcl scripts and example OTcl scripts. Within this directory, all OTcl codes located under a sub-directory called tcl, and most of C++ code, which implements event scheduler and basic network component object classes are located in the main level.

5.3 EXAMPLE OF SIMULATION SCRIPT

We now present a example simulation script and explain what each line means. Consider the following network topology of Figure 24.

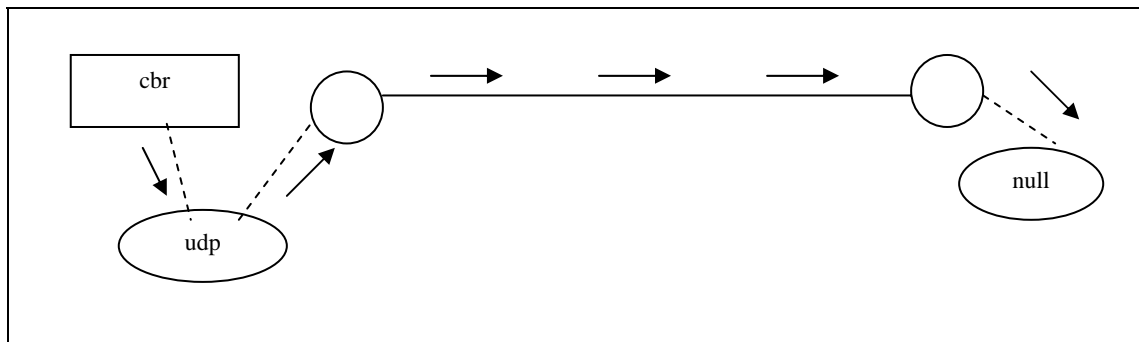


Figure 27: Scenario for a simple Simulation Script

We have two nodes; n0 and n1. The duplex link between n0 and n1 has 2 Mbps of bandwidth and 10 ms of delay. Each node uses a DropTail queue, of which the maximum size is 10. A “udp” agent that is attached to n0 is connected to a “null” agent attached to n1. A “null” agent just frees the packets received. A “cbr” traffic generator is attached to the “udp” agent, and the “cbr” is configured to generate a 500 byte packet every 1 second. The “cbr” is set to start at 0.5 second and stop at 4.5 second.

A simple NS simulation tcl script

```
# Create simulator object
set ns [new Simulator]
# Open the NAM trace file
```

```

set nf [open out.nam w]
$ns namtrace-all $nf
#Define a finish procedure
proc finish {} {
global ns nf
$ns flush-trace
close $nf
exec nam out.nam &
exit 0
}
#Create 2 nodes
set n0 [$ns node]
set n1 [$ns node]
#Create link between the nodes
$ns duplex-link $n0 $n1 1Mb 10ms DropTail
#Setup a UDP agent and attach it to node n0
set udp0 [new Agent/UDP]
$ns attach-agent $n0 $udp0
set null0 [new Agent/Null]
$ns attach-agent $n1 $null0
$ns connect $udp0 $null0
#Setup a CBR over UDP connection
set cbr0 [new Application/Traffic/CBR]
$cbr0 set packetSize_ 500
$cbr0 set interval_ 1.0
$cbr0 attach-agent $udp0
#Schedule events for CBR agent
$ns at 0.5 "$cbr0 start"
$ns at 4.5 "$cbr0 stop"

#Call finish procedure after 5 seconds of simulation time
$ns at 5.0 "finish"

```


#Run the simulation

```
$ns run
```

The following is the explanation of the script above.

- **set ns [new Simulator]** generates a new NS simulator object instance, and assigns it to a variable `ns`. This line creates an event scheduler for the simulation, initializes the packet format and selects the default address format.
- **\$ns namtrace-all nf** tells the simulator to record simulation traces in NAM input format. *nf* is the file name that the trace will be written to later by the command *\$ns flush-trace*.
- **proc finish {}** closes the trace file and starts nam.
- **set n0 [\$ns node]** creates a node. A node in NS is compound object made of address and port classifiers (described in a later section).
- **\$ns duplex-link node1 node2 bandwidth delay queue-type** creates two simplex links of specified bandwidth and delay, and connects the two specified nodes.
- **set udp [new Agent/UDP]** creates a udp agent. Users can create any agent or traffic sources in this way.
- **\$ns attach-agent node agent** attaches an agent object created to a node object.
- **\$ns connect agent1 agent2** connects the two agents specified. After two agents that will communicate with each other are created, the next thing is to establish a logical network connection between them. This line establishes a network connection by setting the destination address to each others' network and port address pair.
- **\$ns at 4.5 "\$cbr start"** tells the CBR agent when to start sending data.
- **\$ns at 0.5 "\$cbr stop"** tells the CBR agent when to stop sending data.
- **\$ns at 5.0 "finish"** tells the simulator object to execute the 'finish' procedure after 5.0 seconds of simulation time
- **\$ns run** starts the simulation.

