

**GUEFRY LEIDER AGREDO MÉNDEZ**



**SELECCIÓN DE VALORES DE PARÁMETROS EDCA  
EN UN QBSS DE VoWLAN**

**Universidad Del Cauca  
Facultad de Ingeniería Electrónica y Telecomunicaciones  
Maestría en Ingeniería  
Área: Electrónica y Telecomunicaciones**

**Popayán  
2008**

**GUEFRY LEIDER AGREDO MÉNDEZ**

**SELECCIÓN DE VALORES DE PARÁMETROS EDCA  
EN UN QBSS DE VoWLAN**

**Tesis presentada a la Facultad de Ingeniería Electrónica y  
Telecomunicaciones de la Universidad del Cauca para la obtención  
del Título de Magíster en Ingeniería.**

**Área:  
Electrónica y Telecomunicaciones.**

**Director:  
Magíster. Ing. JOSE GIOVANNY LÓPEZ PERAFÁN**

**Popayán  
2008**

# Hoja de Aprobación

**Universidad del Cauca**  
Facultad de Ingeniería Electrónica y  
Telecomunicaciones  
CI-FIET

**FORMATO I: ACTA DE SUSTENTACIÓN**

LOS JURADOS DE LA TESIS DE MAESTRÍA TITULADA:

**Selección de Valores de Parámetros EDCA en un QBSS de VoWlan**

HACEN CONSTAR:

Que siendo las 8:00 a.m. del día 10 del mes de octubre de 2008 realizaron la Sustentación del Trabajo de Grado (el) (los) señores (as) **GUEFRY LEIDER AGREDO MÉNDEZ**

**OBTENIENDO EL CONCEPTO DE: APROBADO ( X ) NO APROBADO ( )**

Para constancia se firma en Popayán, a los 10 días del mes de octubre de 2008.

Bajo la dirección del Ingeniero: **M. Sc. GIOVANNY LOPEZ PERAFÁN**

JURADO COORDINADOR: \_\_\_\_\_  
(Nombre)

JURADO : Andrés Nolasco Pacheco  
(Firma)  
Orlando Nolasco  
(Nombre)  
(Firma)

**Universidad del Cauca**  
Facultad de Ingeniería Electrónica y  
Telecomunicaciones  
CI-FIET

**TRABAJOS DE GRADO**

**FORMATO I: ACTA DE SUSTENTACIÓN**

Selección de Valores de Parámetros EDCA en un QBSS de VoWlan

HACEN CONSTAR:

Que siendo las 8 A.M. del día 10 del mes de octubre de 2008 realizaron la Sustentación del Trabajo de Grado (el) (los) señores (as) ING. GUEFRY LEIDER AGREDO MÉNDEZ

**OBTENIENDO EL CONCEPTO DE: APROBADO ( X ) NO APROBADO ( )**

Para constancia se firma en Popayán, a los 10 días del mes de octubre de 2008.

Bajo la dirección del Ingeniero: Mag. **JOSE GIOVANNY LOPEZ PERAFÁN**

JURADO COORDINADOR: \_\_\_\_\_  
(Nombre)

JURADO : \_\_\_\_\_  
(Firma)

JURADO : **JHON JAIRO PADILLA AGUILAR, PH.D.**  
(Nombre)

John Jairo Padilla A.  
(Firma)

# Dedicatoria

*A Luisa que en su compañía  
encuentro amor pero en especial refugio  
A Santiago que con un solo abrazo  
me ilumina la vida y calma mis preocupaciones*

# Agradecimientos

A Dios que me inspira la fortaleza y el ánimo para ver cada nuevo día como una nueva oportunidad.

A mi familia por ser la principal motivación, y por tolerar con paciencia la gran cantidad de veces que fue necesario robarles tiempo para lograr esta meta.

A mi Director de Tesis por sus recomendaciones y sus sutiles insinuaciones para que no diera tregua.

Al Instituto de Postgrado por facilitar en la medida de las posibilidades la Sala del IPET para realizar la experimentación base de este proyecto y a todos los que con su colaboración hicieron posible llevar a cabo las pruebas.

# Resumen

Las redes inalámbricas cuentan en la actualidad con la especificación de multimedia WiFi – WMM, generada por la Alianza WiFi como un subconjunto, para implementación en equipos, del mecanismo de acceso al medio mejorado EDCA (*Enhanced Distributed Channel Access*) el cual hace parte del estándar IEEE 802.11e para proporcionar calidad de servicio por medio de la priorización de paquetes basada en marcas a nivel MAC. De esta forma existen cuatro categorías de acceso, que en orden de prioridad son: voz, video, mejor esfuerzo (*best effort*) y por último fondo (*background*). Dentro de la especificación existen cuatro parámetros que influyen en la probabilidad para que un tipo de tráfico pueda tomar el medio. Estos parámetros son: el AIFS que establece una espera fija, el par CWmin y CWmax que establecen límites a la espera aleatoria (*backoff*) y la oportunidad de transmisión TXOP que determina el tiempo durante el cual un flujo puede enviar la mayor cantidad de paquetes sin tener que realizar la contienda por el medio.

Con este proyecto, desarrollado principalmente con un enfoque experimental, se ha buscado primeramente evaluar el desempeño de una red inalámbrica que transporte tráfico de voz en el marco de implementación que se conoce como VoWLAN (*Voice over WLAN*) cuando se habilita el soporte para WMM/EDCA frente a una red que no lo soporte; se ha trabajado también la simulación como herramienta para la generación de resultados, cuando se ha considerado conveniente o necesario. En segundo lugar, partiendo del hecho que según la especificación WMM/EDCA los valores por defecto de los parámetros pueden modificarse para atender los cambios en la red, se han definido procedimientos para determinar valores que optimicen el desempeño de la red, con la intención de contribuir con el cumplimiento de los requisitos de VoIP cuando el tráfico de datos o de voz sea alto y para mejorar la velocidad efectiva (*throughput*) de las estaciones de datos cuando no se tiene tráfico de voz.

Los resultados demuestran que el mecanismo WMM/EDCA beneficia el desempeño del tráfico de tiempo real como el de VoIP, pero que es viable para un rendimiento óptimo, realizar un proceso de selección de valores de parámetros que permitan obtener un mejor desempeño.

**Palabras Clave:** WMM, QoS, EDCA, VoWLAN, Inalámbricas, Experimentación

# Abstract

At the moment, wireless networks with the WiFi Multimedia Specification - WMM, generated by the WiFi Alliance as a subset, for implementation in equipment, of an improved access to the medium, known as EDCA (Enhanced Distributed Channel Access) which is part of IEEE 802.11e standard to provide quality of service by prioritizing packets based on marks at the MAC layer. In this way there are four categories of access, that in order of priority are: voice, video, best effort and finally background. Within the specification there are four parameters that influence the probability for a type of traffic take the medium. These parameters are: the AIFS establishing a fixed amount of time, the CWmin and CWmax pair which set limits on the expected random backoff and the opportunity of transmission TXOP that determines a time during which a flow can send as many packages as it can, without having to make the content for the medium.

With this project, developed mainly in an experimental way, it has been focus firstly in make a performance test of a wireless network carrying voice traffic within the framework of implementation that is recognized as VoWLAN (Voice over WLAN) when WMM / EDCA is enabled compared to a wireless network that don't support it; a simulation tool has also used to generate results, but it has been used where it was considered desirable or necessary. Secondly, based on the fact that according to the WMM / EDCA specification, default values of the parameters can be modified to accommodate to the changes in the network It have been defined procedures for determining values that optimize the performance of the network, with the intention of contribute to compliance with the requirements of VoIP when data traffic or voice traffic is high and to improve the effective speed (throughput) of stations where there is no voice traffic.

The results show that the WMM / EDCA mechanism improve the performance of real-time traffic such as VoIP, but that is viable for optimum performance, make a parameter values selection process to obtain better performance.

**Keywords:** WMM, QoS, EDCA, VoWLAN, Wireless, Experimentation

# Contenido

Pág

Lista de Tablas

Lista de Figuras

1.	Introducción .....	13
1.1.	Antecedentes .....	13
1.2.	Justificación .....	14
1.3.	Metodología .....	15
1.4.	Alcances y Limitaciones.....	16
1.5.	Contribución.....	17
1.6.	Organización.....	20
2.	Funcionamiento de EDCA.....	21
2.1.	Estándar IEEE 802.11 Nivel MAC.....	21
2.1.1.	Las funciones de coordinación de 802.11: DCF y PCF .....	22
2.1.2.	Limitaciones de calidad de servicio en DCF y PCF .....	24
2.2.	Estándar IEEE 802.11e.....	25
2.2.1.	Provisión de QoS.....	25
2.2.2.	Modificación de las funciones de acceso .....	26
2.3.	EDCA: Acceso al Canal Distribuido Mejorado .....	27
3.	Evaluación de EDCA .....	37
3.1.	Análisis para la Evaluación .....	37
3.2.	Infraestructura para la Experimentación .....	38
3.3.	Desarrollo de la Evaluación .....	40
3.4.	Evaluación por Simulación.....	45
4.	Selección de Valores de Parámetros EDCA.....	51
4.1.	Análisis para la Generación de Hipótesis.....	51
4.2.	Infraestructura para la Experimentación .....	52
4.3.	Primer Escenario: Tráfico Alto de Datos .....	60
4.4.	Segundo Escenario: Tráfico Alto de VoIP .....	66

4.3.1. Selección por Simulación .....	71
4.5. Tercer Escenario: Tráfico Nulo de VoIP .....	76
Conclusiones y Recomendaciones .....	81

## Apéndices

A. Acceso Controlado al Canal HCF (HCCA) .....	83
B. Mecanismos Adicionales de QoS en el nivel MAC de IEEE 802.11e.....	85
C. Configuración de Asterisk para Generación de Flujos de VoIP con QoS.....	89
D. Scripts para la Infraestructura de Experimentación.....	93
E. Cambio de Valores de Parámetros con MadWiFi en Linux.....	95
F. Manejo del Simulador NCTUns 4.0.....	101
G. Modificación de Valores de Parámetros EDCA en NCTUns 4.0 .....	115
H. Metodología de Simulación.....	117
Referencias Bibliográficas.....	123

# Lista de Tablas

	Pág
Tabla 1.1. Clasificación y análisis de las principales referencias .....	19
Tabla 2.1. Parámetros MAC para 802.11a, b, g.....	23
Tabla 2.2. Mapeo de las categorías de tráfico sobre las categorías de acceso .....	27
Tabla 2.3. Mapeo entre categoría de acceso, prioridad de usuario y DSCP.....	31
Tabla 2.4. Explicación de parámetros EDCA. ....	32
Tabla 2.5. Valores de los parámetros predeterminados EDCA.....	34
Tabla 2.6. Duración de los AIFS según la categoría de acceso con EDCA .....	35
Tabla 3.1. Velocidad promedio global de las descargas sin EDCA y con EDCA. ....	44
Tabla 4.1. Velocidad promedio global de las descargas al variar el AIFS[AC_BE] .....	66
Tabla 4.2. Velocidad promedio de la descarga al variar el CWmax[AC_VO] .....	70
Tabla 4.3. Velocidad promedio global de las descargas al variar el AIFS[AC_BE] .....	78

# Lista de Figuras

	Pág.
Figura 2.1 Arquitectura MAC estándar 802.11 .....	21
Figura 2.2 Funcionamiento de DCF .....	22
Figura 2.3 Arquitectura MAC estándar 802.11e .....	27
Figura 2.4 Trama IEEE 802.3 con soporte para QoS de tipo 802.1Q y 802.1p.....	28
Figura 2.5 Trama IEEE 802.11 con soporte para QoS de tipo 802.11e. ....	28
Figura 2.6 Ventana de configuración del DSCP en X-Lite.....	29
Figura 2.7 Ventana de creación de una política de QoS en Windows Vista™ .....	30
Figura 2.8 Lógica de encolamiento de las categorías de acceso de EDCA .....	32
Figura 2.9 Prioridad de acceso al medio dependiente de la AC.....	33
Figura 2.10 Espera mínima y aleatoria de las categorías de acceso en EDCA. ....	34
Figura 3.1 Diagrama en bloques de la infraestructura para experimentación. ....	39
Figura 3.2 Evidencia de la marcación de las tramas de voz. ....	40
Figura 3.3 Resultado de un solo flujo de VoIP con 802.11b sin WMM/EDCA. ....	42
Figura 3.4 Resultado de un solo flujo de VoIP con 802.11b con WMM/EDCA.....	42
Figura 3.5 Resultado de 3 flujos de VoIP y Datos con 802.11b sin WMM/EDCA. ....	43
Figura 3.6 Resultado de 3 flujos de VoIP y Datos con 802.11b con WMM/EDCA. ....	43
Figura 3.7 Topología de simulación de una red 802.11b sin soporte para EDCA. ....	46
Figura 3.8 Topología de simulación de una red 802.11b con soporte para EDCA.....	46
Figura 3.9 Archivo de registro de la simulación sin soporte para EDCA. ....	49
Figura 3.10 Archivo de registro de la simulación con soporte para EDCA.....	49
Figura 3.11 Máximo retardo en la simulación sin EDCA y con EDCA.....	50
Figura 3.12 Porcentaje de pérdidas en la simulación sin EDCA y con EDCA.....	50
Figura 4.1 Diagrama en bloques de la infraestructura para experimentación .....	53
Figura 4.2 Lista de valores de parámetros EDCA tanto en AP como en STA.....	55
Figura 4.3 Técnica para medir tiempo entre cambio de valores de parámetros.....	57
Figura 4.4 Infraestructura para un flujo de VoIP + 5 de Datos .....	61
Figura 4.5 Resultado de un solo flujo de VoIP .....	62
Figura 4.6 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 3 .....	63
Figura 4.7 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 4 .....	63
Figura 4.8 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 5 .....	63
Figura 4.9 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 6 .....	64
Figura 4.10 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 7 .....	64
Figura 4.11 Retardo en ms del flujo de VoIP ante la variación del AIFS[AC_BE] .....	65
Figura 4.12 Pérdida en % del flujo de VoIP ante la variación del AIFS[AC_BE] .....	65
Figura 4.13 <i>Jitter</i> en ms del flujo de VoIP ante la variación del AIFS[AC_BE] .....	65
Figura 4.14 Infraestructura para 5 flujos de VoIP + 1 de Datos.....	67

Figura 4.15 Resultado de cinco flujos de VoIP con $CW_{max} = 4 + 1$ de Datos.....	69
Figura 4.16 Resultado de cinco flujos de VoIP con $CW_{max} = 15 + 1$ de Datos.....	69
Figura 4.17 Resultado de cinco flujos de VoIP con $CW_{max} = 31 + 1$ de Datos.....	70
Figura 4.18 Topología de simulación para la variación del $CW_{max}[AC\_VO]$ .....	72
Figura 4.19 Archivo de registro de colisiones para $CW_{max}[AC\_VO] = 4$ .....	74
Figura 4.20 Archivo de registro de colisiones para $CW_{max}[AC\_VO] = 15$ .....	74
Figura 4.21 Archivo de registro de colisiones para $CW_{max}[AC\_VO] = 31$ .....	75
Figura 4.22 Colisiones de flujos de VoIP ante la variación del $CW_{max}[AC\_VO]$ .....	75
Figura 4.23 Infraestructura para únicamente flujos de Datos.....	77
Figura 4.24 Throughput promedio al variar el $AIFS[AC\_BE]$ sin tráfico de VoIP .....	79

# 1.Introducción

## 1.1. Antecedentes

Las redes inalámbricas IEEE 802.11, conocidas también con el nombre comercial de Wi-Fi, se han convertido en las redes de datos inalámbricas con mayor tasa de penetración a todos los niveles (Agredo y López, 2004): público, empresarial, SoHo y en el hogar; inclusive y como aspecto relevante, esta tecnología tiene la capacidad de permitir la comunicación de datos y voz en zonas rurales (Hurtado, *et al*, 2008) y en *hotspots* en zonas urbanas (Gomez, *et al*, 2008), de tal forma que las posibilidades tecnológicas de poder llevar VoIP sobre redes inalámbricas IEEE 802.11 (WiFi<sup>1</sup>) conocida por diversos nombres como VoWLAN<sup>2</sup>, VoWiFi<sup>3</sup> y VoFi<sup>4</sup>, han despertado un creciente interés tecnológico y comercial; es así, como en la actualidad se cuenta con teléfonos WiFi de empresas como Belkin, SMC, Linksys, Netgear y Dlink y teléfonos duales GSM/WLAN de empresas como Motorola, Nokia, BenQ-Siemens, Alcatel y Sony-Ericsson, entre otras.

Sin embargo, todos estos teléfonos VoIP para el soporte a funciones de calidad de servicio (QoS, *Quality of Service*) a nivel de enlace trabajan con la especificación de multimedia WiFi (WMM, *WiFi Multi-Media*) la cual toma solamente el mecanismo de Acceso Distribuido al Canal Mejorado (EDCA, *Enhanced Distributed Channel Access*) (Alianza WiFi WMM, 2004) del estándar IEEE 802.11e el cual está derivado del principio de funcionamiento de *Diffserv* (Moon y Aghvami, 2003) en el que cada tipo de tráfico se identifica por un descriptor en la trama y de acuerdo a él recibe un tratamiento diferente en su posibilidad de obtener acceso al medio estableciendo prioridad (Sinreich, *et al*. 2006); el otro mecanismo de IEEE 802.11e, el Acceso al Canal Controlado por la HCF<sup>5</sup> (HCCA, *HCF Controlled Channel Access*) derivado del principio de funcionamiento de *IntServ* (Camp, 2002) requiere de capacidad de negociación dado que el cliente le informa al Punto de Acceso (AP, *Access Point*) sus necesidades de ancho de banda y límites de retardo entre otros y el AP debe evaluar si está en capacidad de ofrecérselos al cliente (IEEE 802.11e, 2005), por esto el tráfico puede ser aceptado o rechazado; una característica clave de este segundo método es la necesidad de un control centralizado por lo que solo puede operar en modo infraestructura donde el AP ejerza dicho rol.

Por lo anterior, EDCA facilita la implementación práctica y es más escalable, HCCA implica la necesidad de una mayor capacidad de procesamiento y memoria en el AP lo que desfavorece las intenciones de masificación de la tecnología por los costos económicos implicados (Martell, 2007), por ende, se aprecia que lo que está

---

<sup>1</sup> *Wireless Fidelity* (Fidelidad Inalámbrica) – Certificación de Interoperabilidad de la Alianza WiFi

<sup>2</sup> *Voip over WLAN* (Voz sobre IP sobre redes inalámbricas de área local)

<sup>3</sup> *Voip over WiFi* (Voz sobre IP sobre redes WiFi)

<sup>4</sup> Abreviatura de VoWiFi

<sup>5</sup> *Hybrid Coordination Function* – Función de Coordinación Híbrida

sucediendo es algo similar a lo ocurrido con el estándar original 802.11 en el que también se tenían dos mecanismos de acceso: uno distribuido entre las estaciones y el otro controlado por el AP, de los cuales en la práctica el segundo nunca tuvo la aceptación ni masificación esperada, es decir, la generalidad de los dispositivos comerciales solamente implementaron el soporte para obtener acceso al medio con un esquema que no dependía de tareas de procesamiento del AP sino de mecanismos para evitar la colisión entre los clientes que quisieran acceder al medio. Por tanto, en esta Tesis de Maestría con el interés práctico de realizar procesos de investigación para generar aportes al conocimiento científico global se trabajó, con un enfoque experimental, sobre EDCA que la Alianza WiFi lo ha denominado WMM<sup>6</sup> y que desde mayo de 2006 es el único esquema de calidad de servicio que es certificado (Epshtein, 2006) por la misma.

## 1.2. Justificación

Como se explica en (Mangold, *et al*, 2003), de acuerdo con el estándar IEEE 802.11e el AP utiliza los valores por defecto de EDCA cuando se inicia una red inalámbrica y puede modificarlos posteriormente de forma dinámica dadas las condiciones de tráfico. Sin embargo, ni en dicho estándar ni en la especificación WMM se establece la forma en que deben modificarse estos valores, siendo éste el vacío que se busca comenzar a llenar con el desarrollo del presente proyecto.

En este sentido aunque en este proyecto primero se busca evidenciar o validar por medio de experimentación<sup>7</sup> el impacto sobre el desempeño de VoIP y Datos que tiene una red inalámbrica cuando se hace o no uso de EDCA, luego se centra en determinar con un procedimiento basado en ensayo y error, los valores de dichos parámetros que pueden contribuir a satisfacer los requisitos de VoIP cuando el nivel de tráfico de un QBSS sea alto (un escenario en cuanto a Datos y otro en cuanto a VoIP) y valores que puedan facilitar el logro de *throughputs* más altos en el tráfico de mejor esfuerzo cuando el nivel de tráfico de VoIP sea nulo<sup>8</sup>.

Así, se trató de estimar el impacto de la variación de los parámetros, para establecer si era justificable modificar los definidos por defecto en el estándar, buscando de esta forma un posible proyecto de Doctorado que sería la generación de un algoritmo que realice este proceso dinámicamente.

Del estudio detallado del estándar para encontrar la intencionalidad de cada uno de los parámetros se ha hecho una selección para evaluar su efecto dependiendo del escenario y dado que este no es un estudio multivariable, se ha tomado el AIFS y el par CWmin y CWmax de manera aislada como los parámetros con los cuales trabajar. El parámetro TXOP no se trabajó porque los valores por defecto aplican para las

---

<sup>6</sup> En lo sucesivo se trabajará de forma indistinta los términos WMM y EDCA

<sup>7</sup> Y cuando fue necesario o conveniente por simulación

<sup>8</sup> Inicialmente se planteó que fuera "bajo" pero se tenía la paradoja de la necesidad de proteger el desempeño de VoIP sobre el logro de *throughputs* más altos en el tráfico de *best-effort*, por lo que se optó por hacer el tráfico de VoIP nulo.

categorías de *Best Effort* y *Background*, y el interés principal de este proyecto era buscar valores para optimizar el desempeño de la categoría de Voz, dado que se enmarcó en un QBSS de VoWlan.