5.4 Mac Layer

Depending on the type of physical layer, the MAC layer must contain a certain set of functionalities such as: carrier sense, collision detection, collision avoidance, etc. Since these functionalities affect both the sending and receiving sides, they are implemented in a single Mac object. For sending, the Mac object must follow a certain medium access protocol before transmitting the packet on the channel. For receiving, the MAC layer is responsible for delivering the packet to the link layer.

The IEEE 802.11 distributed coordination function (DCF) Mac protocol has been implemented by 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.

The following API configures for a mobile node with all the given values of adhoc-routing protocol, network stack, channel, topography, propagation model, with wired routing turned on or off (required for wired-cum-wireless scenarios) and tracing turned on or off at different levels (router, mac, agent).

`$ns_ node-config -adhocRouting $opt(adhocRouting) # dsdv/dsr/aodv/tora`

- **`llType $opt(ll)`** # specifies link layer object
- **`macType $opt(mac)`** # specifies mac object
- **`ifqType $opt(ifq)`** # specifies ifq object
- **`ifqLen $opt(ifqlen)`** # specifies length of ifq
- **`antType $opt(ant)`** # specifies antenna object
- **`propInstance [new $opt(prop)]`** # propagation object
- **`phyType $opt(netif)`** # specifies physical layer object
- **`channel [new $opt(chan)]`** # specifies channel object
- **`topoInstance $opto`** # specifies topography
- **`wiredRouting OFF`** # for wired cum wireless simulations
- **`agentTrace ON`** # specifies agent level trace ON/OFF
- **`routerTrace OFF`** # specifies router level trace ON/OFF
- **`macTrace OFF`** # specifies mac level trace ON/OFF

A mobile node is created using the following procedure:

```

for { set j 0 } { $j < $opt(nn) } { incr j } {
    set node_($j) [ $ns_ node ]
    $node_($i) random-motion 0    ;#disable random motion
}

```

This procedure creates a mobile node (split)object, creates an ad hoc routing agent as specified, creates the network stack consisting of a link layer, interface queue, Mac layer, and a network interface with an antenna, uses the defined propagation model, interconnects these components and connects the stack to the channel.

5.5 NS2 HCCA CLASS DIAGRAM

We now present a brief description of the classes that we have extended in order to improve the HCCA:

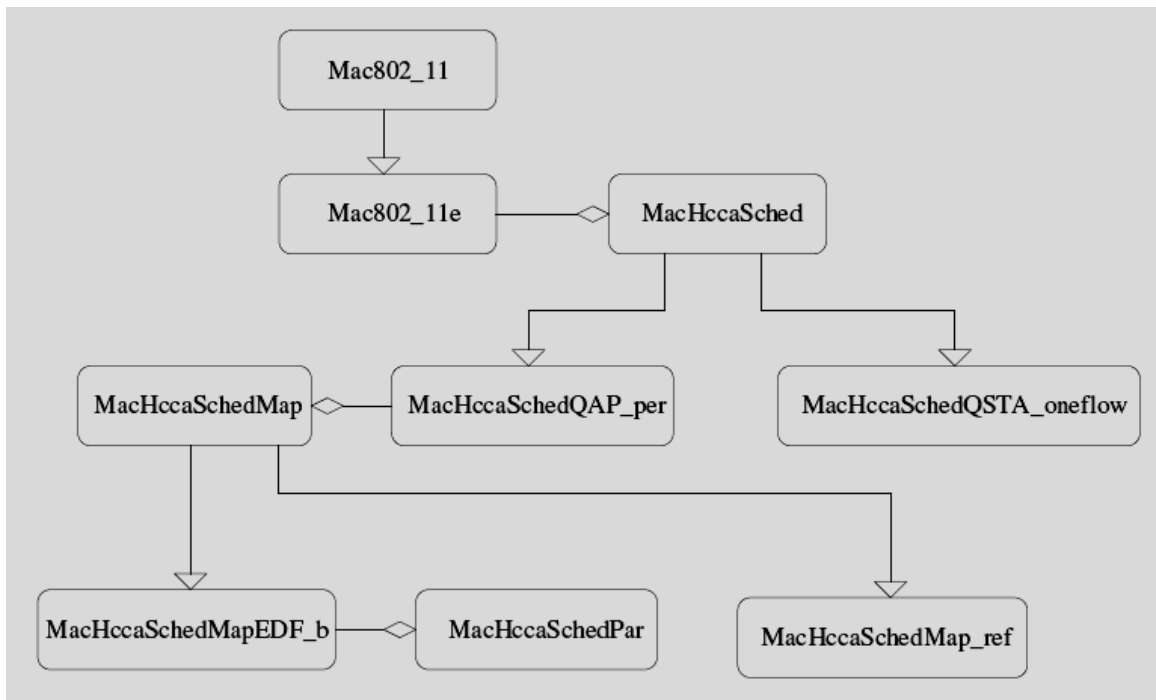


FIG 28: NS2HCCA Class Diagram

_ **Mac802 11** - The functionalities of the legacy MAC layer are implemented by the class Mac802 11. We have extended this class so that it can manage also the new QoS types.

_ **Mac802 11e** - The class is used to extend the legacy MAC layer in order to implement the HCCA function used by the EDF-B scheduler.

_ **MacHccaSched** - We have realized a common interface containing all the functions that should be implemented by any scheduler. This way, we have isolated the scheduler and the MAC layers. A class, which represents a scheduler, should be derived from this class and should implement its virtual methods complying with the given specifications.

_ **MacHccaSchedQSTA oneflow** - This class represents the scheduler used by the QSTAs. It is called oneflow because it manages at most one TS per direction. This choice permits to focus on the performance of the QAP scheduler.

_ **MacHccaSchedQAP periodic** - Both the implemented QAP schedulers exploit an `o_line` builded schedule to serve the TSs at run-time. The online functionalities of the two schedulers are exactly the same, once they are provided with a structure representing the `o_line` schedule. This class is therefore used to implement those common functionalities that are required at run-time. It uses the `o_line` schedule produced by the class MacHccaSchedMap to serve the TSs.

_ **MacHccaSchedMap** - This class provides the MacHccaSchedQAP periodic class with an interface to a data structure that may represent any periodic `o_line` schedule.

_ **MacHccaSchedMap ref** - This class specializes MacHccaSchedMap to produce the `o_line` schedule needed by the reference scheduler. The main data structure is a list of TXOPs whose elements specify the start time and duration of those TXOPs.

_ MacHccaSchedPar - This class is used to perform the parameters mapping procedure needed by EDF-B. We have not included this procedure directly in the class MacHccaSchedMapEDF B because the o_line schedule and the parameters mappings are independent.

5.6 Events Used in HCCA

_ HCCA DATA RECV - This event is used to notify the QAP and QSTA schedulers that a QoS Data(+) frame has been correctly received.

_ HCCA SUCCESS - This event is used to notify the QAP and QSTA schedulers that an acknowledgment to a previously sent frame has been correctly received.

_ HCCA TX END - This event is noti_ed to the scheduler of a station that has sent a QoS frame as soon as the transmission ends and hence when the timer mhTxEnd expires.

_ HCCA HCCATX HAND - This event is noti_ed to the scheduler when the timer mhHCCATxTimer expires and there is the opportunity, for the QoS station, to transmit a frame.

_ HCCA CAP HAND - This event is noti_ed to the scheduler when the timer mhCap expires indicating the start of a new CAP. It is noti_ed only at the QAP. After the timer expiration the QAP has to sense the medium idle for PIFS before gaining the control of the medium.

_ HCCA RECV - This event is noti_ed to the scheduler when any frame destined to this station is received and it is detected the PHY-RXEND indication.

_ HCCA TRANSMIT - This event is noti_ed to the scheduler when the station transmits a frame and it is detected the PHY-TXSTART indication.

_ HCCA MAC COLL - This event is notified to a QSTA when a collision of a transmitted frame is detected at the MAC layer.

_ HCCA MAC ERR - This event is notified to a QoS station that receives incorrectly a frame. The MAC layer check the function error(), which belongs to the common header, to know if the frame is corrupted.

Chapter 6

Simulation Results

HCCA patch was installed under Network Simulator-2.27. We ran Tcl Script for simulating the Reference Scheduler behaviour under different conditions. The o/p of the Tcl scripts were Trace files, Trace Files are the sources for data extraction after the successful simulation of the Tcl Script. Perl Scripts are used to analyze the Trace files. The values extracted are then used to plot graphs under Xgraph so as to evaluate throughput and hence the performance of the protocol.

The PERL script and Trace files are included in appendix.