Adicionalmente, es muy importante indicar que se ha buscado hacer experimentación<sup>9</sup>, porque se consideró que ya se ha recorrido un importante camino en cuanto a simulación y que era necesario pasar a la parte experimental que es donde se puede tener un ambiente con todos los factores que afectan a una comunicación inalámbrica, encontrándose en concordancia con el siguiente fragmento tomado de IEEE Network de Julio de 2005, publicación que se enfocó en la temática de QoS en WLAN y que se traduce a continuación:

*“El desempeño real de los protocolos complejos de QoS es crucial para la planeación de redes. La naturaleza aleatoria de los protocolos MAC, su naturaleza distribuida, y el hecho de que son complicados por diseño, dado que involucran la selección de diversos parámetros, agregado a las condiciones fluctuantes del canal y las múltiples tasas de transmisión existentes, vuelven un desafío a la evaluación de desempeño. Sin embargo, se han desarrollado algunos modelos analíticos. Estos modelos producen la máxima capacidad y retardo tanto en redes 802.11 saturadas como no saturadas. Las simulaciones también son útiles en la evaluación de estos protocolos de red, **pero las pruebas experimentales reales son la única forma para poder estudiar el desempeño integrado en las situaciones de la vida real.**”* (Christos y Vergados, 2005)

Sin embargo, cuando fue necesario o conveniente, como herramienta de simulación se hizo uso del software NCTUns 4.0 (NCTUns, 2008) trabajando con el estándar IEEE 802.11b con el anexo IEEE 802.11e dado que este simulador soporta estas tecnologías y porque la velocidad nominal de la misma facilita al desarrollo experimental alcanzar una intensidad de tráfico suficiente para poder encontrar diferencias. Además, los resultados obtenidos pueden proyectarse a estándares tales como 802.11g y 802.11a.

### 1.3. Metodología

Para el desarrollo general del proyecto se trabajó en fases siguiendo el modelo lineal secuencial, de tal forma que inicialmente se realizó un estudio para apropiar los fundamentos tecnológicos y afianzar la base teórica que daría sustento a la formulación de hipótesis como se verá en el capítulo 2.

El trabajo realizado se enmarcó en un proceso Experimental tal como aparece justificado en la sección anterior, y pese a los inconvenientes presentados en especial para lo tratado en cuanto a la modificación de valores de parámetros EDCA, como se

---

<sup>9</sup> Que valga la oportunidad para indicarlo fue más complejo que la misma simulación porque se requirió de infraestructura y de un proceso arduo tanto para poner en funcionamiento adecuado los equipos como para obtener los resultados.

discutirá más adelante, con dedicación y recursividad<sup>10</sup> fue finalmente posible adecuar una sala<sup>11</sup> para generar un laboratorio en el que se trabajó dentro de un ambiente de software libre (Linux y Drivers GPL) utilizando equipos PC con tarjetas de chipset Atheros que funcionan con el *driver* MadWifi (MadWifi, 2008).

En casos considerados convenientes o necesarios se recurrió a la Simulación, para lo que con fines de referencia se buscó una metodología que sirviera al proceso y se encontró interesante y apropiada la que aparece explicada en el Apéndice H. Esta metodología es de la Universidad de Michigan-Dearborn, la cual, tal como se muestra en (Ulgen, *et al*, 2004), abarca aspectos de otras clases de metodologías todas estas dirigidas hacia una perspectiva práctica.

## 1.4. Alcances y Limitaciones

El presente proyecto trabajó sobre la base del estándar IEEE 802.11b con el anexo IEEE 802.11e por las siguientes razones:

1. La velocidad de esta opción de WLAN facilita la saturación con fines de evaluación.
2. Es la velocidad soportada por el simulador utilizado y por tanto podía darse un proceso de validación entre los resultados de simulación y experimentación.
3. De acuerdo con (Dangerfield, *et al*, 2006) utilizando 802.11b con solo cinco estaciones de datos compitiendo por el medio se induce una pérdida del 10% de paquetes lo que degrada notoriamente la calidad de una comunicación de VoIP y produce inclusive su caída

En especial la tercera razón permitió considerar trabajar con experimentación en el sentido que era posible comprobar el impacto de utilizar EDCA siempre y cuando se contara con al menos 6 computadores que tuvieran que competir por el medio, uno<sup>12</sup> de voz y los otros cinco de datos, que aunque dispendioso, podía buscarse establecer la infraestructura, lo que finalmente fue logrado así hubiese sido necesario el aporte de dispositivos y elementos propios.

Para la parte experimental que era donde se hacía necesario contar con la infraestructura que permitiera llevar a cabo la investigación se trabajó con siete computadores con sistema operativo Linux con Ubuntu versión 8.04, uno configurado como punto de acceso y los otros como estaciones asociadas a dicho punto de acceso. En esta parte de experimentación en laboratorio tuvo que lidiarse con diferentes circunstancias, tales como: infraestructura compartida, horarios de utilización y reinstalaciones y reconfiguraciones. Sin embargo, las circunstancias fueron

---

<sup>10</sup> Tal como se verá especialmente en el Capítulo 4

<sup>11</sup> Sala de Posgrados del IPET

<sup>12</sup> Para permitir que los otros cinco fueran de datos y de esta forma producir problemas al flujo de VoIP

controladas y los problemas técnicos resueltos con paciencia y dedicación, inclusive en los capítulos 3 y 4 que tratan en específico los resultados de investigación, se ha ubicado primero la experimentación, pues, permite contar con un escenario con equipos e infraestructura hardware y software existente y vigente en el momento.

## **1.5. Contribución**

La Tabla 1.1 permite establecer el estado actual de conocimiento y/o aporte sobre el problema de investigación de este proyecto.

REFERENCIA	ANALISIS/RESUMEN	AREA DE APORTE
(Mangold, et al, 2003)	Se realiza la presentación de los mecanismos de 802.11e para proporcionar QoS a las WLAN. Indica que el QAP establece los valores por defecto de los parámetros de EDCA cuando se inicia el QBSS y que <b>los puede cambiar dinámicamente ante variaciones en el medio y/o en las condiciones de tráfico</b> , sin embargo, no presenta como deben hacerlo.	Justificación de la Realización del Proyecto
(Garg, et al, 2003)	Se realiza la evaluación de los mecanismos de QoS del Draft 2 de 802.11e por medio de NS-2.1b6, es interesante porque establece que EDCF (EDCA) proporciona mejoras significativas para el tráfico de alta prioridad, pero a costa del desempeño del de menor prioridad. <b>Plantea la necesidad de la selección de parámetros de EDCA</b> para alcanzar los objetivos de desempeño deseados, finalmente valora su simplicidad y naturaleza descentralizada.	Justificación de la Realización del Proyecto
(Choi, et al, 2003)	Introduce a EDCF (EDCA) del Draft 4 de 802.11e enfocándose en la presentación del mecanismo en si para la mejora del desempeño de la red, pero no trata la selección de valores para los parámetros EDCA, sin embargo, <b>establece la necesidad de poder hacerlo dependiendo de la carga de la red</b> y las aplicaciones que se estén soportando.	Justificación de la Realización del Proyecto
(Kong, et al, 2004)	Se presenta un modelo analítico para evaluar el desempeño de EDCA, por lo que establece la necesidad de limitar el tráfico de alta prioridad para lo que propone un sistema de control de admisión de las llamadas.	Modelo Analítico
(Tao, et al, 2006)	El objetivo era evaluar si WMM/EDCA proporciona un mecanismo suficiente para garantizar la entrega de video para las redes en el hogar en un escenario de Triple Play, realizando simulación con NS-2.26 con el parche de la Universidad Técnica de Berlín y utilizando los parámetros por defecto de EDCA, se concluye que aunque aumenta la probabilidad de tomar el medio, no da garantía de hacerlo porque no se pueden satisfacer los requerimientos en una red altamente competitiva.	Modelo Analítico
(Gao, et al, 2005)	Las Universidades de Hong Kong y la Tecnológica Nanyang han realizado un examen intensivo a los avances en los algoritmos y protocolos para el control de admisión de IEEE 802.11e, sin embargo, las referencias muestran que trabajaron con el Draft 6 de Noviembre de 2003. En este se demuestra que EDCA funciona mejor con bajo tráfico en el QBSS.	Especificación de EDCA y evaluación por simulación.
(Bianchi, et al, 2005)	las Universidades de Roma y Palermo en el marco del desarrollo del proyecto PRIN-TWELVE aportan una explicación detallada de la operación de los mecanismos de EDCA, con un estudio analítico centrado en las distribuciones de probabilidad de acceso, para de esta forma demostrar que el mecanismo de espacio entre trama arbitrado ofrece un mejor desempeño que el de ventana variable, y además que el nuevo sistema presenta alta compatibilidad con los heredados que no tiene QoS, trabajan solamente el esquema EDCA y lo hacen con el Draft 10 de Septiembre de 2004	Evaluación de EDCA por Simulación.
(Guchhait, et al, 2006)	Por medio de Simulación con OPNET Modeler utilizando el módulo de Comunicaciones Inalámbricas se desarrollaron modelos de simulación para determinar la capacidad de tres escenarios de simulación: Voz, Voz y Video y Voz y Video con tráfico de Mejor Esfuerzo. Se demuestra la importancia de contar con un sistema de control de admisión y como con éste se puede dar garantía de servicio para los flujos existentes.	Evaluación de EDCA por Simulación.

(Lera, et al, 2005)	En la Universidad Mediterránea de Calabria se tiene el trabajo con la versión más nueva del Draft (V.13) <sup>13</sup> , siendo soportado sobre la base del simulador NS-2 con los aportes realizados por el Grupo de Redes de Telecomunicaciones de la Universidad Técnica de Berlín y con enfoque hacia EDCA en las redes Ad Hoc dejando sin tratamiento la topología de infraestructura de los entornos empresariales y del hogar	Evaluación de EDCA por Simulación.
(Branchs, et al, 2005)	En la Universidad Carlos III de Madrid, se realizó un estudio experimental para evaluar el mecanismo EDCA de 802.11e, se trabajó con el Draft 13 de la norma, el enfoque estuvo en la evaluación más no en la selección de parámetros de EDCA, pero <b>se concluye la necesidad</b> buscar la forma de poder ofrecer algo de garantía de servicio analizando condiciones de alta y baja carga en la red y de algoritmos para <b>determinar la configuración óptima de EDCA.</b>	Evaluación de EDCA por Experimentación.  Justificación de la Realización del Proyecto
(Dangerfield, et al, 2006)	Se realiza la evaluación experimental de EDCA con 802.11b con el objetivo de comprobar su efectividad para el desempeño de VoWLAN, se enfoca en el aumento en los valores del AIFS del tráfico de baja prioridad para satisfacer los requerimientos de QoS de la Voz.	Evaluación de EDCA por Experimentación.
(Xia y Li, 2004)	Propone un esquema de control de admisión basado en realimentación para ajustar los parámetros del tráfico de mejor esfuerzo mientras se mantienen los requisitos del tráfico de tiempo real. Por medio de simulación se argumenta que ajustando únicamente la Ventana de Contienda Mínima (CWmin) es suficiente y que el ajuste de la Ventana de Contienda Máxima y del Espacio Inter-Trama de Arbitramento es una opción demasiado agresiva y difícil de controlar	Impacto de los Parámetros EDCA
(Pattara, et al, 2004)	Se trabaja con la adaptación dinámica de los parámetros MAC para dar un <i>throughput</i> garantizado, la idea es emplear otro algoritmo para el cálculo de la Ventana de Contienda y volverla determinística.	Procedimiento para la Selección de Valores
(Zhang y Zeadally, 2004)	Se monitorea el retardo y la tasa de pérdidas del tráfico de tiempo real y luego se usa esto para controlar los parámetros de acceso al canal.	Procedimiento para la Selección de Valores

Tabla 1.1. Clasificación y análisis de las principales referencias

En estos últimos trabajos se utiliza un enfoque basado en la realimentación, se monitorea el desempeño en tiempo de ejecución y luego se tratan de ajustar los parámetros de acuerdo a lo observado, adicionalmente, la evaluación de resultados únicamente se basa en simulación. Estos mecanismos implican modificaciones en los APs y en los clientes, sin embargo, la investigación de la que trata este proyecto, a diferencia de investigaciones previas, busca proporcionar al administrador de una infraestructura de red inalámbrica la posibilidad de utilizar los APs que soportan la especificación WMM tal como se encuentra actualmente en el mercado y dependiendo del tráfico esperado tener un referente para configurar los valores de parámetros y propender por el cumplimiento de los requisitos de VoIP cuando sea alto el tráfico de datos o de voz y para aumentar el *throughput* del tráfico de mejor esfuerzo cuando no exista tráfico de VoIP, con un enfoque centrado en la experimentación hasta donde sea posible.

<sup>13</sup> Precisamente el que en Julio de 2005 fue aprobado para publicación como estándar

## 1.6. Organización

Este documento se encuentra elaborado y organizado atendiendo la guía que para tal fin fue establecida por la coordinación del programa de maestría (Jojoa, 2007) y como tal en el segundo<sup>14</sup> capítulo se presentan de forma clara y concisa los fundamentos teóricos del proyecto en específico orientados a IEEE 802.11 y a IEEE 802.11e con un enfoque hacia el funcionamiento de EDCA y los parámetros de la misma. En el tercer capítulo se detalla el proceso de evaluación de EDCA presentando el procedimiento generado para la obtención de los resultados. En el cuarto se presenta el informe del procedimiento de variación de valores de parámetros con el fin de seleccionar los que ayudaron a mejorar el desempeño de acuerdo a los escenarios que fueron enunciados en la Justificación<sup>15</sup>. Luego se tienen las conclusiones y recomendaciones.

El lector debe tener en cuenta que en este documento “*los temas desarrollados por el autor y que complementan el texto principal*” se presentan en los apéndices, siendo por tanto elementos de importancia dentro del proceso, que permitieron cumplir con el límite de extensión que se espera para la parte específica que atañe al documento principal, pero evidenciando el camino recorrido por el autor para el logro de los objetivos de la Tesis, con elementos tales como: procesos previos de configuración y experimentación, manejo del simulador, resultados de síntesis, etc.

De esta manera en este documento se ha buscado atender la guía mencionada, entendiendo y compartiendo el espíritu de la misma.

---

<sup>14</sup> De acuerdo con la guía, la Introducción es el primer capítulo del documento.

<sup>15</sup> Sección 1.2

## 2. Funcionamiento de EDCA

Antes de proceder con la explicación de EDCA a nivel MAC del estándar 802.11e es necesario conocer las características de su predecesor, su arquitectura, métodos de acceso, funcionamiento y limitaciones con el propósito de comprender de una mejor manera los cambios realizados al estándar 802.11 con la generación de la enmienda 802.11e para generar este marco teórico se han tomado referencias estándares de (IEEE 802.11, 1999) y (Alianza WiFi WMM, 2005) y referencias técnicas de (Gast, 2005), (Prasad, 2005) y (Coleman, 2006).

### 2.1. Estándar IEEE 802.11 Nivel MAC

El estándar IEEE 802.11 en su arquitectura básica a nivel MAC, define dos métodos de acceso al medio, como se pueden apreciar en la Figura 2.1 (IEEE 802.11, 1999), los cuales son: Función de Coordinación Distribuida (DCF, *Distributed Coordination Function*) y Función de Coordinación Puntual (PCF, *Point Coordination Function*). El primero provee un servicio *best-effort* o de mejor esfuerzo cuya implementación es obligatoria, mientras que PCF es opcional y provee servicios que exigen límites de tiempo.



Figura 2.1 Arquitectura MAC estándar 802.11

Para estos métodos de acceso existen cuatro periodos de tiempo que se utilizan para controlar el tiempo de espera antes de acceder al medio. Éstos son:

- Espacio Inter-Trama Corto (SIFS, *Short Interframe Space*): Tiempo de espera más corto y por lo tanto la prioridad más alta para acceder al medio.

Es utilizado en mensajes de control de pequeña longitud como tramas CTS, ACK y respuestas a mensajes de *polling*.

- Espacio Inter-Trama PCF (PIFS, *PCF Interframe Space*): Tiempo de espera más largo que SIFS pero más pequeño que el DIFS. Sólo es utilizado por las estaciones que operan bajo PCF.
- Espacio Inter-Trama DCF (DIFS, *DCF Interframe Space*): Tiempo de espera más largo que PIFS y con una menor prioridad de acceso al medio. Utilizado por estaciones que operan bajo DCF y transmiten tramas de datos o de gestión.
- Espacio Inter-Trama Extendido (EIFS, *Extended Interframe Space*): Tiempo de espera con mayor duración utilizado por las estaciones que operan bajo DCF y es empleado cuando ocurre una falla.

### 2.1.1. Las funciones de coordinación de 802.11: DCF y PCF

DCF opera sobre el Período de Contienda y se basa en el método CSMA/CA, cuyo funcionamiento, de forma general, se resume de la siguiente manera: una estación está en la capacidad de transmitir si ésta ha detectado que el medio está libre por al menos la duración de un DIFS. De lo contrario, la estación retrasa la transferencia de datos hasta que la transmisión que se encuentra en curso finalice. Ver Figura 2.2 (Gast, 2005).

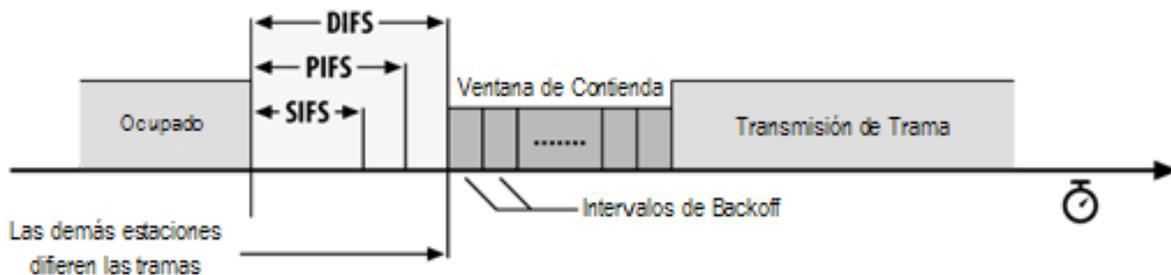


Figura 2.2 Funcionamiento de DCF

El AP transmite la duración de la transmisión del paquete que llega al nivel MAC en el encabezado de la trama MAC sobre el Campo Duración (*Duration Field*) con el fin de actualizar el contador de cada estación llamado Vector de Asignación de Red (NAV, *Network Allocation Vector*) con tal duración expresada en microsegundos.

Una vez el NAV ha llegado a cero, la estación asume que la transmisión en curso ha finalizado y genera un tiempo de *backoff*<sup>16</sup> de la siguiente manera:

$$\text{Tiempo de backoff} = T_{B.O.} = \text{Random} \times \overline{\text{SlotTime}} \quad (2.1)$$

<sup>16</sup> Período de tiempo aleatorio generado por la estación después de verificar que el medio está libre.

En donde  $Random()$  es un entero obtenido a partir de una función distribuida uniformemente<sup>17</sup> sobre el intervalo  $[0, CW]$  y  $CW$  hace referencia a la Ventana de Contienda ( $CW$ , *Contention Window*) que es un entero ubicado entre  $CW_{min}$  y  $CW_{max}$ , dependiendo del estándar se tienen valores de los elementos mencionados como aparece condensando en la Tabla 2.1 (IEEE 802.11a) (IEEE 802.11b) (IEEE 802.11g). El término *SlotTime* está relacionado con la duración de la ranura de tiempo y de la misma manera que el SIFS, su valor se obtiene a partir de: la sumatoria de los retardos causados por el procesamiento a nivel MAC, la detección de la portadora, tiempo de transmisión del preámbulo (PLCP) y tiempo de propagación de la onda, características propias de la tecnología de nivel físico.

Estándar	SIFS(μs)	SlotTime(μs)	PIFS(μs)	DIFS(μs)	$CW_{min}$	$CW_{max}$
802.11a	16	9	25	34	15	1023
802.11b	10	20	30	50	31	1023
802.11g <sup>18</sup>	10	9	19	28	15	1023
		20	30	50	31	1023

Tabla 2.1. Parámetros MAC para 802.11a, b, g

Una estación que hace uso del procedimiento de *backoff* utiliza el mecanismo de detección de portadora (nivel PHY) para determinar si el medio está libre en cada ranura de tiempo. El procedimiento de *backoff* decrementa su temporizador en una ranura de tiempo cada vez que detecta que el medio está libre por un período igual a la duración de una ranura de tiempo; si la estación determina que el medio está ocupado, el procedimiento de *backoff* se suspende, es decir, no decrementa esa ranura de tiempo. Para proceder, el medio deber estar libre un DIFS y la estación comienza la transmisión una vez el temporizador de *backoff* haya expirado (llega al valor cero).

Una colisión se origina si dos o más estaciones transmiten al mismo tiempo, en este caso la ventana de contienda se duplica y se da inicio a un nuevo procedimiento de *backoff*. Como se mencionó anteriormente, la ventana de contienda se encuentra dentro de límites en donde ésta inicia con un valor mínimo ( $CW_{min}$ ) y se duplica hasta llegar a un valor máximo ( $CW_{max}$ ) el cual se mantiene hasta que éste sea re-establecido<sup>19</sup> ( $CW_{min}$ ) después de cada intento de transmisión exitoso. Este proceso continúa hasta que la transmisión termine con éxito o se descarte la trama.

Con el fin de proveer servicios con altas exigencias en tiempo tales como voz, audio, y video, en la especificación 802.11 original se definió la Función de Coordinación Puntual – PCF que requiere de un AP que controle el acceso al medio y sondee las estaciones (*polling*), por lo tanto sólo se puede utilizar en las

<sup>17</sup> Mediante esta función el estándar 802.11 procura brindar la misma probabilidad de transmisión a todas las estaciones que están compitiendo por el medio.

<sup>18</sup> Para este estándar la duración del *slot time* depende si se trabaja en una red completamente 802.11g (9 μs) o si existe interoperabilidad con equipos 802.11b (20 μs)

<sup>19</sup> En términos coloquiales: reseteado

configuraciones de red en modo infraestructura y opera en el Período Libre de Contienda. Este método de acceso utiliza un Punto Coordinador (PC, *Point Coordinator*) el cual funciona en el AP y se emplea para determinar qué estación tiene permiso para transmitir. De hecho, PCF es un mecanismo de acceso de *polling* en donde el PC tiene el papel de sondeo maestro (*polling master*) y mantiene una lista de las estaciones registradas a las cuales autoriza una por una para dar inicio a la transmisión de su información. Ninguna estación está en la capacidad de transmitir a menos que sea autorizada, y de la misma manera reciben información de un AP solamente cuando reciben autorización del punto coordinador.

### 2.1.2. Limitaciones de calidad de servicio en DCF y PCF

Como ya es conocido, las aplicaciones de datos, video y audio tienen requerimientos de transmisión diferentes. Sin embargo, con DCF todas las estaciones y flujos de datos tienen la misma prioridad de acceso al medio, es decir, no hay manera de favorecer el cumplimiento de los parámetros de calidad para cada servicio debido a la ausencia de un mecanismo que brinde prioridad en el acceso al medio según el tipo de paquete. En otras palabras, una estación no tiene la capacidad de diferenciar sus propios paquetes y por lo tanto carece de la capacidad de manejar calidades de servicio diferentes.