Models and Assumptions

For QoS studies, the simulation is modeled as an infrastructure mode where one AP exchanges sequences with multiple stations and each station has only one bidirectional traffic stream. Two real-time services are considered in this study. The voice and video services are modeled as a **Constant Bit Rate** of 64 Kbps (160 bytes per 20 ms frame) and 1 Mbps (1280 bytes per 10 ms)

Performance Metrics

The performance metrics considered in the simulation are defined as follows:

Delay: The time between the arrival of a packet (entering the MAC) and the successful departure of the packet.

Throughput: The fraction of discarded packet caused by transmission violating the delay bound.

Capacity: The number of the stations (at a fixed transmission rate) that the system can support.

Results

The simulations will be quantified based on following Scenarios.

Scenario 1: The number of stations considered in wireless environment are ranging from 5 to 8 .Each has two flows uplink and downlink. Throughput for all the QSTA's each with two flows are simulated and the average throughput is calculated Figure 1 shows the simulation results with number of flows on X axis and corresponding Throughput on Y axis. Trace files and other simulation results are mentioned in appendix.

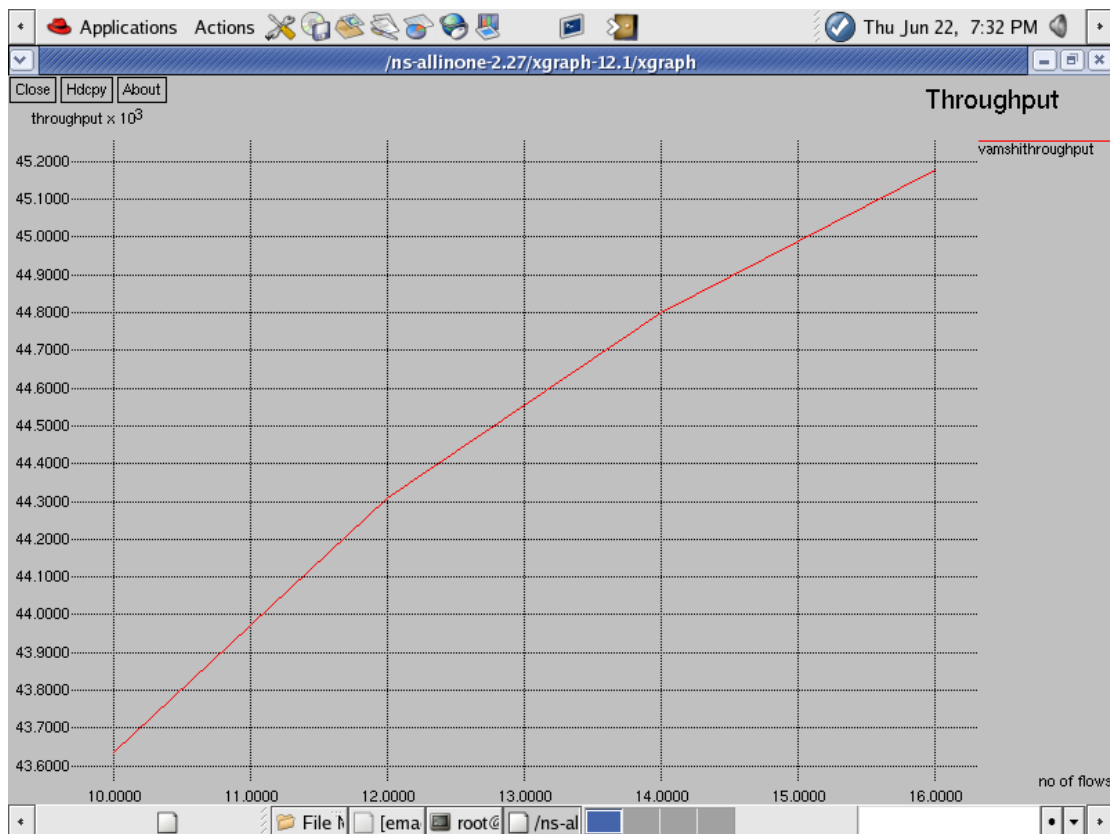


Fig. 29: Number of flows vs throughput (for 5 to 8 stations)

Scenario 2: The number of stations considered in wireless environment are ranging from 8 to 11 .Each has two flows uplink and downlink. Throughput for all the QSTA's each with two flows are simulated and the average throughput is calculated Figure 2 shows the simulation results with number of flows on X axis and corresponding Throughput on Y axis. Trace files and other simulation results are mentioned in appendix.