Aunque PCF se diseñó con el fin de proveer servicios con exigencias de tiempo, este método de acceso tiene ciertos problemas que llevan a un desempeño con una calidad de servicio pobre. Entre estos se encuentran (Intel, 2004):

- La incapacidad de las estaciones de comunicar sus requerimientos de QoS al Punto Coordinador hace difícil optimizar el mecanismo de *sondeo* en el PC.
- La provisión de QoS hace referencia a la transmisión de paquetes durante el Período Libre de Contienda (CFP, *Contention Free Period*), es decir, PCF existe en intervalo de tiempo limitado.
- Los retrasos impredecibles en la trama *Beacon*<sup>20</sup> conducen a un CFP más corto.
- PCF puede garantizar la transmisión de un paquete en cada período de supertrama (CFP+CP<sup>21</sup>). Sin embargo, si un servicio necesita de una transmisión de paquetes con mayor frecuencia, la red no está en la capacidad de soportarlo.

Teniendo en cuenta las anteriores limitaciones, el Grupo de Trabajo E de IEEE se encargó de mejorar el nivel MAC de 802.11 a través de una enmienda con el fin de soportar aplicaciones que tienen diferentes requerimientos de QoS. Como resultado de esto, se aprobó en el mes de Julio de 2005 el estándar 802.11e para

<sup>20</sup> Trama de gestión que contiene información relacionada con el SSID (*Service Set ID*), parámetros de configuración de Salto en Frecuencia (FH, *Frequency Hopping*), Secuencia Directa (DS, *Direct Sequence*), entre otros.

<sup>21</sup> Período de Contienda (CP, *Contention Period*).

su publicación final en noviembre del mismo año, cuya operación a nivel MAC se explicará a continuación.

## 2.2. Estándar IEEE 802.11e

Dado que el principal objetivo de éste estándar es la provisión de calidad de servicio en aplicaciones con determinadas exigencias, como es el caso de la voz para la cual las investigaciones que han permitido generar recomendaciones a la UIT-T (UIT-T G.114) han determinado, que retardos menores a los 150 ms pasan desapercibidos por el oído humano, sin embargo, si ese tiempo aumenta entonces la inteligibilidad de una conversación es deficiente, e inclusive, inaceptable; a continuación se explicarán un conjunto de conceptos que sirven de base para la comprensión de las funciones de coordinación propias del mismo.

De la misma manera que en el estándar 802.11 el BSS es el bloque constitutivo, en el estándar 802.11e ocurre igual con la diferencia de que en éste soporta calidad de servicio y se denota como Conjunto de Servicio Básico con Calidad de Servicio (QBSS, *Quality of Service Basic Service Set*). Además, las estaciones clientes que también soportan QoS se conocen como Estaciones con Calidad de Servicio (QSTA, *Quality of Service Stations*).

### 2.2.1. Provisión de QoS

El estándar 802.11e define dos tipos de soporte de QoS para el tráfico de un QBSS:

- **Calidad de servicio con prioridad:** Su principal tarea es la prioridad de una trama de datos con relación a otra. Se definen ocho Categorías de Tráfico (TC, *Traffic Categories*) las cuales son un conjunto de tramas de datos que se diferencian del resto mediante un Identificador de Categoría de Tráfico (TCID, *Traffic Category Identification*) de acuerdo a un mapeo predefinido de prioridades. Este valor varía desde 0 hasta 7.
- **Calidad de servicio clasificada:** En esta calidad de servicio, las tramas de datos con características de tráfico similares se agrupan en clases de tráfico llamadas Flujos de Tráfico (TS, *Traffic Streams*). La provisión de la calidad de servicio se realiza mediante la definición de las características de tráfico que se otorgan a dichas TS, y que son utilizadas en un QBSS, tales como: tamaño nominal de la Unidad de Datos de Servicio (MSDU, *MAC Service Data Unit*), tasa media de datos, límite de retardos, etc. Un flujo de tráfico es una trama de datos unidireccional que es reconocido por un Identificador de Flujo de Tráfico (TSID, *Traffic Stream Identifier*) el cual lo utilizan las entidades de nivel superior con el propósito de indicar al nivel MAC a que TS pertenece una trama de datos en particular y sus

correspondientes requerimientos de QoS. Este valor se encuentra en el rango de 8 a 15.

- *Generación de un flujo de tráfico*: Esta capacidad la establecen las capas de nivel superior tras la petición de una aplicación, la cual se entrega, posteriormente, a la Entidad de Gestión de Estaciones (SME, *Station Management Entity*) de la QSTA la cual hace parte del plano de gestión del estándar de 802.11. A continuación, la SME le permite al nivel MAC generar un nuevo flujo de tráfico con características específicas cuya definición se conoce como Especificación de Tráfico (*Traffic Specification*, TSPEC) y que son entregadas por la SME (EuroNGI, 2004).
- *Especificación de Tráfico de un TS*: Como se mencionó anteriormente una especificación de tráfico es la definición de las características de un flujo de tráfico (por ejemplo, los parámetros que definen el tipo de tráfico y que serán generados e intercambiados a través de un TS). Estos parámetros pueden ser de dos tipos: el primero describe las características generales del TS tales como direcciones fuente y destino, dirección del flujo, tipos de ACKs requeridos por las MSDUs (una nueva funcionalidad que tiene como opciones los mecanismos de reconocimiento: ACK Normal, No ACK y ACK de Bloque<sup>22</sup>) y el tipo de TS que describe su patrón de tráfico (periódico o no); el segundo grupo de parámetros está relacionado con las características de las tramas para un TS en particular y sus requerimientos para su transmisión.

### 2.2.2. Modificación de las funciones de acceso

El estándar 802.11e establece en su arquitectura que se presenta en la Figura 2.2 (IEEE 802.11e, 2005) una nueva función de acceso llamada Función de Coordinación Híbrida (*Hybrid Coordination Function*, HCF) la cual define dos funciones que se diferencian según el método de acceso que utilicen. En el método de Acceso al Canal Distribuido Mejorado (*Enhanced Distributed Channel Access*, EDCA) una de las funciones de 802.11e consiste en una modificación de la función DCF existente soportando calidad de servicio con prioridad, y en obtener acceso al canal basándose en la contienda por el medio. Por otra parte, la segunda función llamada Acceso al Canal Controlado HCF (*HCF Hybrid Controlled Channel Access*, HCCA) controla el acceso al canal con el fin de proveer una calidad de servicio clasificada. El término híbrido hace referencia a la capacidad de operar tanto en el Período de Contienda como en el Libre de Contienda.

---

<sup>22</sup> Block ACK

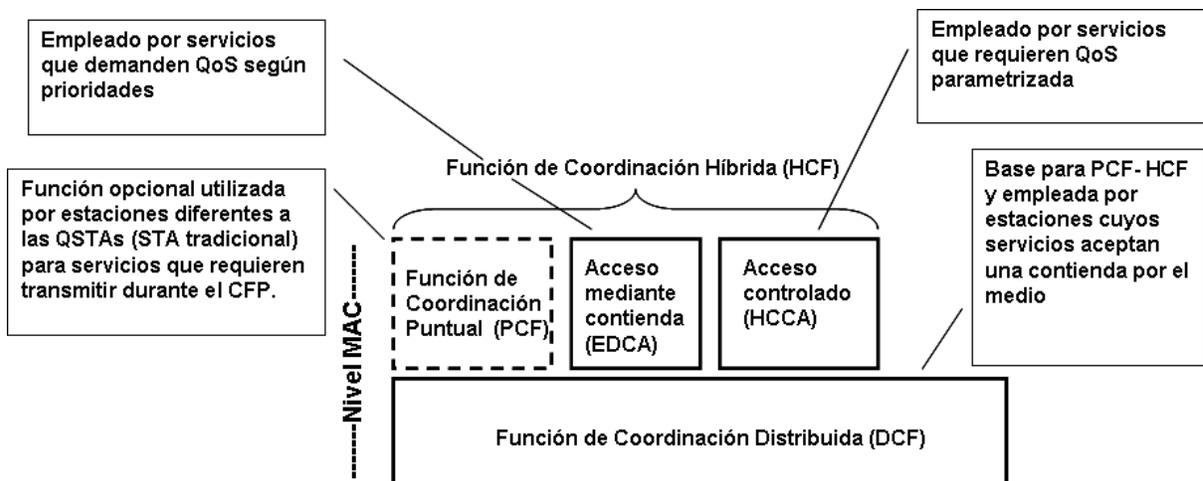


Figura 2.3 Arquitectura MAC estándar 802.11e

## 2.3. EDCA: Acceso al Canal Distribuido Mejorado

Ésta función es una mejora de las funcionalidades básicas de DCF soportando adicionalmente las categorías de tráfico de QoS con prioridad descritas previamente e intenta determinar una transmisión justa para cada trama. Para el soporte de EDCA, 802.11e define cuatro Categorías de Acceso (AC, *Access Categories*) y cada TC se mapea en una de estas AC según se muestra en la Tabla 2.2 (Alianza WiFi WMM, 2005).

Nivel de prioridad	Categoría de Tráfico (TC)	Categoría de Acceso (AC)	Tipo de Tráfico
Más baja ↓ Más alta	1	AC_BK	<i>Background</i>
	2	AC_BK	<i>Background</i>
	0	AC_BE	<i>Best Effort</i>
	3	AC_BE	<i>Best Effort</i>
	4	AC_VI	Video
	5	AC_VI	Video
	6	AC_VO	Voz
	7	AC_VO	Voz

Tabla 2.2. Mapeo de las categorías de tráfico sobre las categorías de acceso

Las categorías de tráfico heredan las denominadas prioridades de usuario del estándar IEEE 802.1Q e IEEE 802.1q de las redes Ethernet cableadas, la Figura 2.4 (Clark, 1999) muestra la trama 802.3 y la Figura 2.5 (IEEE 802.11e, 2005) la trama 802.11, ambas con soporte para QoS en nivel de enlace.

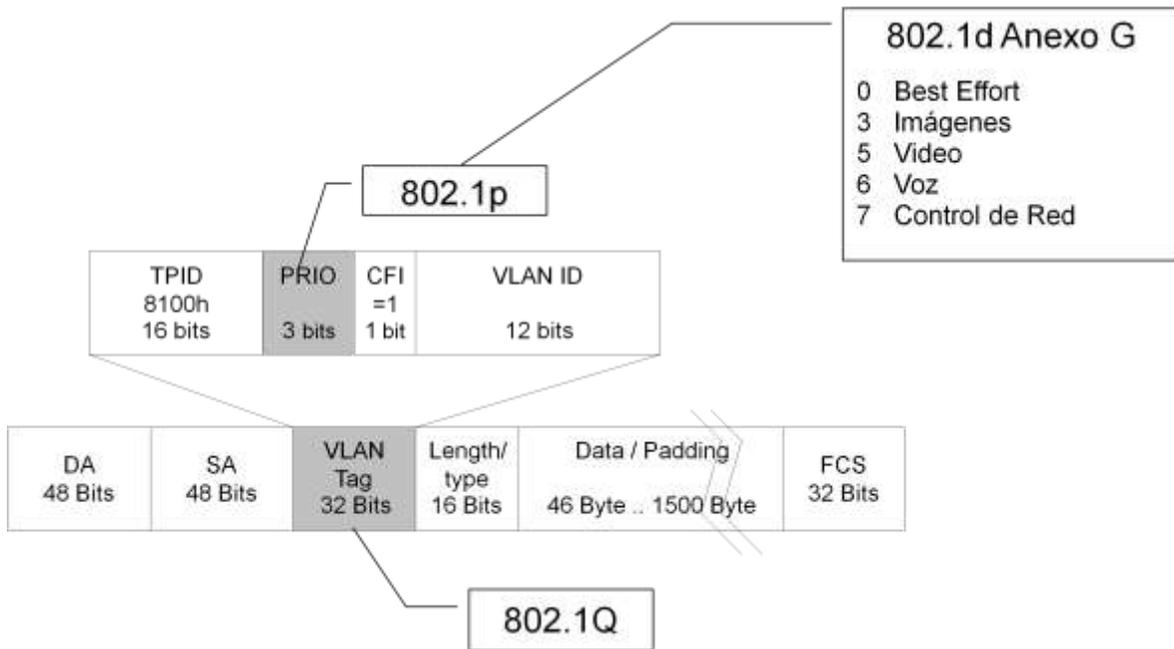


Figura 2.4 Trama IEEE 802.3 con soporte para QoS de tipo 802.1Q y 802.1p.

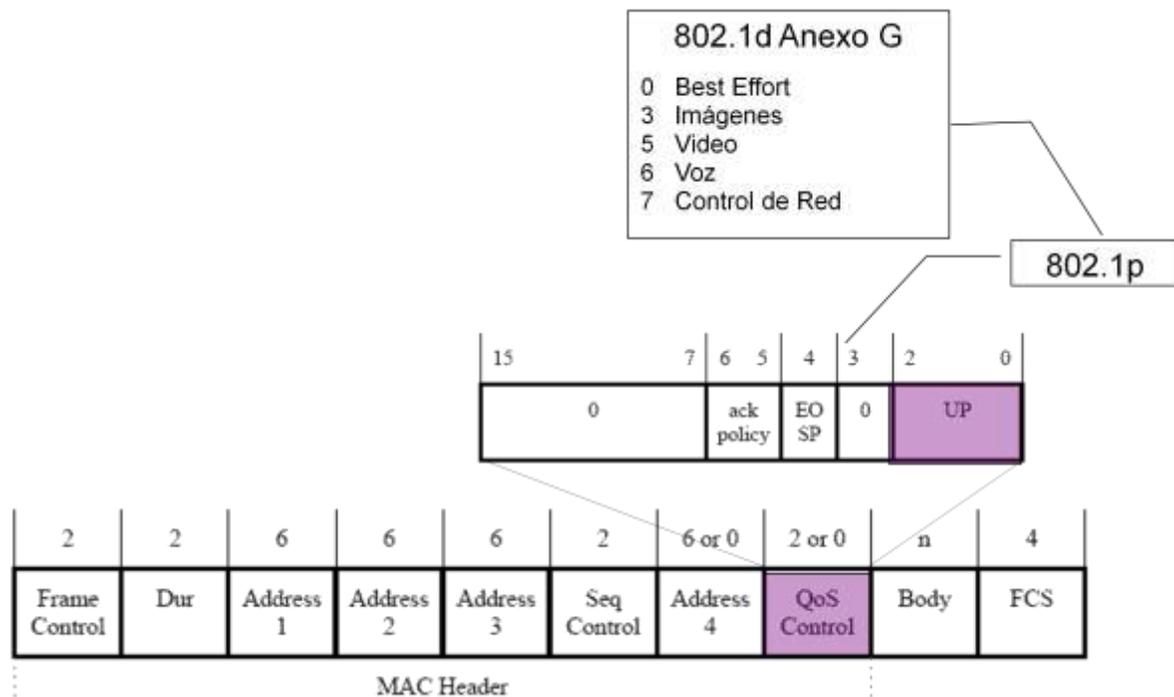


Figura 2.5 Trama IEEE 802.11 con soporte para QoS de tipo 802.11e.

Una comparación de las dos figuras anteriores muestra la intención de facilitar el mapeo de las prioridades de usuario de una red cableada Ethernet y las categorías de tráfico de una inalámbrica WiFi y viceversa.

Adicionalmente las aplicaciones extremo a extremo trabajan en el nivel de red y es en este escenario donde marcan los flujos de tráfico para que independiente de la tecnología de telecomunicaciones que se utilice conserven el tratamiento a lo largo de la ruta recorrida, haciendo uso en el momento de los Códigos para Servicios Diferenciados (DSCP, *Diferenciated Services Code Point*) o el Tipo de Servicio (ToS, *Type of Service*) de IP. Como ejemplos de esto, la Figura 2.6 muestra un Softphone bastante conocido en el mundo de VoIP denominado X-Lite específicamente en la sección donde puede configurarse el valor DSCP para flujos de VoIP, Video y Señalización; así mismo la Figura 2.7 muestra la capacidad que trae el sistema operativo Windows Vista™ para establecer políticas de QoS en las que se configura el valor DSCP para el tráfico generado por las aplicaciones.

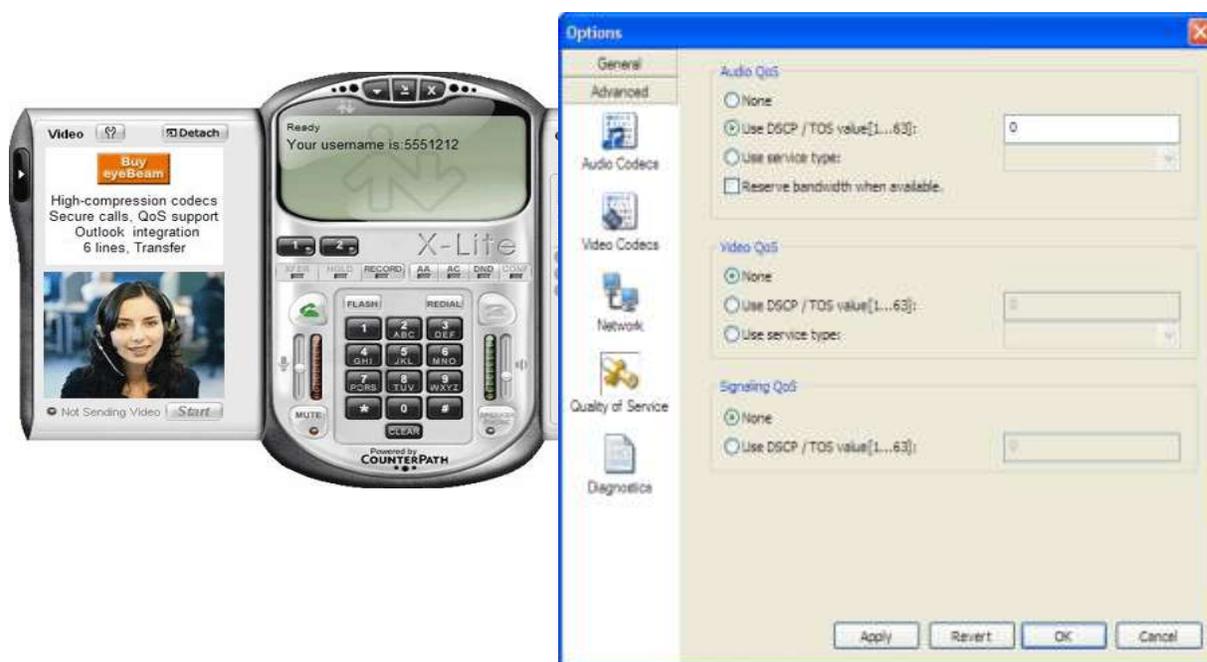


Figura 2.6 Ventana de configuración del DSCP en X-Lite.

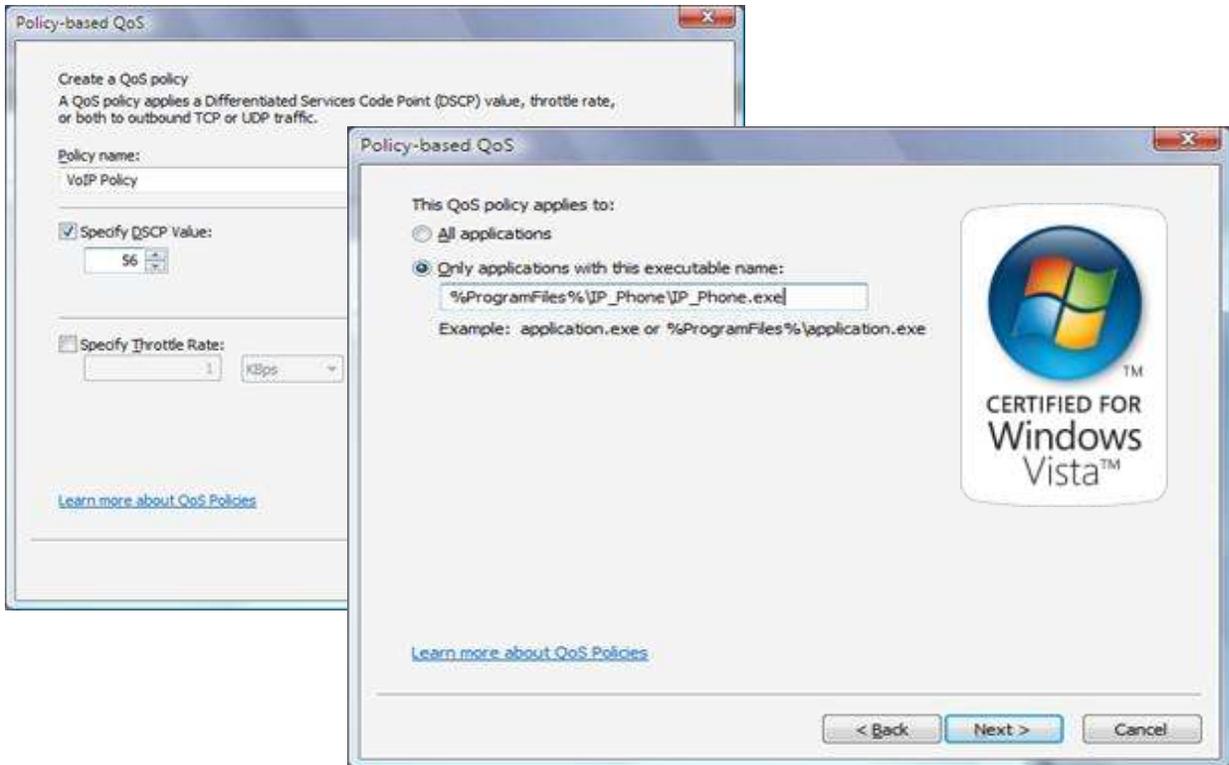


Figura 2.7 Ventana de creación de una política de QoS en Windows Vista™.

De esta manera la Tabla 2.3 (Avaya, 2005) (Alianza WiFi WMM, 2004) resume el mapeo que automáticamente deben hacer los controladores (*drivers*) en conjunción con el sistema operativo en un computador y los dispositivos tales como un teléfono de VoIP tanto cableado como inalámbrico para que se respeten las etiquetas y su connotación en el desempeño de los servicios.

Designación	Categoría de Acceso	Descripción	Etiqueta 802.1p	Prioridad de Usuario	DSCP
Voz	AC_VO	Comunicaciones de VoIP que requieren bajos retardos por lo que deben tener la más alta prioridad	7	Control de Red	56 - 63
			6	Voz	48 - 55
Video	AC_VI	Permite al Video tener una mayor prioridad sobre los datos pero menor sobre la voz.	5	Video	40 - 47
			4	Carga Controlada	32 - 39
Mejor Esfuerzo (Best Effort)	AC_BE	Para aplicaciones que son menos sensibles al retardo pero que si fuera alto sería bastante inaceptables como Navegar	3	Excelente Esfuerzo	24 - 31
			0	Mejor Esfuerzo	0 - 7
Background	AC_BK	La menor prioridad. Tolerar altos retardos, pero que posiblemente esperan buen throughput	2	Background	8 - 15
			1	Otros	16 - 23

Tabla 2.3. Mapeo entre categoría de acceso, prioridad de usuario y DSCP.

Así en las redes inalámbricas, EDCA es una mejora a las funcionalidades básicas del acceso al medio de 802.11 dado por DCF soportando las categorías de tráfico de QoS con prioridad, buscando de manera estadística facilitar el acceso al medio para los flujos más sensibles a retardos y pérdidas. Debe recordarse de la tabla 2.2 que en las redes inalámbricas las ocho categorías de tráfico se agrupan en solo cuatro categorías de acceso según se muestra en la Tabla 2.3.

Una estación (STA, *Station*) que opera con mecanismos de QoS como EDCA de 802.11e se denomina una QSTA, de igual forma, un AP con QoS se denomina QAP y a su vez un Conjunto Básico de Servicio BSS que trabaja con EDCA se denomina un QBSS. En una QSTA el funcionamiento opera de acuerdo a las mismas reglas generales definidas para DCF pero con la diferencia de que provee una cola (separada) para cada categoría de acceso, tal como se muestra en la Figura 2.8 (Avaya, 2005).

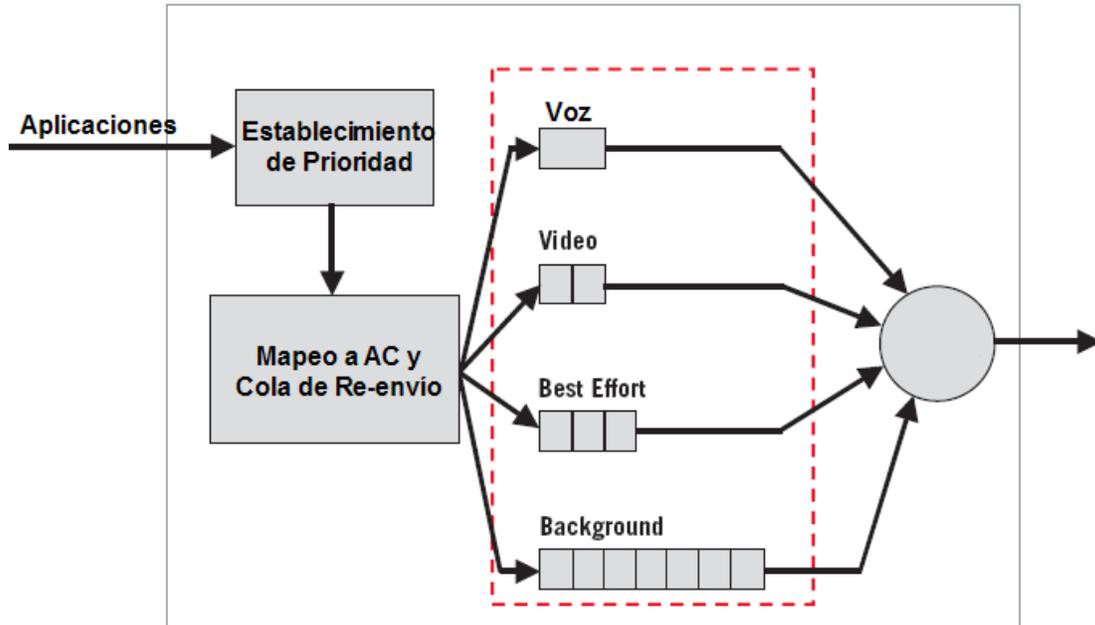


Figura 2.8 Lógica de encolamiento de las categorías de acceso de EDCA

Los cuatro parámetros de EDCA son: 1) Número de Espacios Inter-Trama de Arbitramento (AIFSN, *Arbitration Inter-Frame Space Number*), 2) Ventana de Contienda Mínima (CWmin, *Minimum Contention Window*) 3) Ventana de Contienda Máxima (CWmax, *Maximum Contention Window*) y 4) Oportunidad de Transmisión (TXOP, *Transmission Opportunity*). Estos parámetros se explican en la Tabla 2.4 (IEEE 802.11e, 2005) (Alianza WiFi WMM, 2005).

PARÁMETRO	EXPLICACIÓN
AIFSN	Especifica el número de intervalos de tiempo que un flujo de una categoría de acceso debe esperar antes de iniciar la contienda por el medio.
CWmin	Es el valor de entrada para el algoritmo que determina el tiempo inicial de espera aleatoria para intentar o re-intentar una transmisión. El valor proporcionado por este parámetro da el límite superior del rango entre el que debe tomarse dicho valor inicial. Si el primer valor expira antes que puedan ser enviados los datos, se incrementa el contador de re-intentos y se dobla este valor, este proceso continúa hasta cuando se alcanza el valor de la ventana de contienda máxima.
CWmax	Es el límite máximo para el crecimiento del valor de la espera aleatoria en milisegundos. El proceso de aumento del valor continúa hasta que el paquete se envía o hasta que se alcanza el valor de CWMax.
TXOP	Es una cantidad de tiempo expresada en microsegundos que una QSTA tiene para realizar transmisiones sin tener que competir con el medio.

Tabla 2.4. Explicación de parámetros EDCA.

Así por ejemplo, el tiempo de escucha de cada cola previo a la transmisión, es diferente y recibe el nombre de Espacio Inter-Trama de Arbitramento (AIFS, *Arbitration Inter-Frame Space*) dependiendo del AIFSN de la categoría correspondiente. Por ejemplo, si la categoría de acceso  $i$  tiene una prioridad menor que la categoría de acceso  $j$  entonces el AIFS de la primera es mayor que el de la segunda:  $(P(AC_i) < P(AC_j) \Rightarrow AIFS_i > AIFS_j)$ . Es decir que las Unidades

de Datos del Protocolo MAC (MPDUs, *MAC Protocolo Data Units*) de la categoría  $i$  deben esperar a que el medio esté libre por más tiempo que las de la categoría  $j$ . De esta manera EDCA provee prioridad de acceso al medio a las MPDUs con una prioridad más alta. El AIFS de cada categoría de acceso se calcula mediante la expresión (2.2) en la cual el valor del término AIFSN es un número entero que depende de la categoría de acceso.

$$AIFS_{AC_i} = AIFSN_{AC_i} \times SlotTime + SIFS \quad (2.2)$$

La Figura 2.9 (IEEE 802.11e, 2005) ilustra de manera efectiva lo mencionado en el párrafo anterior.

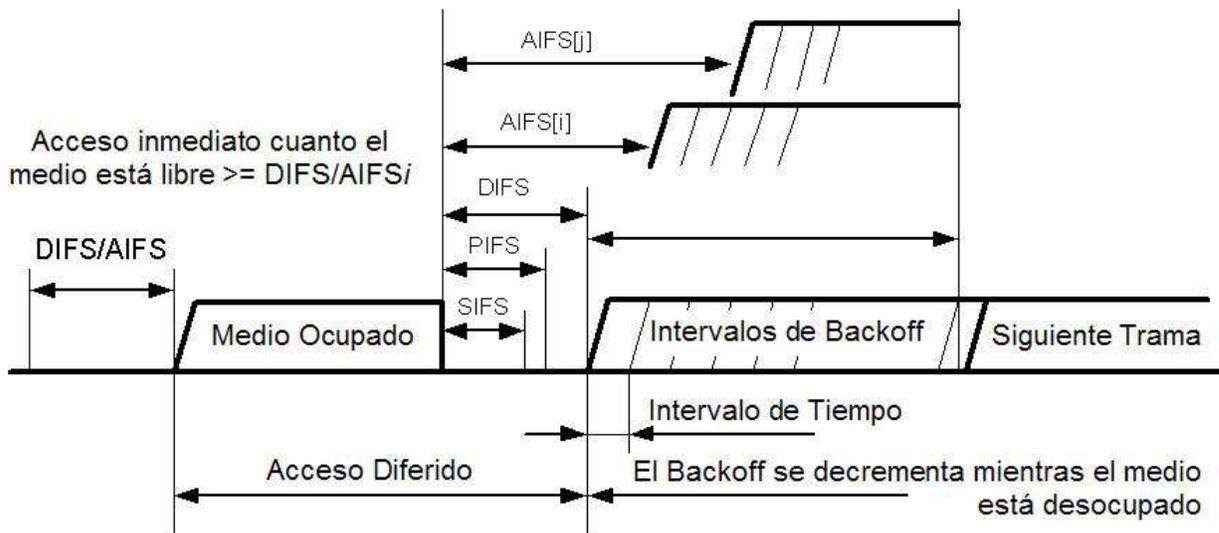


Figura 2.9 Prioridad de acceso al medio dependiente de la AC.

Cada AC tiene sus propios valores de ventana de contienda mínima y máxima ( $CW_{min}$ ,  $CW_{max}$ ), contadores de tiempo de espera aleatoria (*backoff*) y la transmisión de paquetes se realiza de manera independiente una vez que su contador de *backoff* haya llegado a cero. Es así como el tráfico con la más alta prioridad tendrá, *de una manera estadística*<sup>23</sup>, tiempos de *backoff* más cortos que el tráfico con una prioridad más baja. Dada la situación en que los contadores de *backoff* de dos colas en la misma QSTA llegan al valor de cero al mismo tiempo, la cola con la menor prioridad retrasa su transmisión y cede el derecho a transmitir a la cola con prioridad más alta. En este caso, la cola con menor prioridad asume este hecho como la presencia de una colisión en el medio de transmisión (asume que el medio está ocupado), en el estándar 802.11e a esto se le conoce como Resolución de Colisión Interna (IEEE 802.11e, 2005).

<sup>23</sup> Pero no garantizada, de allí que cobre importancia generar un procedimiento para evaluar si la modificación de los valores de parámetros de EDCA beneficia al tráfico de VoIP cuando el tráfico es alto en un QBSS.

Si una estación gana el acceso al medio durante el período de contienda, en lugar de transmitir una sola trama reconocida por un ACK, puede transmitir más tramas (de la misma cola), una tras otra, ya sea al mismo destino o a un destino diferente hasta que la Oportunidad de Transmisión (TXOP, *Transmisión Oportunidad*) finalice. Este es un intervalo de tiempo en el Período de Contienda (CP, *Contention Period*) definido por un tiempo de inicio y una duración máxima (TXOP *Limit*) durante el cual una QSTA está en la capacidad de transmitir y de tomar decisiones con respecto a qué paquetes transmitir. Las transmisiones que tienen lugar dentro de la TXOP se separan por un SIFS y se conocen como Ráfagas Libres de Contienda (CFB, *Contention Free Burst*).

El resto de estaciones actualizan su vector de reserva de recursos de red (NAV, *Network Allocation Vector*) para posponer su transmisión. La información acerca de TXOP *Limit*, CW<sub>min</sub>, CW<sub>max</sub> y AIFS, se difunde a través de un nuevo campo en la trama *Beacon* llamado *QoS Parameter Set Element*. Los valores por defecto de los parámetros de EDCA según la categoría de acceso se encuentran en la Tabla 2.5 (IEEE 802.11e-Anexo D, 2005).

AC	AIFSN		TXOP Límite	CW <sub>min</sub>	CW <sub>Max</sub>
	QSTA	QAP			
VO	2	1	0	7	15
VI	2	1	0	15	31
BE	3	3	a/g: 3008 $\mu$ s; b: 6016 $\mu$ s	31	1023
BK	7	7	a/g: 1504 $\mu$ s; b: 3264 $\mu$ s	31	1023

Tabla 2.5. Valores de los parámetros predeterminados EDCA.

La Figura 2.10 muestra gráficamente la espera mínima y la espera aleatoria por contienda que introducen los valores de los parámetros EDCA sobre las cuatro categorías de acceso. Se ilustra como el tráfico de *Background* es el que más debe sacrificar desempeño a expensas del ofrecer una mayor oportunidad a la Voz y el Video.

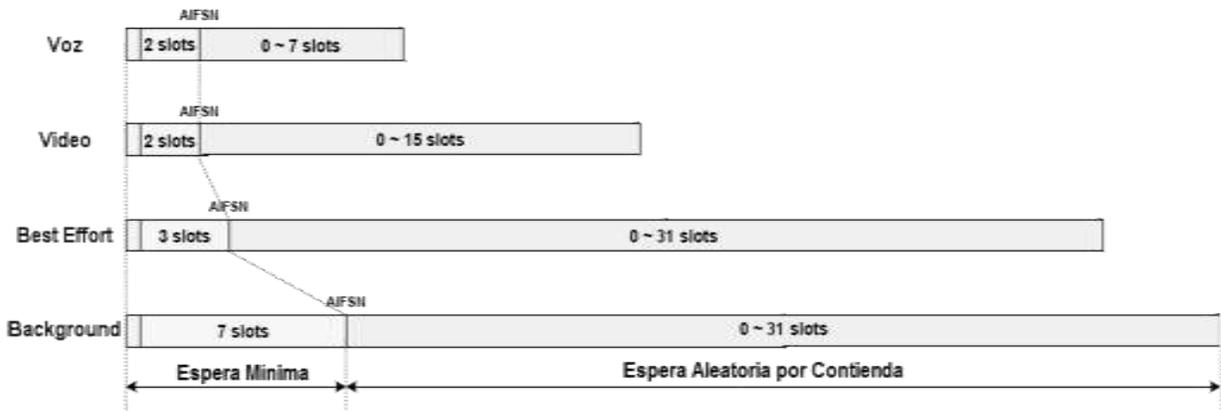


Figura 2.10 Espera mínima y aleatoria de las categorías de acceso en EDCA.

Tal como en la Tabla 2.1 se encuentran las duraciones<sup>24</sup> de los espacios entre tramas – IFS de las tecnologías 802.11a, b y g, sin QoS; en la Tabla 2.6 se presentan de forma específica las duraciones de estos mismos valores teniendo en cuenta al tipo de categoría de acceso cuando se aplica QoS.

	AIFS [VO] ( $\mu$ s)	AIFS[VI] ( $\mu$ s)	AIFS [BE] ( $\mu$ s)	AIFS [BK] ( $\mu$ s)
AIFSN	2	2	3	7
802.11a	34	34	43	79
802.11b/g	50	50	70	150
802.11g <sup>25</sup>	28	28	37	73

Tabla 2.6. Duración de los AIFS según la categoría de acceso con EDCA

Al comparar la Tabla 2.1 y la Tabla 2.6 se deduce que la probabilidad de acceso al medio de una STA 802.11b y una QSTA con tráfico en la categoría de acceso de voz (AC\_VO, 802.11b) es similar, debido a que los tiempos de contienda son los mismos (50  $\mu$ s). Esto conduce a una disminución del desempeño del estándar 802.11e cuando opera con estaciones que trabajan bajo el esquema MAC basado en DCF, es decir, cuando se tiene operación en modo mixto, ya que el acceso al medio por parte de una categoría de acceso de voz debe competir en igualdad de condiciones que las de una STA que desee transmitir tráfico de datos (tal como HTTP, SMTP, SSH, etc), dejando a un lado la prioridad del tráfico de voz sobre la transmisión del resto de estaciones. Por esta situación es muy importante que la implementación de QoS en un BSS implique que todas las estaciones soporten la capacidad establecida por EDCA de 802.11e o por la especificación WMM.

En el Apéndice A se trata HCCA y en el Apéndice B se tratan esquemas adicionales en el nivel MAC que proporciona el estándar IEEE 802.11e.

<sup>24</sup> En microsegundos -  $\mu$ s

<sup>25</sup> Sólo para redes completamente 802.11g



## 3. Evaluación de EDCA

Con lo expuesto hasta este punto se ha explicado el funcionamiento de EDCA que es el tema central de la Tesis de Maestría. Tal como se ha presentado, el acceso al medio basado en EDCA no depende de un control central sino que está distribuido entre las estaciones que participan de un QBSS en modo Infraestructura o *Ad Hoc*. El tráfico de una estación se clasifica en una de cuatro categorías de acceso como se resume en la Tabla 2.2, de acuerdo con esto cuando una estación desea transmitir debe obedecer lo indicado por los parámetros propios de cada AC explicados en la Tabla 2.4 con un conjunto de valores predeterminados establecidos en el estándar IEEE 802.11e y en la especificación WMM 1.1 de la Alianza WiFi que son los que aparecen en la Tabla 2.5. De estas cuatro categorías de acceso solamente la voz ha cobrado mayor importancia como se ve reflejado en los productos de las empresas<sup>26</sup> que han generado teléfonos con esta capacidad; el video no es todavía una aplicación que se encuentre realmente implementada y las cámaras de video WLAN actuales son en realidad dispositivos con servidores web empotrados a los que se entra por medio de un navegador para ver imágenes que cambian cada pocos segundos por lo que sería tráfico HTTP y no RTP, las otras dos categorías de acceso: la de tráfico de *background* (AC\_BK) y de mejor esfuerzo (AC\_BE) tienden en la práctica a ser combinadas solamente como BE, por tanto a nivel empresarial y del hogar el escenario actual más real es el de tener diferenciado al tráfico de VoIP. En este sentido, el ambiente en el que se realizó esta Tesis de Maestría fue un QBSS con comunicaciones de VoIP y el resto tráfico de tipo mejor esfuerzo en el sentido del cliente al AP.

### 3.1. Análisis para la Evaluación

Como en EDCA no existe un mecanismo de control de admisión que ofrezca garantía a los flujos existentes en un QBSS (la agregación de tráfico adicional degrada las comunicaciones que se estén llevando a cabo como se constata en el trabajo de Agredo y Gaviria (2006) especialmente para el tráfico inelástico o de tiempo real como VoIP), inicialmente se buscó primero responder la siguiente pregunta de Investigación:

¿Cuál es el desempeño de los valores por defecto de los parámetros EDCA en la protección del tráfico de VoIP y cuál su impacto sobre el *throughput* del tráfico de mejor esfuerzo, comparado con el mecanismo original de las WLANs?

La cual se atiende en este capítulo realizando procesos de Experimentación y por conveniencia de Simulación apoyados en la metodología que aparece explicada

---

<sup>26</sup> Indicadas en la sección Antecedentes en la Introducción

en el Apéndice H, para poder evaluar el funcionamiento del mecanismo EDCA en un QBSS con los parámetros indicados en el estándar IEEE 802.11e con respecto a un BSS que no soporte esta especificación.

## 3.2. Infraestructura para la Experimentación

Con el fin de lograr el objetivo de evaluar el funcionamiento del mecanismo EDCA en un QBSS con los parámetros indicados en el estándar IEEE 802.11e<sup>27</sup> (IEEE 802.11e, 2005) (Alianza WiFi WMM, 2005) con respecto a un BSS que no soporte esta especificación, y dada la dificultad de conseguir una infraestructura con una cantidad considerable de equipos para la experimentación, como se explicó en la sección 1.4 de la Introducción, se estableció un escenario en el que se tuviera un equipo de voz y cinco de datos, de esta manera si en un BSS este número de flujos de datos traía inconvenientes al flujo de voz (Dangerfield, *et al*, 2006), era posible verificar el aporte de EDCA para la búsqueda de QoS.

El montaje a utilizar para el proceso de evaluación experimental de EDCA contó con servidor de VoIP con la PBX IP Asterisk, un equipo cliente para las descargas de datos, un computador configurado como punto de acceso (AP) soportado por MadWiFi y seis equipos cada uno con la PBX IP Asterisk<sup>28</sup> y servidores HTTP para utilizar dependiendo de la prueba; la topología utilizada se muestra en la Figura 3.1. Los equipos tuvieron como sistema operativo Linux Ubuntu 8.04 excepto el PC 8 que era un Windows XP Professional desde donde se operaban las descargas.

Es importante mencionar que el sentido en el que se generó el tráfico fue importante dado que EDCA se trata de una contienda basada en prioridades de diversas estaciones, así, para que se pudiera evaluar los dos efectos claves para este proyecto: el desempeño de EDCA y el efecto de la variación de los parámetros por defecto, los flujos de tráfico debieron generarse en las estaciones que competían por el medio, esto se evidencia de manera rápida tomando, como ejemplo, un flujo de HTTP: en el sentido de los clientes al AP se genera el tráfico principal en el otro sentido solamente se tienen ACKs de TCP. Así las estaciones inalámbricas tuvieron que ser servidores de HTTP con un cliente en la red cableada que generara el sentido del tráfico, este fue el PC8 de la Figura 3.1.

---

<sup>27</sup> Mismos que aparecen en la especificación WMM de la Alianza WiFi

<sup>28</sup> Tal como se explica en (Agredo y Gaviria, 2006) es necesario que todos los equipos tengan Asterisk para utilizar IAX y hacer uso de comandos como el `iax2 show netstats` que permiten obtener datos sobre retardo, pérdidas y *jitter*.

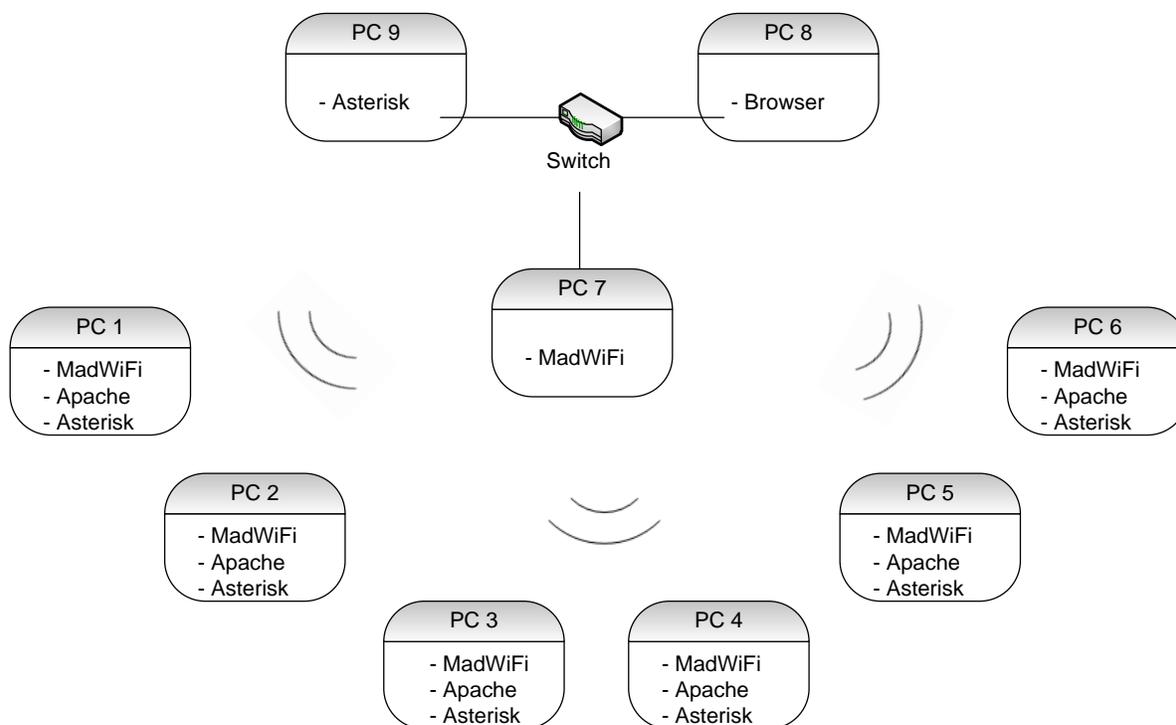


Figura 3.1 Diagrama en bloques de la infraestructura para experimentación.