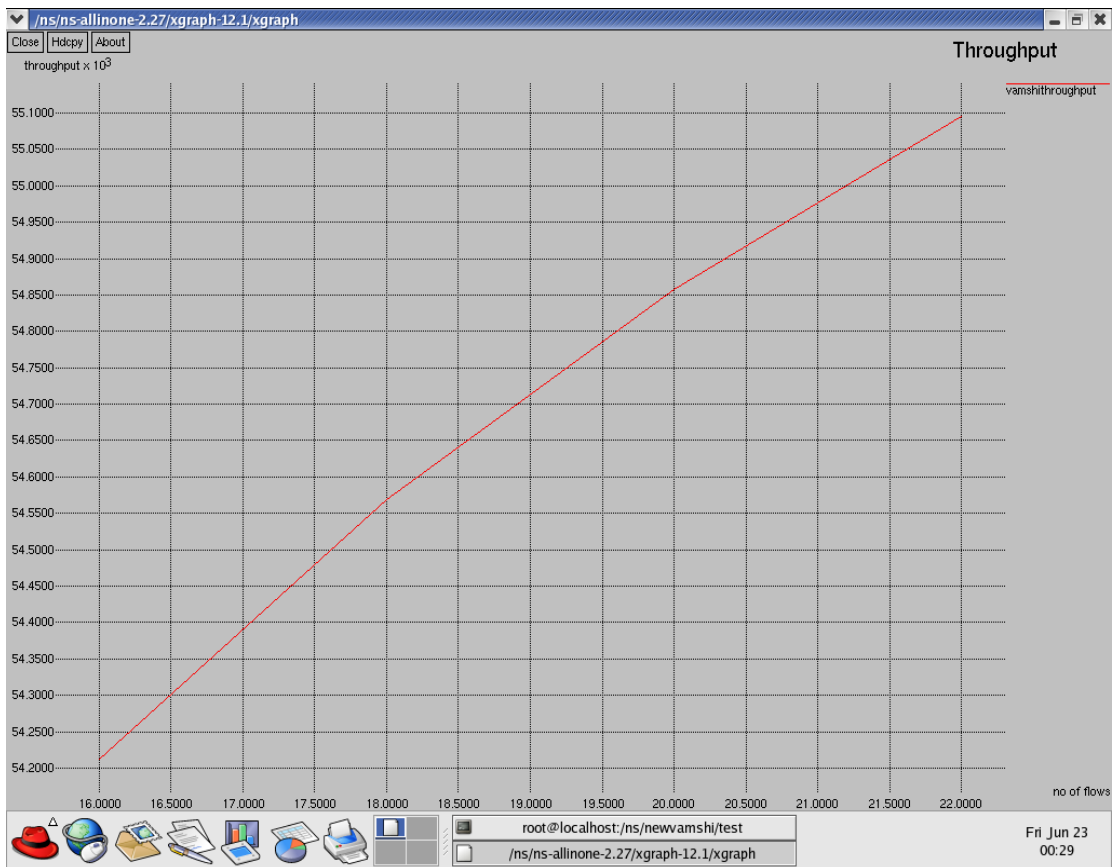


Fig. 30: Number of flows vs throughput (for 8 to 11 stations)

Scenario 3: The number of stations considered in wireless environment are ranging from 5 to 11 .Each has two flows uplink and downlink. Throughput for all the QSTA's each with two flows are simulated and the average throughput is calculated Figure 3 shows the simulation results with number of flows on X axis and corresponding Throughput on Y axis. Trace files and other simulation results are mentioned in appendix.