Se adecuaron los diferentes equipos para recibir/enviar flujos exclusivos de voz para lo cual se utilizó la PBX IP Asterisk con flujos de VoIP donde el protocolo de tiempo real (RTP, *Real Time Protocol*) empleaba como códec la especificación G.711<sup>29</sup> por ser la que con menos flujos saturaba la red (Agredo y Gaviria, 2006) y como protocolo de señalización a IAX<sup>30</sup> para de esta manera establecer la comunicación entre servidores Asterisk en modo cliente – servidor. Este aspecto fue uno de los más críticos del proceso de experimentación, pero se considera uno de sus principales aportes, puesto que era necesario contar con un flujo de VoIP marcado en nivel 3 de forma que cuando el driver MadWiFi lo mapeara a nivel dos adquiriera el tratamiento de categoría de voz (AC\_VO) en la parte inalámbrica con WMM/802.11e-EDCA y la prioridad de usuario en la parte cableada 802.3 con soporte para 802.1Q/p con VoIP. En el Apéndice C se describe la configuración de Asterisk para la generación de flujos de Voz.

Para la medición de los parámetros de calidad del flujo de VoIP se utilizaron los comandos `iax2 show channels` y `iax2 show netstats` de Asterisk que presentan la pérdida de paquetes, el retardo y el *jitter* de las llamadas (flujos de VoIP), soportándose para esto en los datos que suministra el protocolo RTCP<sup>31</sup> de las llamadas tanto en el cliente como en el servidor para verificar que se encuentren dentro de los límites (Agredo y Gaviria, 2006).

<sup>29</sup> Pues al ser la que ma

<sup>30</sup> Inter Asterisk eXchange

<sup>31</sup> Protocolo de Control de Tiempo Real

Es importante recordar que para que el mecanismo de QoS de EDCA sea efectivo y funcione es necesario que los paquetes de las categorías de acceso de Voz, Video y *Background* sean marcados, en el caso específico de este proyecto se trabajó con la categoría de Voz motivo por el cual se recurrió a Asterisk que como se presenta en el Apéndice C puede configurarse para marcar los paquetes de voz de tal forma que cuando los drivers de MadWiFi hagan el mapeo al nivel MAC se tenga la marca de prioridad de usuario 6 que corresponde a la Voz y pueda entonces convertirse en la red inalámbrica en flujos de categoría de Voz (AC\_VO) para recibir el tratamiento prioritario establecido por los parámetros de EDCA. La Figura 3.2 evidencia la marcación para la diferenciación de servicios que es obtenida de la captura de paquetes con el Sniffer Commview™, en esta trama aparece como tipo (*Type*) la opción 2 que corresponde a Datos (*Data*) y como subtipo (*Subtye*) la opción 8 que corresponde a Datos con QoS (*Qos Data*), adicionalmente en el campo de Control de QoS (*QoS Control*) el valor es 6 que identifica a un flujo de categoría de voz (AC\_VO) como se indicó en la tabla 2.3. En una trama de datos que no soporte QoS, el sub-tipo aparecería con el valor 0 correspondiendo a Datos y no se tendría el campo de control de QoS.

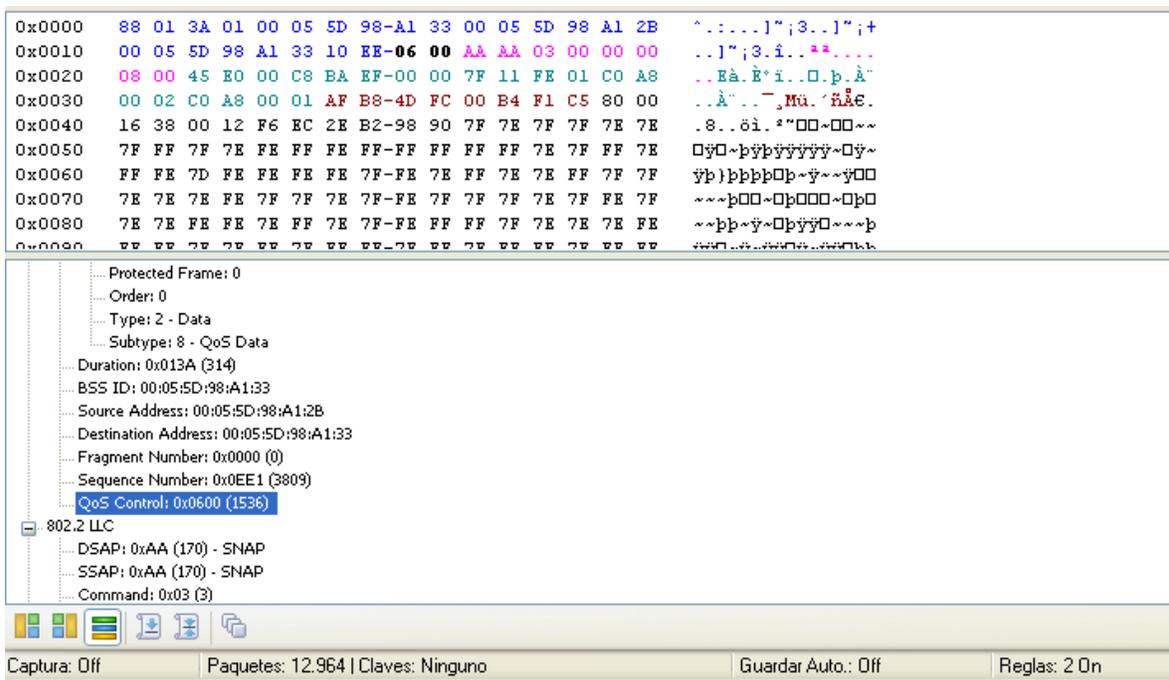


Figura 3.2 Evidencia de la marcación de las tramas de voz.

### 3.3. Desarrollo de la Evaluación

Los requerimientos que se deben cumplir cuando se trabaja con voz sobre IP son:

- Retardo máximo en un sentido no mayor a 150 ms (de acuerdo con la recomendación ITU-T G.114).

- Pérdida de paquetes mínima: VoIP no es tolerante a las pérdidas de paquetes, aún con un 1% de paquetes perdidos se puede degradar enormemente una comunicación de voz así se esté utilizando el códec G.711, en caso de utilizar códecs con mayor tasa de compresión la pérdida es prácticamente intolerable (Szigeti y Hattingh, 2005) (Intel, 2005). Sin embargo, en (Dangerfield, et al, 2006) se considera tolerable hasta el 10% de pérdida.
- El *jitter* promedio no debe ser mayor que 30 ms (Szigeti y Hattingh, 2005) aunque algunos autores hablan de hasta 50 ms (Chou, 2005) (Mlinarsky, 2005).

Partiendo de lo anterior, para evaluar el funcionamiento del mecanismo EDCA en un QBSS con los parámetros por defecto con respecto a un BSS que no soporte esta especificación, una vez lista la infraestructura para experimentación se procedió con la generación de un flujo de voz y cinco flujos de datos, inicialmente sin activar el soporte para WMM del driver MadWiFi para Linux.

### Resultados:

La Figura 3.3 muestra el resultado que aparece en el terminal luego de ejecutar los comandos<sup>32</sup>

```
iax2 show channels
iax2 show netstats
```

Comandos<sup>33</sup> que permiten tener el resultado en cuanto a retardo<sup>34</sup> y la pérdida de paquetes<sup>35</sup> para verificación del cumplimiento de los límites para dar garantía del cumplimiento de QoS.

<sup>32</sup> Desde la consola de Asterisk.

<sup>33</sup> Se ejecuta en el PC como AP (PC7 – ipet7) por eso debe verse la columna REMOTE

<sup>34</sup> Aparece como Del por abreviar *Delay*

<sup>35</sup> Aparece como *Lost*

```

Asterisk Console on 'ipet7' (pid 6816)
Archivo Editar Ver Terminal Solapas Ayuda
*CLI> iax2 show channels -- Executing [s@tutorial:2] Goto("IAX2/cliente3-23", "tutorial|s|1") in new stack
-- Goto (tutorial,s,1)
-- Executing [s@tutorial:1] Background("IAX2/cliente3-23", "demo-instruct") in new stack
-- <IAX2/cliente3-23> Playing 'demo-instruct' (language 'en')
channels
Channel      Peer           Username      ID (Lo/Rem)  Seq (Tx/Rx)  Lag      Jitter  JitBuf  Format
IAX2/cliente3-23  172.16.15.103 cliente3      00023/00001  00021/00020  00253ms  0189ms  0098ms  ulaw
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente3-23  319  36  98   4    0    0    0    3  189  253  44    0    2  113  3
1 active IAX channel

```

Figura 3.3 Resultado de un solo flujo de VoIP con 802.11b sin WMM/EDCA.

Luego se activó la capacidad de WMM en el los equipos con el comando

```
iwpriv ath0 wmm 1 36
```

La Figura 3.4 muestra el resultado que aparece en el terminal pero en esta ocasión teniendo habilitado el soporte para WMM/EDCA

```

Asterisk Console on 'ipet7' (pid 6816)
Archivo Editar Ver Terminal Solapas Ayuda
*CLI> iax2 show channels
Channel      Peer           Username      ID (Lo/Rem)  Seq (Tx/Rx)  Lag      Jitter  JitBuf  Format
IAX2/cliente3-2  172.16.15.103 cliente3      00002/00002  00011/00010  00220ms  0021ms  0062ms  ulaw
(None)       172.16.15.107 (None)       00004/00000  00001/00000  00000ms  -0001ms  0000ms  unknow
(None)       172.16.15.109 (None)       00015/00000  00001/00000  00000ms  -0001ms  0000ms  unknow
(None)       172.16.15.108 (None)       00016/00000  00001/00000  00000ms  -0001ms  0000ms  unknow
(None)       172.16.15.106 (None)       00017/00000  00001/00000  00000ms  -0001ms  0000ms  unknow
(None)       172.16.15.104 (None)       00018/00000  00001/00000  00000ms  -0001ms  0000ms  unknow
6 active IAX channels
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente3-2  4    22  62   0    0    0    0    1  155  220  2    1    0  12  1
1 active IAX channel

```

Figura 3.4 Resultado de un solo flujo de VoIP con 802.11b con WMM/EDCA.

### Análisis de Resultados:

La información obtenida en los resultados fue muy gratificante, pues, evidenció que con la implementación de WMM/EDCA las medidas reportadas sobre retardo y pérdida de paquetes disminuyeron. Específicamente:

1. El retardo pasó de 253 ms a 220 ms
2. Las pérdidas pasaron de 44 a 2
3. El *jitter* pasó de 189 ms a 155 ms

<sup>36</sup> Por su parte el comando `iwpriv ath0 wmm 0` deshabilita WMM

Sin embargo, aún superaban los límites de QoS lo cual daba toda la justificación al objetivo de la Tesis de buscar valores de parámetros EDCA que contribuyeran a satisfacer los requerimientos de QoS en flujos de VoIP, lo cual será tratado en el capítulo 4.

Basado en lo anterior y con la intención de evaluar en otro escenario el funcionamiento de WMM/EDCA se procedió a realizar otra experimentación, para la que se dispuso de 3 flujos de voz y 3 flujos de datos.

### Resultados:

La Figura 3.5 muestra el resultado que aparece en el terminal luego de ejecutar los comandos teniendo 3 flujos de voz y 3 de datos activos en el BSS<sup>37</sup>.

```
Asterisk Console on 'ipet7' (pid 6816)
Archivo Editar Ver Terminal Solapas Ayuda
*CLI> iax2 show channels -- Executing [s@tutorial:2] Goto("IAX2/cliente3-22", "tutorial|s|1") in new stack
-- Goto (tutorial,s,1)
-- Executing [s@tutorial:1] Background("IAX2/cliente3-22", "demo-instruct") in new stack
-- <IAX2/cliente3-22> Playing 'demo-instruct' (language 'en')
channels
Channel      Peer           Username      ID (Lo/Rem)  Seq (Tx/Rx)  Lag       Jitter  JitBuf  Format
IAX2/cliente6-10  172.16.15.106 cliente3      00010/00003  00021/00020  00246ms  0023ms  0083ms  ulaw
IAX2/cliente5-19  172.16.15.105 cliente3      00019/00003  00017/00016  00141ms  0030ms  0082ms  ulaw
IAX2/cliente3-22  172.16.15.103 cliente3      00022/00002  00021/00020  00307ms  0027ms  0083ms  ulaw
3 active IAX channels
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente6-10  141  23  83   0    0    0    1    3  197  246  37    1    5  31    3
IAX2/cliente5-19  248  30  82   0    0    0    0    2   96  141   3    0    1  15    2
IAX2/cliente3-22  148  27  83   1    0    0    1    3  232  307  14    0    1  99    3
3 active IAX channels
```

Figura 3.5 Resultado de 3 flujos de VoIP y Datos con 802.11b sin WMM/EDCA.

La Figura 3.6 muestra el resultado que aparece en el terminal luego de ejecutar los comandos teniendo 3 flujos de voz y 3 de datos activos en el QBSS<sup>38</sup>.

```
Asterisk Console on 'ipet7' (pid 6816)
Archivo Editar Ver Terminal Solapas Ayuda
*CLI> iax2 show channels
Channel      Peer           Username      ID (Lo/Rem)  Seq (Tx/Rx)  Lag       Jitter  JitBuf  Format
IAX2/cliente5-3  172.16.15.105 cliente3      00003/00002  00011/00010  00060ms  0018ms  0061ms  ulaw
IAX2/cliente3-11  172.16.15.103 cliente3      00011/00003  00011/00010  00060ms  0015ms  0062ms  ulaw
IAX2/cliente6-20  172.16.15.106 cliente3      00020/00002  00009/00008  00062ms  0009ms  0065ms  ulaw
3 active IAX channels
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-3  2   16  61   4    0    0    0    1    7  60   0    0    0    0    1
IAX2/cliente3-11  7   15  61   2    0    0    0    1    7  60   2    0    0    0    1
IAX2/cliente6-20  2    9  65   0    0    0    0    1    7  62   0    0    0    0    1
3 active IAX channels
```

Figura 3.6 Resultado de 3 flujos de VoIP y Datos con 802.11b con WMM/EDCA.

<sup>37</sup> sin WMM/EDCA  
<sup>38</sup> con WMM/EDCA

### Análisis de Resultados:

En esta ocasión la revisión de las columnas `Jit`, `Del` y `Lost`, en especial las dos primeras, evidencian que con el uso de WMM/EDCA las medidas de *Jitter*, Retardo y Pérdidas, respectivamente, cumplen con los límites establecidos para VoIP.

En específico si se compara la primera línea luego de ejecutar el comando `iax2 show netstats` que inicia con IAX2/Cliente6-10 en la Figura 3.5 con la equivalente en la Figura 3.6 que inicia IAX2/Cliente5-3<sup>39</sup> los cambios que se tienen para el flujo son:

1. El retardo pasó de 246 ms a 60 ms
2. Las pérdidas pasaron de 37 a 0
3. El *jitter* pasó de 197 ms a 7 ms

Para este caso específico donde el *throughput* de cada flujo de datos estaba en promedio alrededor de los 1.3 Mbps<sup>40</sup> era lógico esperar que al reemplazar dos flujos de datos por dos de voz, se obtendría este tipo de resultado por la menor demanda de ancho de banda, pues, el consumo de cada flujo de VoIP oscilaba apenas alrededor de los 80 Kbps.

Lo más importante que se encontró es que WMM/EDCA si beneficia al tráfico de mayor prioridad y por tanto ofrece un mejor desempeño a las aplicaciones de tiempo real como VoIP, sin embargo, se evidencia la necesidad de realizarse un ajuste a los valores de los parámetros para contribuir a satisfacer los requerimientos de QoS, lo cual será atendido en el Capítulo 4.

Para determinar el impacto sobre el *throughput* del tráfico de mejor esfuerzo se tomó el valor generado por el *log* del gestor de descarga para cada archivo transferido, los resultados aparecen en la Tabla 3.1.

EDCA	ipet45	ipet46	ipet51	ipet55	ipet58	<i>Throughput</i> Promedio (Kbytes/sg)
No	166,3	172,9	172,2	168,4	180	172,0
Si	168,4	166,3	185	166,3	166,8	170,6

Tabla 3.1. Velocidad promedio<sup>41</sup> global de las descargas<sup>42</sup> sin EDCA y con EDCA.

<sup>39</sup> Los identificadores de IAX2 se asignan con cada inicio de flujo, razón por la cual son diferentes en cada prueba.

<sup>40</sup> Como se puede determinarse a partir de la tabla 3.1

<sup>41</sup> En Kbytes/segundo

<sup>42</sup> De datos.

Como se aprecia, se presenta una disminución en el promedio del *throughput* global de datos, resultante de dar un tratamiento prioritario a los flujos de VoIP.

### 3.4. Evaluación por Simulación

Como se explicó en el Anteproyecto de la Tesis de Maestría y en la Introducción de este documento, la simulación se utilizaría cuando se considera conveniente o necesaria, en este caso puntual, se considera conveniente con fines de comparación con los resultados obtenidos por Experimentación.

Para proceder con la Simulación se recurrió al Simulador de Red NCTUns en su versión 4.0 liberada en el año 2007, en el Apéndice F se ha generado una guía sobre su manejo y es especialmente útil como referencia, pues, este simulador adolece de documentación en sus capacidades avanzadas.

En específico para realizar la evaluación por simulación del beneficio de contar con el mecanismo de QoS EDCA en una red inalámbrica frente a no tenerlo para el tráfico de VoIP, se estableció una topología equivalente a la del montaje realizado en laboratorio. Y mientras en experimentación el soporte EDCA se habilitaba o deshabilitaba dependiendo de la situación, en el Simulador se hizo uso de dispositivos/elementos diferentes, de esta forma en la Figura 3.7 se muestra la topología para evaluar la red 802.11b sin EDCA y en la Figura 3.8 con EDCA. Nótese que los dispositivos inalámbricos tienen una “e” acompañando al elemento cuando soportan 802.11e.

Debe recordarse que las cuatro categorías de acceso de WMM/EDCA tienen una correspondencia con las 8 prioridades de usuario, tal como aparece en la Tabla 2.3. Esto es importante porque la interfaz del NCTUns pregunta la prioridad de usuario cuando se adicionan estaciones 802.11e a la topología.

Aprovechando las facilidades de la herramienta de simulación gracias a las que, a diferencia de la experimentación, se pueden agregar cuantos equipos (*Hosts*) se requieran, se dispusieron en la parte cableada, tantos como fuese necesario para generar una correspondencia uno a uno, lo cual permitió tener un archivo de registro (*log*) diferenciado para cada uno de los flujos en el equipo receptor y/o transmisor.

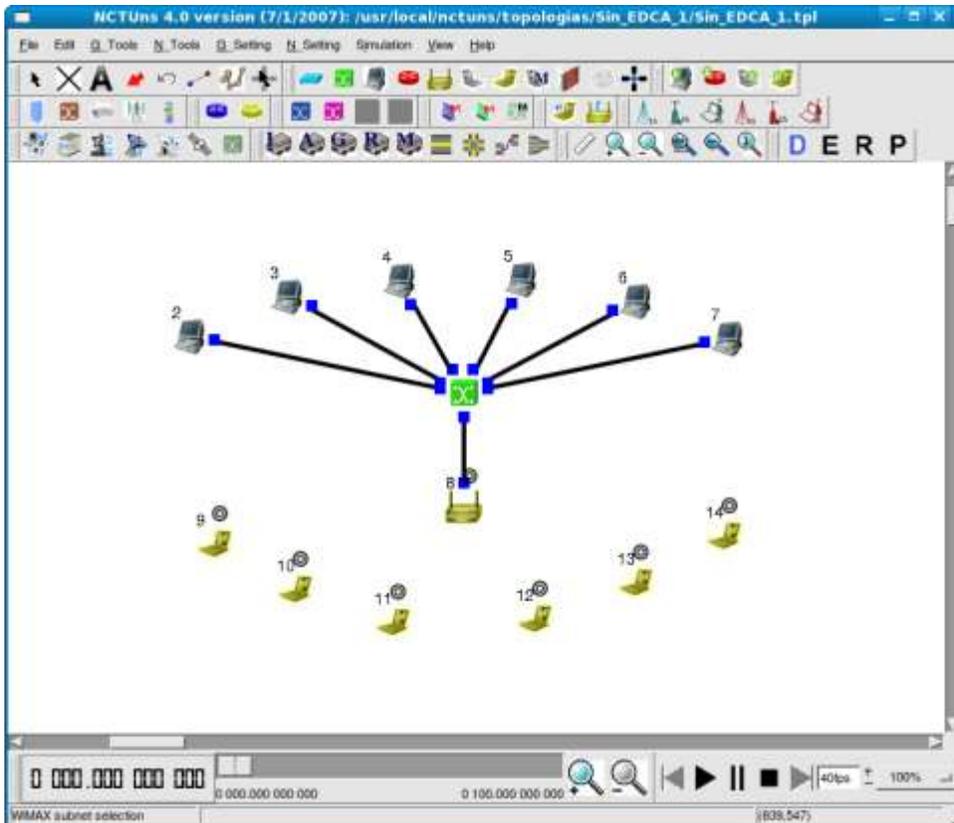


Figura 3.7 Topología de simulación de una red 802.11b sin soporte para EDCA.

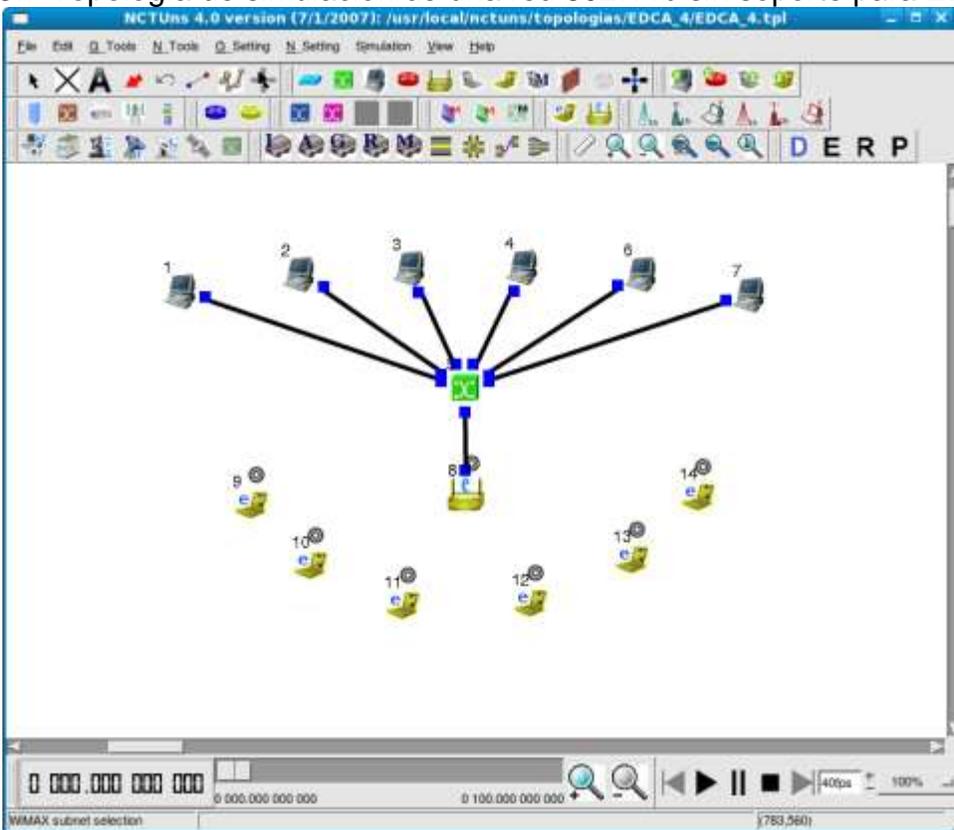


Figura 3.8 Topología de simulación de una red 802.11b con soporte para EDCA.

De esta forma, el equipo identificado con el número 14 enviaba un flujo de VoIP con trazo del códec G.711 para hacerlo equivalente con la experimentación y en el equipo identificado con el número 7 se generaba el archivo de registro con la información sobre retardo y pérdida de paquetes, todos los demás equipos generaban flujos de datos TCP configurados para ser equivalentes al comportamiento de HTTP<sup>43</sup>, pues el simulador no tiene explícitamente este protocolo. El sentido de los flujos era desde los equipos inalámbricos hacia los equipos de la red cableada, como ya se ha explicado antes, para que se diera la contienda por el medio en la parte inalámbrica.

La generación de los flujos de TCP<sup>44</sup> se hizo con el comando `ttcp` y el de UDP<sup>45</sup> con la combinación de comandos `stg` y `rtg`, tal como aparece explicado en el Apéndice F.

### Resultados:

Los resultados para la simulación de la topología sin soporte para EDCA se obtuvieron del archivo de registro de la estación de VoIP en la que se configuró se registrará la pérdida de paquetes y el retardo con el comando `rtg` que aparece a continuación:

```
rtg -u -p 8006 -o Retardo_Perdidas_SinEDCA.log
```

Aunque la sintaxis se explica en el Apéndice F, este comando permite que del tráfico UDP (`-u`) que llegue por el puerto 8006 (`-p 8006`) se registre la información de pérdidas y retardo en el archivo `Retardo_Perdidas_SinEDCA.log`

La Figura 3.9 muestra el final de este archivo de registro donde se puede visualizar medidas de retardos que están por encima de los 150 ms y entre el total de paquetes enviados durante el tiempo de simulación (4950) y el acumulado de paquetes perdidos (516), da un porcentaje de pérdida de 10,4%, medidas estas que evidencian y corroboran lo establecido en la experimentación

Los resultados de la simulación de la topología con soporte para EDCA se obtuvieron de forma similar, pero en este caso se hizo uso de la topología de la Figura 3.8. El comando equivalente que se estableció en la estación receptora del flujo VoIP fue:

```
rtg -u -p 8006 -o Retardo_Perdidas_ConEDCA.log
```

<sup>43</sup> Que se transporta sobre TCP

<sup>44</sup> Para HTTP (Datos – AC\_BE)

<sup>45</sup> Para RTP (VoIP – AC\_VO)

La Figura 3.10 muestra el final de este archivo de registro donde se puede visualizar que en esta ocasión algunos de retardos están por encima de los 150 ms pero se repiten con menos frecuencia y con valores que no sobrepasan los 200 ms<sup>46</sup> y no se presenta pérdida de paquetes.

#### Análisis de Resultados:

De acuerdo con lo obtenido, en las Figuras 3.11 y 3.12, se presentan respectivamente el máximo retardo y las pérdidas obtenidas sin EDCA y con EDCA, se puede observar al trabajar con simulación estableciendo una topología equivalente a la de experimentación, se encuentra que una estación de VoIP tendría problemas para cumplir los límites de QoS si se tiene al mismo tiempo 5 flujos de datos y no se está habilitado WMM/EDCA en la red, pero de estarlo es notoria la mejora que puede llegar a lograrse.

Comparando lo obtenido en la Experimentación con lo obtenido por Simulación, y sin ser los resultados idénticos, se ve claramente el aporte que WMM/EDCA hace a una red inalámbrica WiFi para el funcionamiento mejorado de un QBSS de VoWLAN. Ambos procedimientos muestran una mejora evidente pero también permiten identificar que es necesario un afinamiento con el que se pueda lograr un mejor desempeño de VoIP, aspecto que se tratará en el capítulo siguiente.

---

<sup>46</sup> Como si les ocurría a un número considerable de paquetes en el caso sin EDCA.

```

Retardo_Perdidas_SinEDCA.log - Bloc de notas
Archivo Edición Formato Ver Ayuda
pkt no.4921 size= 160 byte, delay time= 0.19798 sec
pkt no.4922 size= 160 byte, delay time= 0.08793 sec
pkt no.4923 size= 160 byte, delay time= 0.05229 sec
pkt no.4924 size= 160 byte, delay time= 0.03731 sec
pkt no.4925 size= 160 byte, delay time= 0.07137 sec
pkt no.4926 size= 160 byte, delay time= 0.04232 sec
pkt no.4927 size= 160 byte, delay time= 0.29241 sec
pkt no.4928 size= 160 byte, delay time= 0.17454 sec
pkt no.4929 size= 160 byte, delay time= 0.13319 sec
pkt no.4930 size= 160 byte, delay time= 0.03128 sec
packet no.4931 to no.4932 lost, accumulate 516 pkts lost!!
pkt no.4933 size= 160 byte, delay time= 0.12776 sec
pkt no.4934 size= 160 byte, delay time= 0.01900 sec
pkt no.4935 size= 160 byte, delay time= 0.11684 sec
pkt no.4936 size= 160 byte, delay time= 0.17299 sec
pkt no.4937 size= 160 byte, delay time= 0.05717 sec
pkt no.4938 size= 160 byte, delay time= 0.02163 sec
pkt no.4939 size= 160 byte, delay time= 0.06701 sec
pkt no.4940 size= 160 byte, delay time= 0.23339 sec
pkt no.4941 size= 160 byte, delay time= 0.12946 sec
pkt no.4942 size= 160 byte, delay time= 0.09545 sec
pkt no.4943 size= 160 byte, delay time= 0.05295 sec
pkt no.4944 size= 160 byte, delay time= 0.07437 sec
pkt no.4945 size= 160 byte, delay time= 0.24100 sec
pkt no.4946 size= 160 byte, delay time= 0.05593 sec
pkt no.4947 size= 160 byte, delay time= 0.14589 sec
pkt no.4948 size= 160 byte, delay time= 0.02761 sec
pkt no.4949 size= 160 byte, delay time= 0.27006 sec
pkt no.4950 size= 160 byte, delay time= 0.08566 sec

```

Figura 3.9 Archivo de registro de la simulación sin soporte para EDCA.

```

Retardo_Perdidas_ConEDCA.log - Bloc de notas
Archivo Edición Formato Ver Ayuda
pkt no.4923 size= 160 byte, delay time= 0.18692 sec
pkt no.4924 size= 160 byte, delay time= 0.06135 sec
pkt no.4925 size= 160 byte, delay time= 0.08523 sec
pkt no.4926 size= 160 byte, delay time= 0.08411 sec
pkt no.4927 size= 160 byte, delay time= 0.05341 sec
pkt no.4928 size= 160 byte, delay time= 0.05770 sec
pkt no.4929 size= 160 byte, delay time= 0.08016 sec
pkt no.4930 size= 160 byte, delay time= 0.05026 sec
pkt no.4931 size= 160 byte, delay time= 0.05179 sec
pkt no.4932 size= 160 byte, delay time= 0.06909 sec
pkt no.4933 size= 160 byte, delay time= 0.18913 sec
pkt no.4934 size= 160 byte, delay time= 0.04345 sec
pkt no.4935 size= 160 byte, delay time= 0.05267 sec
pkt no.4936 size= 160 byte, delay time= 0.05426 sec
pkt no.4937 size= 160 byte, delay time= 0.04370 sec
pkt no.4938 size= 160 byte, delay time= 0.05760 sec
pkt no.4939 size= 160 byte, delay time= 0.04821 sec
pkt no.4940 size= 160 byte, delay time= 0.06452 sec
pkt no.4941 size= 160 byte, delay time= 0.04433 sec
pkt no.4942 size= 160 byte, delay time= 0.08243 sec
pkt no.4943 size= 160 byte, delay time= 0.05460 sec
pkt no.4944 size= 160 byte, delay time= 0.06213 sec
pkt no.4945 size= 160 byte, delay time= 0.05893 sec
pkt no.4946 size= 160 byte, delay time= 0.05423 sec
pkt no.4947 size= 160 byte, delay time= 0.06324 sec
pkt no.4948 size= 160 byte, delay time= 0.06809 sec
pkt no.4949 size= 160 byte, delay time= 0.05103 sec
pkt no.4950 size= 160 byte, delay time= 0.04296 sec

```

Figura 3.10 Archivo de registro de la simulación con soporte para EDCA.

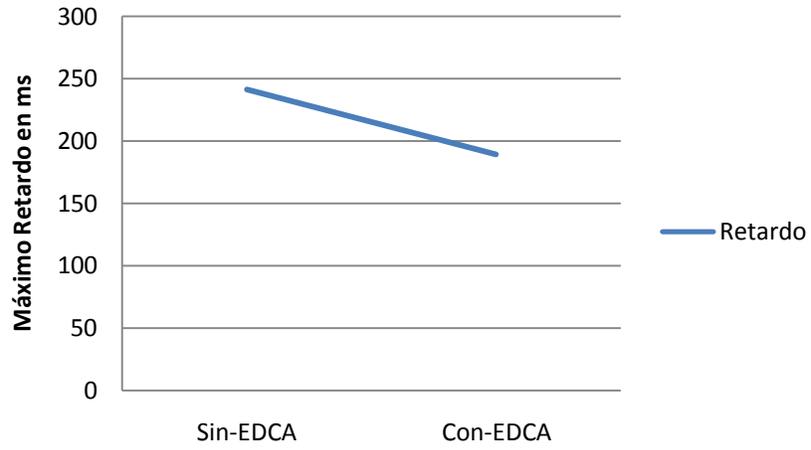


Figura 3.11 Máximo retardo en la simulación sin EDCA y con EDCA.

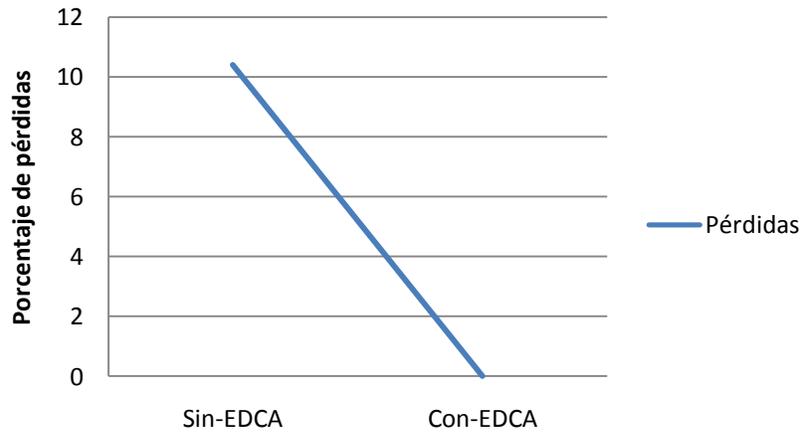


Figura 3.12 Porcentaje de pérdidas en la simulación sin EDCA y con EDCA.

## 4. Selección de Valores de Parámetros EDCA

Luego de la evaluación realizada a EDCA en el capítulo anterior, en este capítulo se presenta el procedimiento generado para resolver la pregunta de Investigación:

¿Qué valores de parámetros EDCA: 1) pueden de mejor manera<sup>47</sup> contribuir a satisfacer los requisitos de VoIP en cuanto a retardo y pérdida de paquetes cuando el nivel de tráfico del QBSS sea alto tanto con tráfico de *best-effort* como con tráfico de VoIP y 2) pueden facilitar el logro de *throughputs* más altos en el tráfico de mejor esfuerzo cuando el nivel de tráfico de VoIP sea nulo<sup>48</sup>?

Lo anterior determinó tres escenarios caracterizados por:

1. Una comunicación de Voz y las demás comunicaciones de datos
2. Una comunicación de Datos y las demás de voz
3. Solamente comunicaciones de Datos sin Voz

Los dos primeros se constituyeron en los ambientes de tráfico alto de datos y voz, y el tercero tráfico exclusivo de datos.

Debe recordarse que el estándar IEEE 802.11e establece que ante variación en las condiciones de tráfico de un QBSS deben realizarse ajustes a los valores por defecto de los parámetros EDCA para obtener mejores desempeños de acuerdo con el interés, sin embargo, el vacío es que no define ni los parámetros que deben cambiarse ni la forma de hacerlo, por lo que con el desarrollo de esta Tesis de Maestría se buscó contribuir a llenar dicho vacío, en especial determinando que parámetros variar, generando un procedimiento para hacerlo, obteniendo valores y estableciendo si era justificable este proceso en términos del logro de mejor desempeño en las flujos de tráfico objetivo.

### 4.1. Análisis para la Generación de Hipótesis

Del trabajo de estudio del funcionamiento de EDCA, el objetivo de sus parámetros, la evaluación tratada en el capítulo anterior y en general del camino recorrido se concluyó “preliminarmente” que

1. Si el QBSS solamente tiene tráfico de una Categoría de Acceso como la Voz, EDCA no es solución a problemas que se puedan presentar con las comunicaciones de voz en alto tráfico; para los flujos de mejor esfuerzo se tendría una disminución de la velocidad efectiva especialmente por el mayor espacio entre tramas para tomar el medio.

---

<sup>47</sup> Con respecto a los valores por defecto de EDCA

<sup>48</sup> Se recuerda que se cambió de “bajo” a “nulo” para no entrar en la paradoja de buscar optimización de *best effort* y cumplimiento de requerimientos de VoIP ya que estos dos objetivos son disyuntos o inversamente proporcionales.

2. Si se tienen flujos tanto de voz como de mejor esfuerzo, altos niveles de tráfico en los primeros podrían llegar a producir anulación de los segundos o en otras palabras imposibilidad para lograr el medio; por otra parte, altos niveles de tráfico en los segundos podrían afectar el cumplimiento de los límites de retardo, pérdidas de paquetes y *jitter* por lo que no se tendría garantía en las comunicaciones.
3. Cuando el tráfico del QBSS es alto en cuanto a VoIP, los flujos de la categorías de acceso de voz [AC\_VO] pueden presentar altas colisiones debido a los valores relativamente bajos de CWmin y CWmax, por lo que se degradaría el desempeño de esta AC incidiendo posiblemente en la pérdida de paquetes y podría ser un indicativo para suponer una debilidad al realizar la diferenciación de servicios basada en estos parámetros.
4. Cuando el nivel de tráfico sea bajo, el CWmax no tendría un impacto notorio porque para las estaciones no sería necesario llegar hasta él, pero si se tiene un QBSS congestionado podría ser útil para disminuir las tasas de colisión de flujos de alta prioridad [AC\_VO].
5. El uso del AIFS para la diferenciación de servicios con respecto al uso de CWmin y CWmax podría ser más eficiente por motivos como:
  - a. Pueden controlarse de manera más consistente el desempeño de una AC porque el valor del AIFS permanece fijo y no tienen un comportamiento estocástico como con la CW.
  - b. El enfoque de intervalos de tiempo dada por el AIFS podría considerarse más efectivo para controlar el comportamiento de un determinado flujo.
  - c. El impacto del AIFS podría ser más consistente que el de CWmin y CWmax con respecto a las condiciones de tráfico del QBSS.

Por lo anterior, para el primer escenario planteado se decidió cambiar el valor del AIFS[AC\_BE] con la intención de contribuir a que el flujo de VoIP pueda satisfacer los requerimientos de QoS; para el segundo escenario, se modificó el valor del CWmax [AC\_VO] con la intención de verificar la disminución en las colisiones y para el tercer escenario dado que no se tendrá tráfico de VoIP se disminuyó el valor del AIFS[AC\_BE]. La enunciación de las hipótesis a validar aparece respectivamente en cada una de las secciones siguientes a la sección 4.2.

## 4.2. Infraestructura para la Experimentación

Para el desarrollo del procedimiento de selección que permitiera encontrar los valores adecuados de los parámetros EDCA que sirvieran a los propósitos de este proyecto explicados en la Introducción y al inicio de este capítulo, se utilizó la misma infraestructura del capítulo 3 la cual se reproduce en la Figura 4.1

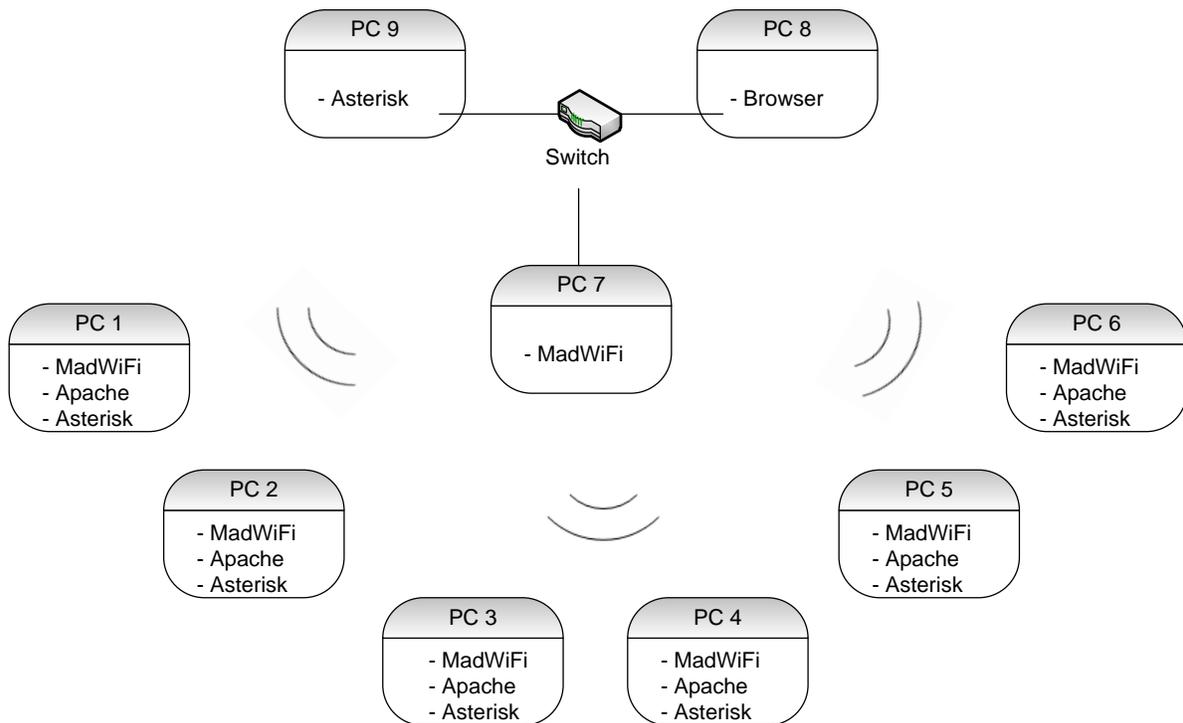


Figura 4.1 Diagrama en bloques de la infraestructura para experimentación

Una vez instalada la infraestructura con las herramientas indicadas, se inició una fase de pruebas generales para asegurar el adecuado funcionamiento de la misma, las cuales determinarían las condiciones bajo las cuales estaría sujeto el desarrollo de la experimentación.

A continuación se detalla cada una de estas pruebas incluyendo los problemas técnicos que pudieron llegar a impedir la ejecución del proyecto y sus soluciones:

Inicialmente se configuró el PC 7 de la Figura 4.1 como Access Point (modo AP) y el PC 1 como estación (modo STA); en el caso del PC 7 es necesario ejecutar los comandos que se encuentran en el *script* para la puesta en funcionamiento de un PC como Punto de Acceso en Linux (Apéndice D). Una vez ejecutado dicho *script* se hace que el AP trabaje con 802.11b para que la red de trabajo quede lista para ser utilizada, con el siguiente comando:

```
iwpriv ath0 mode 2
```

En el PC 1 no fue necesario realizar algún cambio porque el modo de operación por defecto de MadWiFi es STA. Se procedió entonces, a asociar a PC 1 con el AP soportado por PC 7 asegurando que se realizara de forma efectiva la conexión.

En este punto deben comenzar a utilizarse comandos propios de la operación del Driver MadWiFi (MadWiFi, 2007) para iniciar el proceso.

Lo primero que debe hacerse es verificar que la característica de QoS esté habilitada<sup>49</sup> o en otras palabras que soporte EDCA y como se dijo en el capítulo 2 asegurarse que todos los equipos en el QBSS lo hagan para que se puedan respetar las prioridades de los parámetros. Para lo cual se ejecuta el comando:

```
iwpriv ath0 get_wmm
```

Si el resultado es que la QoS está en 0, se debe cambiar este valor a 1 (para habilitar QoS) por medio de:

```
iwpriv ath0 wmm 1
```

Hecho lo anterior se tiene la seguridad que los equipos están soportando EDCA.

Con el comando que se muestra a continuación, se verificó los valores en que se encuentran por defecto los parámetros EDCA, tanto en el PC 1 como en el PC 7,

```
wlanconfig ath0 list wme
```

La ejecución de este comando arroja una lista de los parámetros EDCA cada uno con sus respectivos valores por defecto. La primera fila de cada parámetro corresponde a los valores locales y la segunda a los valores del equipo remoto. En la Figura 4.2 se muestra un ejemplo de las respuestas a este comando.

AP

```
AC_BE cwmín 5 cwmáx 7 aifs 3 txopLimit 0
      cwmín 5 cwmáx 10 aifs 3 txopLimit 0
AC_BK cwmín 5 cwmáx 10 aifs 7 txopLimit 0
      cwmín 5 cwmáx 10 aifs 7 txopLimit 0
AC_VI cwmín 4 cwmáx 5 aifs 1 txopLimit 6016
      cwmín 4 cwmáx 5 aifs 2 txopLimit 6016
AC_VO cwmín 3 cwmáx 4 aifs 1 txopLimit 3264
      cwmín 3 cwmáx 4 aifs 2 txopLimit 3264
```

(a)

<sup>49</sup> Esto quiere decir que la red inalámbrica puede trabajar sin soporte para QoS, es decir, sin manejar EDCA.