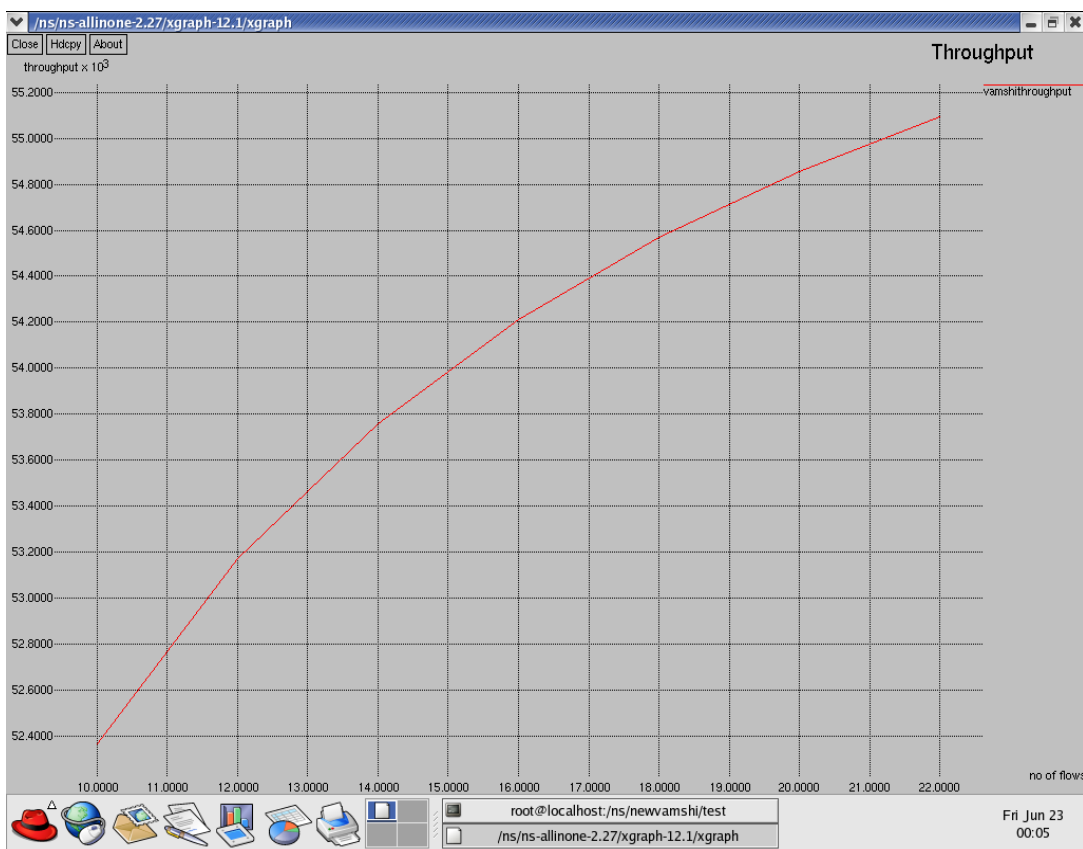


Fig. 31: Number of flows vs throughput (for 5 to 11 stations)

Observations

The results prove that with the increase in the number of flows the throughput increases linearly to some extent (here 22 flows) after which the performance starts degrading because of loss of Admission Control. This infers that the Reference scheduler has consistent results but to some possible extent only.

Conclusion and Future Scope

Conclusion

As we know the most popularly and widely used Wireless LAN standard all over the world today is IEEE 802.11, but it lacks proper quality of service (QoS) support for multimedia applications. The present study makes much familiar with functioning of PCF Protocol and its shortcomings hence need for QoS support which are provided by HCCA Protocol. HCCA with Reference Scheduler is used to provide QoS for Multimedia Applications. The Simulation and thus Performance of the HCCA protocol with Reference Scheduler is realized from the results obtained. All the simulations are done using Network Simulator2 in LINUX 9.0.

Future Scope

Further enhancements can be made to improve Quality of Services for WLans with HCCA protocol. This scheduler assumes that all types of traffic are CBR, so the queue length increases linearly according to the constant application data rate. However, a lot of real-time applications, such as videoconferencing, have variable bit rate (VBR) characteristics. The simple HCCA scheduler may cause the average queue length to increase and may possibly drop packets. Even if the mean transmission rate of the application is lower than the rate specified in QoS requirements, peaks of transmission rate may not be absorbed by *TXOPs* allocated according to the QoS requirements. Some adaptive schemes that take into account fluctuation of traffic transmission rates are thus necessary.

Appendix A

PERL SCRIPT

#This perl script is used to analyze the trace files generated by the linking of network simulator2 and HCCA Reference Scheduler with TCL scripts.

```
#!/usr/bin/perl
```

```
$first=1;
$starttime=0;
$finishtime=0;
#initialize starttime and finishtime with 0
```

```
$throughputvamshi=$ARGV[0];
# command line argument for throughput
```

```
print "Throughput file: $throughputvamshi\n";
#while input available from stdin which is stored in special variable $_
while ( <STDIN> ) {
#
```

```
    open(vamshiout,">>$throughputvamshi") || die "Can't open $throughputfile\n";
#opening the out file throughputvamshi
```

```
    if ( $_ =~ /agent/ ) {
#checking the condition whether agent or not
```

```
        #pattern matching....checking for agent in the current line
        ($event , $other) = split(/,/, $_, 2);
#splitting values into table format
```

```
        my %log=();
```

```
        while ( $other ) {
            ($tag, $value, $other) = split //, $other, 3;
#            print "Now: $tag, $value \n";
            $log{$tag} = $value;
        }
```

```
# Now all the variables are stored in log. To find a particular value use the Index as the
```

```
#Outhroughput parameter used in trace file. e.g. To find the flow id, use $log{"-If"}
```

```
# extract flow-id,packet-id,time,packet-size
```

```
    my $flow = $log{"-If"};
    my $id = $log{"-Ii"};
```

```

my $time = $log{"-t"};
my $size = $log{"-l"};

$starttime = $time if $first;
$first = 0 if $first;
$finishtime = $time;
#assign starttime with time, first with 0 and finishtime with time

if ( $event eq 's' and $stat[$flow][$id] == 0 ) {
#if event is equal to send and stat[flow][id] is zero

    $delay[$flow][$id] = $time;
    #assign delay[flow][id] with time

    $stat[$flow][$id] = 1;
    #assign stat[flow][id] with 1
} elsif ( $event eq 'r' and $stat[$flow][$id] == 1 ) {
#if event is equal to receive and stat[flow][id] is one

    $throughput[$flow] += $size;
    #add size to old throughput[flow]
    $stat[$flow][$id] = 2;
    #assign stat[flow][id] with 2

    }
}

}

$duration = $finishtime - $starttime;
#calculate duration which is the difference between finishtime and starttime

$n_flows = 0;
$avghthroughput=0.0;
#assign number of flows and average throughput with 0

for my $flow ( 0 .. $#delay ) {

# repeat the loop for flow ranging from 0 to delay

    $throughput[$flow] = 8.0 * $throughput[$flow] / $duration;

```

```

        #converting throughput[flow] from bytes to bits
        $avgthroughput+=$throughput[$flow];

#calculate the averagethroughput for number of flows
    ++$n_flows;

#incrementing number of flows

    print "flow $flow: throughput ".
        $throughput[$flow] ."bps\n";
        #printing throughput for each flow in bits per sec
}

$avgthroughput/=$n_flows;
#calculating averagethroughput for all the stations with two flows each(uplink and
#downlink)

print vamshiout "$#delay ";
print "Avg. Throughput: ".$avgthroughput."\n";
print vamshiout $avgthroughput ."\n";
#display the delay, average throughput and store the averagethroughput in the output
file (vamshiout).

```

TRACE FILES

```

# for QSTA 5,number of flows =10.
Throughput file: vamshithroughput
flow 0: tput 10.08bps
flow 1: tput 57585.6bps
flow 2: tput 38400bps
flow 3: tput 57600bps
flow 4: tput 38400bps
flow 5: tput 57600bps
flow 6: tput 38400bps
flow 7: tput 57600bps
flow 8: tput 38400bps
flow 9: tput 57600bps
flow 10: tput 38400bps
Avg. Throughput: 43635.9709090909

```

```

# for QSTA 11 number of flows =22
Throughput file: vamshithroughput
flow 0: tput 20.16bps
flow 1: tput 57585.6bps
flow 2: tput 38400bps
flow 3: tput 57600bps
flow 4: tput 38400bps
flow 5: tput 57600bps
flow 6: tput 38400bps
flow 7: tput 57600bps
flow 8: tput 38400bps
flow 9: tput 57600bps
flow 10: tput 38400bps
flow 11: tput 57600bps
flow 12: tput 38400bps
flow 13: tput 57600bps
flow 14: tput 38400bps
flow 15: tput 57600bps
flow 16: tput 38400bps
flow 17: tput 57600bps
flow 18: tput 38400bps
flow 19: tput 57600bps
flow 20: tput 38400bps
flow 21: tput 57600bps
flow 22: tput 38400bps
Avg. Throughput: 45913.2939130435

```

Numeric Results of fig 31.

Number of flows	Throughput
10	43635.9709090909
12	44307.4892307692
14	44799.936
16	45176.5129411765
18	45473.8105263158
20	45714.48
22	45913.2939130435