## STA

```

AC_BE cwmín  5 cwmáx 10 aifs  3 txopLimit  0
      cwmín  4 cwmáx 10 aifs  3 txopLimit  0
AC_BK cwmín  5 cwmáx 10 aifs  7 txopLimit  0
      cwmín  4 cwmáx 10 aifs  7 txopLimit  0
AC_VI cwmín  4 cwmáx  5 aifs  2 txopLimit 6016
      cwmín  3 cwmáx  4 aifs  2 txopLimit 3008
AC_VO cwmín  3 cwmáx  4 aifs  2 txopLimit 3264
      cwmín  2 cwmáx  3 aifs  2 txopLimit 1504

```

(b)

Figura 4.2 Lista de valores de parámetros EDCA tanto en AP como en STA

El Apéndice E contiene el trabajo previo que fue necesario en el cual se estableció la forma de realizar el cambio de valores de los parámetros EDCA/WMM con MadWiFi en Linux, en éste se encontró (tal como lo indicaba la teoría) que es posible cambiar los valores de los parámetros de EDCA desde el AP, sin embargo, también se encontró que al intentar modificar desde el AP el AIFS de BE de la STA que por defecto está en 3 (IEEE 802.11e-Anexo D, 2005) para que tuviera un valor de diferente, (por ejemplo 6), no se obtenía el resultado esperado. El comando a utilizar de acuerdo con la sintaxis establecida en (MadWiFi, 2007) es:

```
iwpriv ath0 aifs 0 1 6
```

La cadena '0 1 6' indica que se debe cambiar al valor del AC\_BE<sup>50</sup> (0), en el equipo remoto<sup>51</sup> (1) al valor de seis<sup>52</sup> (6).

Una vez ejecutado, se esperaba obtener en una lista como la de la figura 2 (b) el valor del AIFS de la AC\_BE en 6 en la primera fila<sup>53</sup>, pero este cambio no ocurrió. De igual forma se exploraron los demás parámetros EDCA para la categoría de BE, tales como: 'CWMAX', 'CWMIN', 'TXOPLIMIT', pero no se consiguieron los cambios esperados.

Era importante para la investigación en el sentido práctico trabajar con AC\_BE porque la mayoría de tráfico que actualmente circula en una red tiene que ver con esta categoría y por tanto era con la que se quería hacer la confrontación frente a la de VoIP (AC\_VO).

Este fue un punto crítico para el proyecto, pues, de no encontrar una forma para realizar el cambio se debería realizar ajustes a las intenciones perseguidas con el mismo.

<sup>50</sup> *Best Effort*

<sup>51</sup> La STA como se tenía la expectativa dada la capacidad de la trama *Beacon* en un QBSS

<sup>52</sup> Para que tuviera una espera fija de 3 intervalos adicionales

<sup>53</sup> La primera fila corresponde a los valores de los parámetros en el equipo donde se ejecuta el comando

Así, se agotaron diferentes procesos técnicos de exploración y finalmente se encontró que si era posible cambiar los valores de los parámetros EDCA directamente en las STAs y para el caso particular de esta prueba inicial, en el PC 1, con el siguiente comando:

```
iwpriv ath0 aifs 0 0 6
```

La cadena '0 0 6' indica que se debe cambiar al valor del AC\_BE (0), en el equipo local (0) al valor de seis (6), para todos los demás parámetros y valores se procede de forma similar, de esta forma los cambios se efectuaron de forma correcta y así se pudo superar este impase que puso en riesgo el desarrollo del proyecto.

Asegurando, entonces, que en las estaciones se podía hacer un cambio local de los valores de los parámetros, para continuar con el procedimiento se hizo un monitoreo de estos valores consistente en estarlos listando de forma constante, inmediatamente después de haber realizado algún cambio. Con esto, lo que se encontró fue que al cabo de un tiempo los valores de los parámetros previamente modificados, volvían a los valores por defecto.

Una hipótesis con respecto a este comportamiento era que la trama *Beacon* al tener una transmisión constante desde el AP y como era su nueva función<sup>54</sup>, después de un determinado tiempo, le enviaba a las estaciones los valores de los parámetros que tenía configurado el AP, pero como aspecto intrigante el tiempo de difusión del *Beacon* es normalmente de 100 ms, esto teóricamente implicaba que de ser esta la razón no se tendría forma de aprovechar el cambio de valores que se había logrado en los parámetros de forma local porque prácticamente de manera instantánea volverían a estar en los valores por defecto.

Pero lo que se había encontrado es que no se estaba haciendo de forma instantánea sino que se tenía un período de tiempo en el cual los valores configurados en las STAs se mantenían. Así que esta era una "falla" afortunada pues, permitía tener un intervalo de tiempo para estimar el impacto de la variación de los parámetros en el desempeño de los tráficos en especial el de VoIP. Lo que se hizo entonces, fue proceder a tomar ese tiempo para saber de cuánto se disponía para realizar cada prueba del proyecto.

Así se decidió contabilizar dicho intervalo entre transmisiones, de la siguiente manera: se realizó un cambio de parámetro en la estación y se monitoreó constantemente los valores de los parámetros hasta observar que se retornaban a los valores iniciales<sup>55</sup>, en ese preciso instante se cambiaba el valor de un parámetro y con cronómetro se iniciaba la toma del tiempo hasta volver a

<sup>54</sup> Con el campo *QoS Parameter Set Element*

<sup>55</sup> Se realizó un *script* de Shell de Linux que repetía el comando para listar los parámetros cada segundo de tal forma que no era necesaria su introducción manual y así más confiable la toma del tiempo.

observar, en la lista que genera el comando, que el parámetro volvía a tomar su valor por defecto y en ese instante se detenía el cronómetro. Este proceso se ilustra en la Figura 4.3, en la que tx BEACON<sup>56</sup> se asumió como el momento en que las estaciones realmente tomaban la indicación de volver a sus parámetros por defecto.

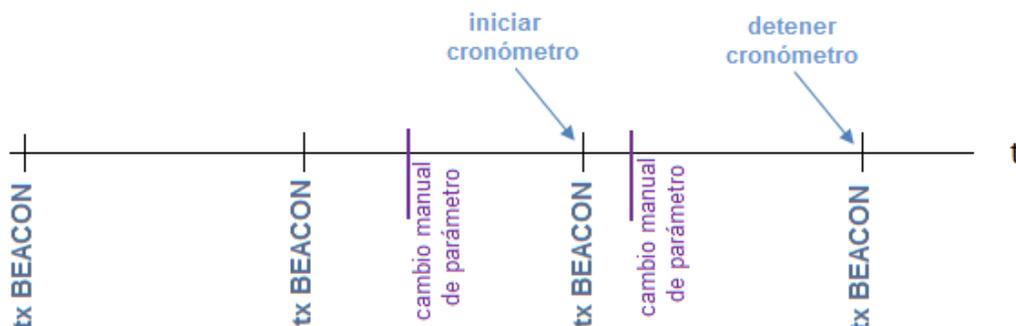


Figura 4.3 Técnica para medir tiempo entre cambio de valores de parámetros.

Con este proceso el intervalo de tiempo que se pretendía medir, es decir, el tiempo que transcurrió entre iniciar cronómetro y detener cronómetro, fue 8 minutos 40 segundos. Afortunadamente este tiempo permitía realizar las pruebas requeridas pero se tenía que iniciar el tráfico de información de una forma casi instantánea a penas se cambiaran los valores de las estaciones. Este proceso fue dispendioso porque implicaba cambiar en cada estación el valor y luego iniciar los tráficos de datos y de voz<sup>57</sup>. Afortunadamente con *scripts* de Shell y colaboración en la sala donde se llevó a cabo el proceso, fue posible, la obtención de resultados y en últimas la ejecución del procedimiento generado para encontrar valores adecuados de los parámetros EDCA en pro de los propósitos de este proyecto.

#### Nota sobre el *Beacon*:

Suponiendo que el tiempo tomado en el numeral anterior se mantendría al introducir tráfico y aumentar estaciones, se exploró la posibilidad de incrementar el tiempo entre *Beacons*, lo cual teóricamente se logra con el siguiente comando:

```
iwpriv ath0 bintval t
```

Donde  $t$  indica el tiempo en milisegundos. Como se indicó anteriormente se encontró que el valor establecido por defecto es de 100 ms, lo que no tiene concordancia con el resultado del cronómetro de 8 minutos 40 segundos en cambiar el valor de los parámetros. Así que algo fuera de lo esperado ocurría.

<sup>56</sup> Fue una forma de denominarlo basado en que el *beacon* tiene en un QBSS la responsabilidad de establecer el valor de los parámetros EDCA.

<sup>57</sup> Con lo indicado hasta aquí se espera dar una idea del esfuerzo y dedicación que fue necesario para el proceso de evaluación por medio de experimentación.

Como prueba de descarte se fijó la variable  $\tau$  en diferentes valores, pero los resultados no demostraban un comportamiento lógico, dado que al fijar este valor por encima de 100 ms, donde se esperaría que el tiempo entre el cambio de valores de parámetros aumentara, en ocasiones ocurría lo contrario, y de igual forma, al fijar  $\tau$  por debajo de 100 ms, el tiempo entre el cambio ocasionalmente, aumentaba.

Por lo tanto, se dejó quieto al *Beacon*<sup>58</sup> y se descartó para cualquier procedimiento posterior, contando con que el tiempo que permitían localmente modificar el valor de los parámetros era suficiente siempre y cuando se trabajara con rapidez.

Así mismo, dado que esto ocurría con un PC configurado para ser punto de acceso, se procedió a utilizar en su lugar un dispositivo AP Linksys™ WRT54GL con firmware DD-WRT v24<sup>59</sup> con la intención de realizar las mismas pruebas anteriormente descritas, esperando conseguir mejores resultados, pero los que se obtuvieron no cumplieron con las expectativas:

1. En el primer caso, los parámetros de la estación no seguían los cambios que se le hacían desde el AP.
2. En el segundo caso, el tiempo entre *Beacons* obtenido fue 4 minutos 42 segundos, por lo que era menor.
3. En el tercer caso, se configuraron más PC s como STAs y se cambió el valor del intervalo de transmisión del *Beacon* haciendo uso del campo *beacon interval*, a través de la interfaz de configuración que ofrece el firmware DD-WRT, con lo que se obtuvieron distintos comportamientos: a pesar que el AP indicaba un amplio rango para la configuración del *beacon interval* (0-65000ms), cuando se le introducía un valor demasiado pequeño con respecto a 100ms o un valor mayor a 3000 ms la red dejaba de funcionar, dejando a todas las estaciones aisladas.

Por lo anterior, se decidió trabajar nuevamente con el equipo PC configurado como AP (PC 7).

Así, con un PC como AP y solamente tres configurados como STAs se decidió probar si el tiempo de modificación de los parámetros asumido como realizado por tx BEACON de la Figura 4.3 se daba en el mismo instante para todas las estaciones. Para lo cual se realizó el cambio manual de parámetros como se ha indicado anteriormente y se monitorearon en conjunto, pero se encontró que el cambio no se daba en el mismo momento en todas las estaciones y no se encontró alguna relación lógica entre la diferencia de esos intervalos de tiempo de llegada –se consideró esto como una oportunidad para un proceso posterior de investigación en el entendimiento de este fenómeno– llegando hasta el punto de

<sup>58</sup> Aunque antes, como opción “desesperada” a este inconveniente, se pensó en deshabilitar la transmisión del *Beacon*, pero, resultaba en algo paradójico, pues, el *Beacon* es el latido de las redes inalámbricas, así que no se tomó como opción.

<sup>59</sup> Pues, en esta forma el dispositivo queda con la posibilidad de trabajar con WMM.

considerar que el valor de los parámetros en la práctica con este montaje no está obedeciendo a la transmisión del *Beacon*.

Así las cosas, finalmente se decidió realizar el cambio de parámetros localmente y estar alerta del momento en el que en una estación este parámetro volviera a su valor por defecto, pues, este sería el momento en el cual se detendría la ejecución de la prueba. Para esto y con el fin de observar constantemente el valor de los parámetros de cada estación (se usó el *script* que aparece en el Apéndice D) que se encargaba de listar cada segundo los parámetros de cada estación, y además porque los procesos de monitoreo realizados hasta el momento se habían hecho de forma manual lo cual era engorroso y no se contaba con el personal que se encargara de forma efectiva para cada STA<sup>60</sup>.

Ya superadas las dificultades anteriores, que pusieron en aprietos al desarrollo del proyecto, y teniendo asegurado y verificado el funcionamiento apropiado del soporte WMM/EDCA en los diferentes equipos, se procedió entonces a configurar el QBSS completo indicado por la Figura 4.1, de esta forma: El PC 7 siguió funcionando como AP, los computadores 1 a 6 quedaron como STAs y dependiendo de la situación permitirían la transferencia de Datos o la de Voz.

Para las descargas de Datos cada equipo disponía de un archivo publicado en un directorio por el servidor web Apache que se podía descargar de manera remota mediante el uso de una herramienta<sup>61</sup> de descarga simultánea ubicada en el PC 8 el cual actuó como cliente de HTTP.

Para generar los flujos de VoIP se atendió el procedimiento que aparece en (Agredo y Gaviria, 2006) en el que se instala el servidor de VoIP Asterisk en todos los equipos que cursen este tipo de tráfico, ya que Asterisk puede operar también como cliente, de modo que todo el trabajo, configuración y escritura de *scripts* se hizo para manejar las llamadas en Asterisk, en específico se aprovecha que para generar una llamada entre servidores (no sobra decir que con tráfico adecuadamente marcado a nivel 3 para que MadWiFi lo mapeara a nivel 2), se debe ubicar el archivo `sample.call` de Asterisk en `/tmp` e ingresar el siguiente comando en la consola:

```
cp /tmp/sample.call /var/spool/asterisk/outgoing/sample1.call
```

El contenido del archivo `sample.call` aparece en el Apéndice C.

Hasta este punto que buscaba adecuar y asegurar la infraestructura para la realización de las experimentaciones y que fue el relato de un espacio crítico para el proyecto, no había tráfico ya que lo que se buscaba con lo realizado era

---

<sup>60</sup> Este último *script* se ejecutó de forma remota desde un único PC que tenía acceso a todas las STAs, el tráfico adicional de este proceso al aparecer en todas las experimentaciones realizadas no modificaba el resultado final buscado.

<sup>61</sup> Para esto se utilizó el Software Getright que permite dos capacidades: iniciar descargas simultáneamente desde varios servidores y generar registros (logs) con el *throughput* promedio por descarga.

conocer y asegurar el comportamiento esperado del soporte WMM/EDCA para poder tener un laboratorio para realizar las pruebas posteriores de los escenarios descritos al inicio de este capítulo y que se tratan a continuación.

### 4.3. Primer Escenario: Tráfico Alto de Datos

El primer escenario establecido tenía un flujo de VoIP y cinco de Datos (para lo cual se trabajó con HTTP como se indicó en la sección anterior) pertenecientes a las categorías AC\_VO y AC\_BE respectivamente.

Se buscó resolver en específico la siguiente pregunta de investigación:

¿Qué valores de parámetros EDCA pueden de mejor manera<sup>62</sup> contribuir a satisfacer los requisitos de VoIP en cuanto a retardo y pérdida de paquetes cuando el nivel de tráfico del QBSS sea alto con tráfico de *best-effort*?

El procedimiento realizado para resolver esta pregunta se explica a continuación:

Una STA con flujo de VoIP UDP (RTP/G.711)<sup>63</sup> + 5 STAs de Datos TCP (HTTP)

*Hipótesis a verificar:* con el aumento del AIFS del tráfico de *Best Effort* se puede contribuir a satisfacer los requerimientos de voz, pues, si las STAs de Datos deben esperar más para tomar el medio aumenta la estadística de veces que VoIP encuentra el medio desocupado disminuyendo por tanto el retardo y la pérdida de paquetes global.

#### Pruebas a realizar:

La infraestructura utilizada y la distribución de roles entre los equipos se muestra en la Figura 4.4.

AIFS[AC\_Vo]= 2 (El de la Voz se deja en 2 - Como por defecto).

AIFS[AC\_BE]= 3 Se inicia con 3<sup>64</sup> y se aumenta hasta 7<sup>65</sup>. Una vez se le cambia el valor del parámetro AIFS a todas las estaciones de datos, se inicializa el tráfico en el QBSS y se verifica en cada uno si disminuyen el retardo y la pérdida de paquetes de voz.

A las STAs de BE se les toma el *throughput* promedio con cada una de las modificaciones al AIFS con fines de identificar el impacto sobre este indicador de desempeño.

<sup>62</sup> Con respecto a los valores por defecto de EDCA

<sup>63</sup> Flujo al que se le busca proteger modificando los valores del AIFS del AC\_BE para establecer si es posible encontrar un valor que permita cumplir los límites de QoS.

<sup>64</sup> El que viene por defecto.

<sup>65</sup> Que lo hace equivalente a *Background*

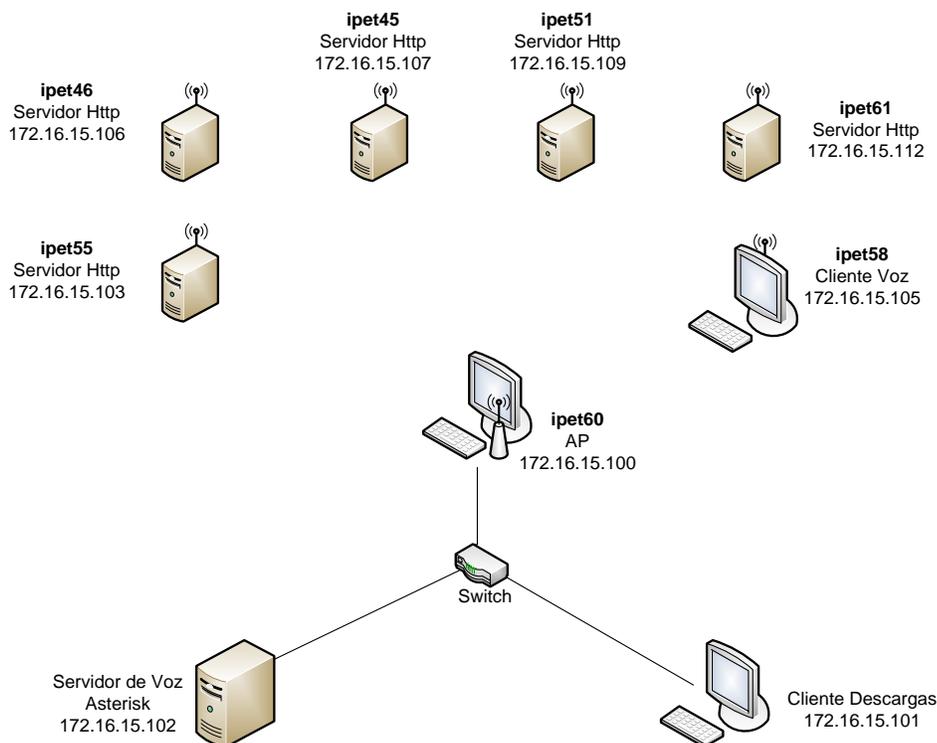


Figura 4.4 Infraestructura para un flujo de VoIP + 5 de Datos

Procedimiento:

1. Realizar el cambio de valor del parámetro AIFS de forma local en cada una de las estaciones con el comando

```
iwpriv ath0 aifs 0 0 X
```

Donde X es el valor deseado de AIFS (3-7).

2. Iniciar el monitoreo remoto de todas las estaciones incluido el AP con el *script* que lista cada segundo los valores de los parámetros de cada estación que aparece en el Apéndice D. Debe observarse permanentemente el listado de parámetros de cada estación hasta el momento en el que cambie el valor del AIFS al valor fijado en el AP, momento en el que debe concluir la prueba<sup>66</sup>.
3. Iniciar la llamada<sup>67</sup> de VoIP de una estación.
4. Iniciar las descargas simultáneas con el PC destinado para tal fin como se indica en la Figura 4.4.
5. Observar el comportamiento del tráfico de voz ejecutando<sup>68</sup> el comando:

<sup>66</sup> Sin embargo, en la práctica se escogieron archivos cuya descarga se diera antes que cambiaran los valores.

<sup>67</sup> Recordar el proceso con el archivo sample.call y Asterisk

<sup>68</sup> Desde la consola de asterisk.

```
iax2 show netstats
```

Comando<sup>69</sup> que permite tener el resultado en cuanto a retardo<sup>70</sup> y la pérdida de paquetes<sup>71</sup> con el fin de ver si se contribuía a no superar los límites. La Figura 4.5 muestra el resultado de ejecutar el comando cuando solamente se tenía la comunicación de VoIP y no había tráfico de datos.

```

Asterisk Console on 'ipet7' (pid 5782)
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel          RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-9   2    1   62   0     0     0     2
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel          RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-9   2    1   62   0     0     0     2
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel          RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-9   2    1   62   0     0     0     2
1 active IAX channel

```

Figura 4.5 Resultado de un solo flujo de VoIP

Los resultados de la Figura 4.5 muestran que el retardo (*Del*) y pérdida de paquetes (*Lost*) son respectivamente: 60 milisegundos y 0 paquetes, es decir, satisfacen los requerimientos de VoIP como era de esperarse. Es bueno tener como referencia estos valores para contrastar con los resultados generados cuando se introduce el tráfico adicional de datos de las 5 estaciones.

6. Una vez se finalicen las descargas se detiene la ejecución del comando `iax2 show netstats`. Habiendo tomado datos sobre el valor arrojado.
7. Como valor adicional revisar los resultados del impacto sobre el *throughput* promedio de las descargas, lo cual se determina de promediar los resultados que aparecen en los archivos `.log` que genera el software GetRight cada vez que finaliza una descarga

El procedimiento anterior se repite para los valores de AIFS desde 4 hasta 7 como se indicó y en cada ocasión se toman los resultados.