References

1. Lassaâd Gannoune, Stephan Robert and Daniel Rodellar , “**A Survey of QoS Techniques and Enhancements for IEEE 802.11 Wireless LANs**”.
2. Qiang Ni, Lamia Romdhani, Thierry Turlletti, “**A Survey of QoS Enhancements for IEEE 802.11 Wireless LAN**”.
3. Stefan Mangold, Sunghyun Choi, Peter May, Ole Klein, Guido Hiertz, Lothar Stibor, “**IEEE 802.11e Wireless LAN for Quality of Service**”.
4. **Providing QoS in WLANs - How the IEEE 802.11e Standard QoS Enhancements Will Affect the Performance of WLANs**, Whitepaper by Intel.
5. Takehiro Kawata, Sangho Shin, Andrea G. Forte, “**Using Dynamic PCF to Improve the Capacity for VoIP Traffic in IEEE 802.11 Networks**”.
6. Anders Lindgren, Andreas Almquist and Olov Schelén, “**Quality of Service Schemes for IEEE 802.11 Wireless LANs – An Evaluation**”.
7. **IEEE Std. 802.11, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”, 1999.**
8. S. Xu, “**Advance in WLAN QoS for 802.11: an Overview**”, PIMRC, 2003.
9. W. Pattara-Atikom, P. Krishnamurthy and S. Banerjee, “**Distributed Mechanisms for Quality of Service in Wireless LANs**”, IEEE Wireless Communication, 2003.
10. W. Liu, W. Lou, X. Chen and Y. Fang, “**A QoS-enabled MAC Architecture for Prioritized Service in IEEE 802.11 WLANs**”, GLOBECOM, 2003.
11. A. Grilo and M. Nunes, “**Performance Evaluation of IEEE 802.11E**”, PRIMC, 2002.
12. D. He and C. Q. Shen, “**Simulation Study of IEEE 802.11e EDCF**”, VTC, 2003.
13. D. Chen D. Gu and J.Zhang, “**Supporting Real-time Traffic with QoS in IEEE 802.11e Based Home Networks**”, CCNC, 2004.
14. Q. Ni and T. Turlletti, “**QoS Support for IEEE 802.11 Wireless LAN**”.
15. **The ns Manual Notes and Documentation**, Kevin Fall and Kannan Varadhan.

Web References

1. www.isi.edu/nsnam/ns/index.html
2. www.info.iet.unipi.it/~cng/
3. http://www.tkn.tu-berlin.de/research/802.11e_ns2/
4. <http://www.monarch.cs.rice.edu>
5. <http://web.syr.edu/~dchen02/resources.htm>
6. www.ieeexplore.ieee.org/

List of Abbreviations

A

AC	Access Category
ACM	Admission Control Mandatory
ACU	Admission Control Unit
AIFS	Arbitrary Interframe Spacing
AMBS	Aggregate Maximum Burst Size
AP	Access Point
ARROW	Adaptive Resource Reservation Over WLAN

B

BSA	Basic Service Area
BSS	Basic Service Set

C

CAP	Controlled Access Period
CBR	Constant Bit Rate
CC	Controlled Contention
CCI	Controlled Contention Interval
CCOP	Controlled Contention Opportunity
CF	Contention Free
CFB	Contention Free Burst
CFP	Contention Free Period
CP	Contention Period
CSMA/CA	Carrier Sense Multiple Access / Collision Avoidance
CW	Contention Window

D

DCCOP	Duration of Controlled Contention Opportunity
DCF	Distributed Coordination Function
DIFS	DCF Interframe Space
DS	Distribution System

E

EDCA	Enhanced Distributed Channel Access
EIFS	Extended Interframe Space
ESS	Extended Service Set

H

HCCA	HCF Controlled Channel Access
HCF	Hybrid Control Function

I

IBSS	Independent Basic Service Set
IEEE	Institute of Electrical and Electronics Engineers
IFS	Interframe Space

M

MSI	Minimum Service Interval
MTD	Minimum TXOP Duration
MAC	Medium Access Control
MBS	Maximum Burst Size
MPDU	MAC Protocol Data Unit
MSDU	MAC Service Data Unit
MSI	Maximum Service Interval
MTD	Maximum TXOP Duration

N

NAV	Network Allocation Vector
NCCOP	Number of Controlled Contention Opportunity
NOS	Network Operating System

P

PC	Point Coordinator
PCF	Point Coordination Function
PIFS	PCF Interframe Space
PRBAC	Physical Rate Based Admission Control

Q

QAP	QoS Access Point
QBSS	QoS Basic Service Set
QoS	Quality of Service
QS	Queue Size
QSTA	QoS Station

R

RF	Radio Frequency
RR	Reservation Request

S

SETT-EDD	Scheduling based on Estimated Transmission Times–Earliest Due Date
SI	Service Interval
SIFS	Short Interframe Space
STA	Station

T

TBTT Target Beacon Transmission Time
TD TXOP Duration
TGe Task Group e
TID Traffic priority identifier
TS Traffic Stream
TSPEC Traffic Specification
TXOP Transmission Opportunity

U
UP User Priority

V
VBR Variable Bit Rate
VoIP Voice over Internet Protocol

W
WLAN Wireless Local Area Network
WSTA Wireless Station