<sup>69</sup> Se ejecuta en el PC como AP (PC7 – ipet7) por eso debe verse la columna REMOTE

<sup>70</sup> Aparece como Del por abreviar *Delay*

<sup>71</sup> Aparece como *Lost*

Resultados:

De la Figura 4.6 a la Figura 4.10 se muestran los resultados de la variación del valor del AIFS[AC\_BE] sobre el flujo de VoIP.

```

Asterisk Console on 'ipet7' (pid 5782)
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-4  3  13  61   0    0    0    1
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-4  3  12  61   0    0    0    1
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-4  3  12  61   0    0    0    1
1 active IAX channel

```

Figura 4.6 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 3

```

Asterisk Console on 'ipet7' (pid 5782)
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
-- <IAX2/cliente5-3> Playing 'demo-instruct' (language 'en')
iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-3 261  10  64   2    0    0    3
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-3 261  10  64   2    0    0    3
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-3 261  10  64   2    0    0    3
1 active IAX channel

```

Figura 4.7 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 4

```

Asterisk Console on 'ipet7' (pid 5782)
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-4  2   6  62   0    0    0    3
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-4  2   6  62   0    0    0    3
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-4  2   7  62   0    0    0    3
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-4  2   7  62   0    0    0    3
1 active IAX channel

```

Figura 4.8 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 5

```

Asterisk Console on 'ipet7' (pid 5782)
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-10  48  9  63  0    0    0    3
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-10  48  10  63  0    0    0    3
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-10  48  10  63  0    0    0    3
1 active IAX channel

```

Figura 4.9 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 6

```

Asterisk Console on 'ipet7' (pid 5782)
Archivo  Editar  Ver  Terminal  Solapas  Ayuda
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-7  14  8  62  0    0    0    2
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-7  14  8  62  0    0    0    2
1 active IAX channel
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  % Drop  000  Kpkts
IAX2/cliente5-7  14  8  62  0    0    0    2
1 active IAX channel

```

Figura 4.10 Resultado de un solo flujo de VoIP + 5 de Datos con AIFS = 7

De la Figura 4.11 a la Figura 4.13 se muestran los gráficos que resumen los resultados de la variación del AIFS[AC\_BE] sobre el flujo de VoIP.

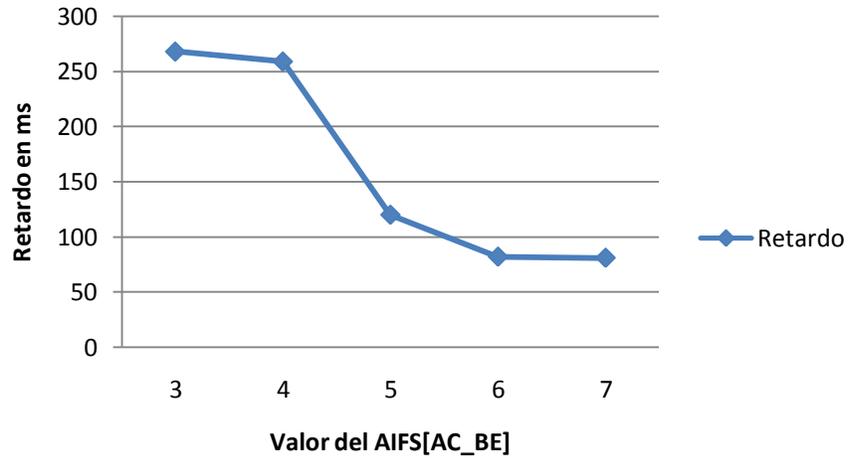


Figura 4.11 Retardo en ms del flujo de VoIP ante la variación del AIFS[AC\_BE]

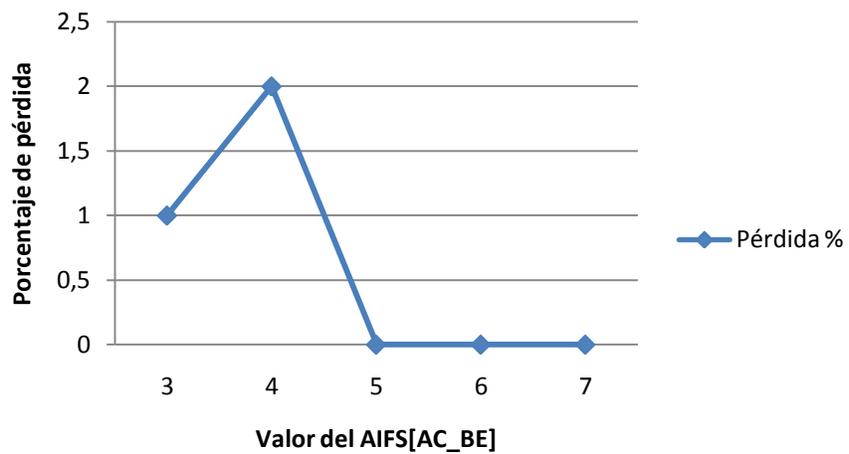


Figura 4.12 Pérdida en % del flujo de VoIP ante la variación del AIFS[AC\_BE]

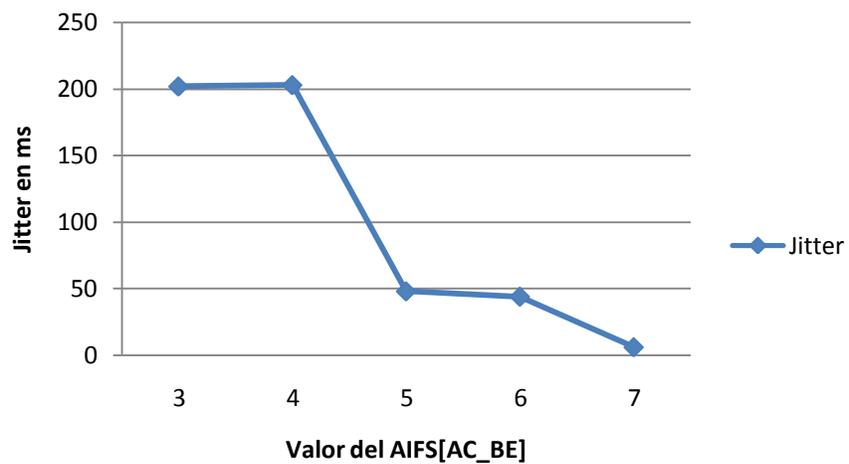


Figura 4.13 *Jitter* en ms del flujo de VoIP ante la variación del AIFS[AC\_BE]

Con el ánimo de identificar el impacto la velocidad promedio general de los flujos de datos, se tomó el valor del *throughput* de cada descarga a partir del *log* generado por la aplicación Getright los cuales se resumen en la Tabla 4.1

AIFS[AC_BE]	ipet45	ipet46	ipet51	ipet55	ipet61	Throughput Promedio (Kbytes/sg)
3	152,8	175,2	172,9	160,2	158,3	163,9
4	152,8	177,5	156,4	164,2	154,6	161,1
5	175,2	152,8	158,3	156,4	158,3	160,2
6	156,4	170,9	160,2	158,3	158,3	160,8
7	156,4	160,2	156,4	164,2	157	158,8

Tabla 4.1. Velocidad promedio<sup>72</sup> global de las descargas al variar el AIFS[AC\_BE]

#### Análisis de Resultados:

Según los resultados obtenidos es contundente que para este primer escenario planteado seleccionar un valor de 5 en el AIFS[AC\_BE] es suficiente para que se pueda contribuir a satisfacer los requerimientos de VoIP. Se evidencia que al aumentar el valor por encima de 5 disminuye aún más el retardo y el *jitter*, pero prácticamente no justifica realizarlo más cuando lo hace a costas de disminuir el desempeño global de los flujos de datos.

Por tanto, el procedimiento permite establecer que es posible proteger al flujo de VoIP con la hipótesis de aumentar el AIFS[AC\_BE] de las estaciones de datos y que el valor óptimo para el escenario planteado es 5.

## 4.4. Segundo Escenario: Tráfico Alto de VoIP

El segundo escenario establecido tenía cinco flujos de VoIP y uno de Datos pertenecientes a las categorías AC\_VO y AC\_BE respectivamente.

Se buscó resolver en específico la siguiente pregunta de investigación:

¿Qué valores de parámetros EDCA pueden de mejor manera<sup>73</sup> contribuir a satisfacer los requisitos de VoIP en cuanto a retardo y pérdida de paquetes<sup>74</sup> cuando el nivel de tráfico del QBSS sea alto con tráfico de VoIP?

El procedimiento realizado para resolver esta pregunta se explica a continuación:

Cinco STAs con flujo de VoIP UDP (RTP/G.711)<sup>75</sup> + 5 STAs de Datos TCP (HTTP)

<sup>72</sup> En Kbytes/segundo

<sup>73</sup> Con respecto a los valores por defecto de EDCA

<sup>74</sup> Relacionando la pérdida de paquetes con una mayor o menor cantidad de colisiones.

**Hipótesis a verificar:** Si existe alto tráfico de VoIP aumentar el  $CW_{max}[AC\_VO]$  al  $CW_{max}[AC\_VI]$  para disminuir la probabilidad de colisión, pues si se tiene un espacio mayor para la espera aleatoria se disminuyen las colisiones de VoIP.

**Pruebas a realizar:**

La infraestructura utilizada y la distribución de roles entre los equipos se muestra en la Figura 4.14.

$CW_{max}[AC\_VO]$  como lo trae MadWiFi por defecto es 4, se cambia a 15 y luego a 31 para ver el impacto en la variación de este parámetro.

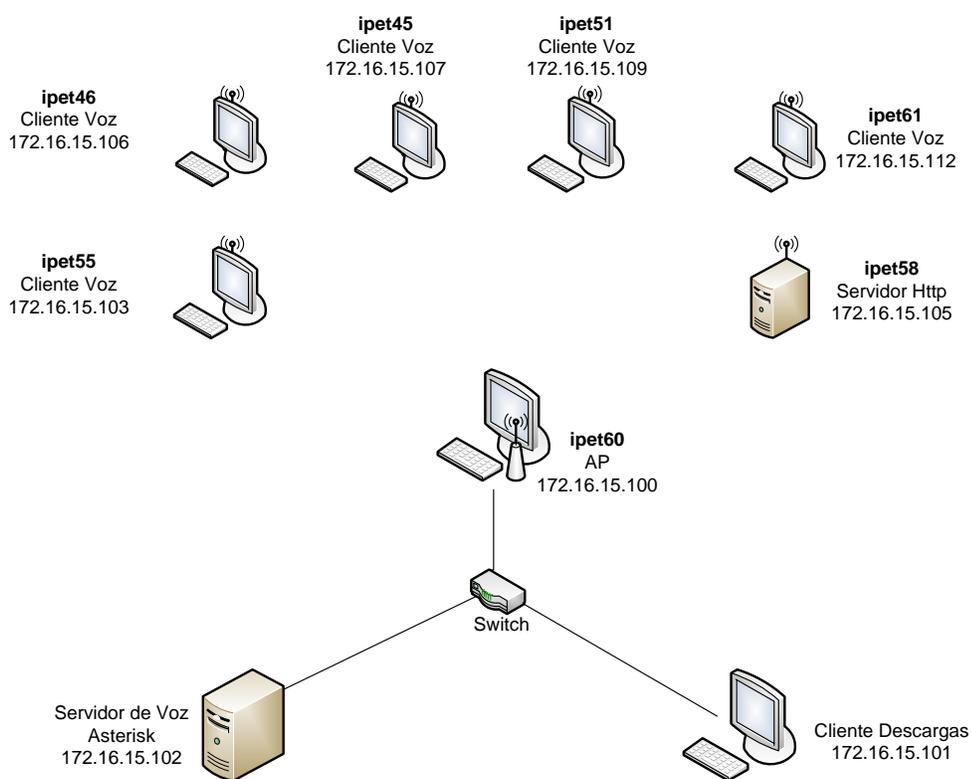


Figura 4.14 Infraestructura para 5 flujos de VoIP + 1 de Datos

**Procedimiento:**

1. Realizar el cambio de valor del parámetro  $CW_{max}$  de las estaciones desde el AP<sup>75</sup> con el comando:

<sup>75</sup> En este caso se buscaba proteger de forma global estos flujos, era suficiente con que uno no cumpliera los límites de QoS.

<sup>76</sup> Debe recordarse que a la única categoría que no fue posible cambiarle los valores desde el AP fue al *Best Effort*, como en este caso se trabaja sobre valores del  $CW_{min}[AC\_VO]$  se podía cambiar en el AP y apenas iniciaban los flujos de VoIP este valor se veía reflejado en las estaciones.

```
iwpriv ath0 cwmax 3 1 X
```

Donde X es el valor deseado de CWmax (4, 15, 31)<sup>77</sup>.

2. Iniciar el monitoreo remoto de todas las estaciones incluido el AP con el *script* que lista cada segundo los valores de los parámetros de cada estación que aparece en el Apéndice D.
3. Iniciar las 5 llamadas. Se verifica que en el listado de parámetros de cada estación el valor del CWmax cambia al valor fijado en el AP (numeral 1).
4. En el cliente de descargas de la Figura 4.14, iniciar la descarga por medio del gestor GetRight.
5. Observar el comportamiento del tráfico de voz ejecutando repetidamente el siguiente comando en el Servidor de Voz<sup>78</sup>:

```
iax2 show netstats
```

6. Una vez finalice la descarga se detiene la ejecución del comando `iax2 show netstats`. Habiendo tomado datos sobre el valor arrojado.
7. Como adicional se revisa el resultados del impacto sobre el *throughput* de la descarga, lo cual aparece en el archivo `.log` que genera el software GetRight cada vez que finaliza una descarga.

El procedimiento anterior se repite para los valores de CWmax[AC\_VO] con 15 y 31.

### Resultados:

De la Figura 4.15 a la Figura 4.17 se muestran los resultados de la variación del valor del CWmax[AC\_VO] sobre el flujo de VoIP.

<sup>77</sup> Teóricamente el valor por defecto de la ventana de contienda inicial máxima para la [AC\_VO] es de 15 para 802.11b, pero el driver utilizado en el proyecto arrojaba 4 como valor por defecto, por tanto, se partió desde este valor.

<sup>78</sup> Mayor explicación en el procedimiento del primer escenario.

Asterisk Console on 'ipet7' (pid 5782)

```

Archivo  Editar  Ver  Terminal  Solapas  Ayuda
Channel  RTT  Jit  Del  Lost  %  Drop  000  Kpkts  Jit  Del  Lost  %  Drop  000  Kpkts
IAX2/cliente7-1  88  249  307  10  1  13  53  4  7  61  6  0  0  0  3
IAX2/cliente6-7  19  225  282  9  1  11  69  3  6  63  8  0  0  0  3
IAX2/cliente3-15  335  326  394  8  1  14  49  3  6  64  10  0  0  1  3
IAX2/cliente9-20  10  250  330  7  0  11  54  4  6  61  5  0  0  0  3
IAX2/cliente12-21  18  254  320  11  0  13  54  3  7  62  7  0  0  3  3
5 active IAX channels
*CLI> iax2 show netstats
----- LOCAL -----
Channel  RTT  Jit  Del  Lost  %  Drop  000  Kpkts  Jit  Del  Lost  %  Drop  000  Kpkts
IAX2/cliente7-1  88  233  308  10  1  13  53  4  7  61  6  0  0  0  3
IAX2/cliente6-7  19  234  283  9  1  11  69  3  6  63  8  0  0  0  3
IAX2/cliente3-15  335  326  394  8  1  14  49  3  6  64  10  0  0  1  3
IAX2/cliente9-20  10  250  330  7  0  11  54  4  6  61  5  0  0  0  3
IAX2/cliente12-21  18  254  320  11  0  13  54  4  7  62  7  0  0  3  3
5 active IAX channels
*CLI> iax2 show netstats
----- LOCAL -----
Channel  RTT  Jit  Del  Lost  %  Drop  000  Kpkts  Jit  Del  Lost  %  Drop  000  Kpkts
IAX2/cliente7-1  88  213  288  10  1  14  53  4  7  61  6  0  0  0  3
IAX2/cliente6-7  19  233  302  9  1  11  69  3  6  63  8  0  0  0  3
IAX2/cliente3-15  335  343  394  8  1  14  49  3  6  64  10  0  0  1  3
IAX2/cliente9-20  10  250  330  7  0  11  54  4  6  61  5  0  0  0  3
IAX2/cliente12-21  18  238  300  11  0  14  54  4  7  62  7  0  0  3  3
5 active IAX channels

```

Figura 4.15 Resultado de cinco flujos de VoIP con CWmax = 4 + 1 de Datos

Asterisk Console on 'ipet7' (pid 5782)

```

Archivo  Editar  Ver  Terminal  Solapas  Ayuda
Channel  RTT  Jit  Del  Lost  %  Drop  000  Kpkts  Jit  Del  Lost  %  Drop  000  Kpkts
IAX2/cliente6-3  2  9  63  0  0  0  0  3  6  60  1  0  0  0  3
IAX2/cliente3-15  12  5  63  1  0  0  0  3  5  62  0  0  0  0  3
IAX2/cliente12-16  12  11  67  11  0  0  0  3  6  64  12  1  0  0  3
IAX2/cliente9-17  2  7  59  0  0  0  0  3  6  63  0  0  0  0  3
IAX2/cliente7-19  8  7  64  1  0  0  0  3  7  60  0  0  0  0  3
5 active IAX channels
*CLI> iax2 show netstats
----- LOCAL -----
Channel  RTT  Jit  Del  Lost  %  Drop  000  Kpkts  Jit  Del  Lost  %  Drop  000  Kpkts
IAX2/cliente6-3  2  9  63  0  0  0  0  3  6  60  1  0  0  0  3
IAX2/cliente3-15  12  5  63  1  0  0  0  3  5  62  0  0  0  0  3
IAX2/cliente12-16  12  10  67  11  0  0  0  3  6  64  12  1  0  0  3
IAX2/cliente9-17  2  7  59  0  0  0  0  3  6  63  0  0  0  0  3
IAX2/cliente7-19  8  8  65  1  0  0  0  3  7  60  0  0  0  0  3
5 active IAX channels
*CLI> iax2 show netstats
----- LOCAL -----
Channel  RTT  Jit  Del  Lost  %  Drop  000  Kpkts  Jit  Del  Lost  %  Drop  000  Kpkts
IAX2/cliente6-3  2  9  63  0  0  0  0  3  6  60  1  0  0  0  3
IAX2/cliente3-15  12  6  64  1  0  0  0  3  5  62  0  0  0  0  3
IAX2/cliente12-16  12  10  67  11  0  0  0  3  6  64  12  1  0  0  3
IAX2/cliente9-17  2  7  59  0  0  0  0  3  6  63  0  0  0  0  3
IAX2/cliente7-19  8  8  65  1  0  0  0  3  7  60  0  0  0  0  3
5 active IAX channels

```

Figura 4.16 Resultado de cinco flujos de VoIP con CWmax = 15 + 1 de Datos

```

Asterisk Console on 'ipet7' (pid 5782)
Archivo Editar Ver Terminal Solapas Ayuda
Channel      RTT  Jit  Del  Lost  %  Drop  000  Kpkts  Jit  Del  Lost  %  Drop  000  Kpkts
IAX2/cliente6-7  1000  31  84  0  0  0  0  0  0  0  0  0  0  0  0
IAX2/cliente3-15  1000  33  81  0  0  0  0  0  0  0  0  0  0  0  0
IAX2/cliente12-16  1000  34  81  0  0  0  0  0  0  0  0  0  0  0  0
IAX2/cliente9-17  1000  36  85  0  0  0  1  0  0  0  0  0  0  0  0
IAX2/cliente7-18  1000  36  81  0  0  0  0  0  0  0  0  0  0  0  0
5 active IAX channels
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  %  Drop  000  Kpkts  Jit  Del  Lost  %  Drop  000  Kpkts
IAX2/cliente6-7  1000  31  84  0  0  0  0  0  0  0  0  0  0  0  0
IAX2/cliente3-15  1000  33  81  0  0  0  0  0  0  0  0  0  0  0  0
IAX2/cliente12-16  1000  33  81  0  0  0  0  0  0  0  0  0  0  0  0
IAX2/cliente9-17  1000  35  85  0  0  0  1  0  0  0  0  0  0  0  0
IAX2/cliente7-18  1000  35  81  0  0  0  0  0  0  0  0  0  0  0  0
5 active IAX channels
*CLI> iax2 show netstats
----- LOCAL -----
Channel      RTT  Jit  Del  Lost  %  Drop  000  Kpkts  Jit  Del  Lost  %  Drop  000  Kpkts
IAX2/cliente6-7  1000  34  84  0  0  0  0  0  0  0  0  0  0  0  0
IAX2/cliente3-15  1000  35  81  0  0  0  0  0  0  0  0  0  0  0  0
IAX2/cliente12-16  1000  33  81  0  0  0  0  0  0  0  0  0  0  0  0
IAX2/cliente9-17  1000  35  85  0  0  0  1  0  0  0  0  0  0  0  0
IAX2/cliente7-18  1000  35  81  0  0  0  0  0  0  0  0  0  0  0  0
5 active IAX channels

```

Figura 4.17 Resultado de cinco flujos de VoIP con CWmax = 31 + 1 de Datos

Con el ánimo de identificar el impacto sobre la velocidad del flujo de datos, se tomó el valor del *throughput* para la descarga a partir del *log* generado por la aplicación Getright con cada variación de la CWmax, que se resume en la Tabla 4.2.

CWmax[AC_VO]	ipet58 (Kbytes/sg)
4	525,5
15	571,2
31	597,2

Tabla 4.2. Velocidad promedio<sup>79</sup> de la descarga al variar el CWmax[AC\_VO]

#### Análisis de Resultados:

En este caso, el retardo y el porcentaje de pérdida de los paquetes que pueden tomar el medio siempre están por debajo de los límites establecidos para VoIP, pero se nota una variación en la columna `Lost` que como aspecto curioso no genera variación en la de % de pérdida, pero aún así se evidencia una disminución a la pérdida de algunos paquetes que es posible que estén asociados a pérdidas por colisión.

En cuanto a las descargas, se identificó una mejoría en el *throughput* a medida que se incrementaba el valor de CWmax.

<sup>79</sup> En Kbytes/segundo

Por tanto, si la columna `Lost` indica pérdida por colisiones, las variaciones del `CWmax` para el escenario establecido evidencian una disminución de las mismas, pero no se hace evidente la optimización para cumplir los límites de retardo y pérdida de paquetes en una red tan pequeña en cuanto a tráfico de voz, quedando la expectativa de que el cambio de este parámetro sí afecte positivamente el comportamiento de una red con una cantidad elevada de estaciones de voz o cuando entra en juego tráfico de video marcado para ser diferenciado como [AC\_VI] .

Dado que a diferencia del anterior escenario y del siguiente, donde el procedimiento realizado por experimentación es suficiente para identificar el impacto de la variación de parámetros, en este segundo escenario, se plantea la necesidad de contar con una mayor cantidad de equipos, por lo que se consideró necesario realizar un proceso de simulación. Por este motivo a continuación se explica el procedimiento y los resultados obtenidos.

#### **4.3.1. Selección por Simulación**

Una vez entendida la necesidad de contar con un mayor número de equipos, dado que no fue suficiente para un proceso de experimentación evidenciar el impacto de la modificación de la ventana de contienda máxima (`CWmax`) de la categoría de Voz para disminuir el número de colisiones, y basado en los resultados de (Agredo y Gaviria, 2006) en los que se obtuvo que con 12 estaciones que generaran VoIP se tenía problemas para el cumplimiento de los límites de QoS, se estableció en el Simulador NCTUns 4.0 una topología con 14 estaciones que generarían 14 flujos de VoIP como la que aparece en la Figura 4.18.

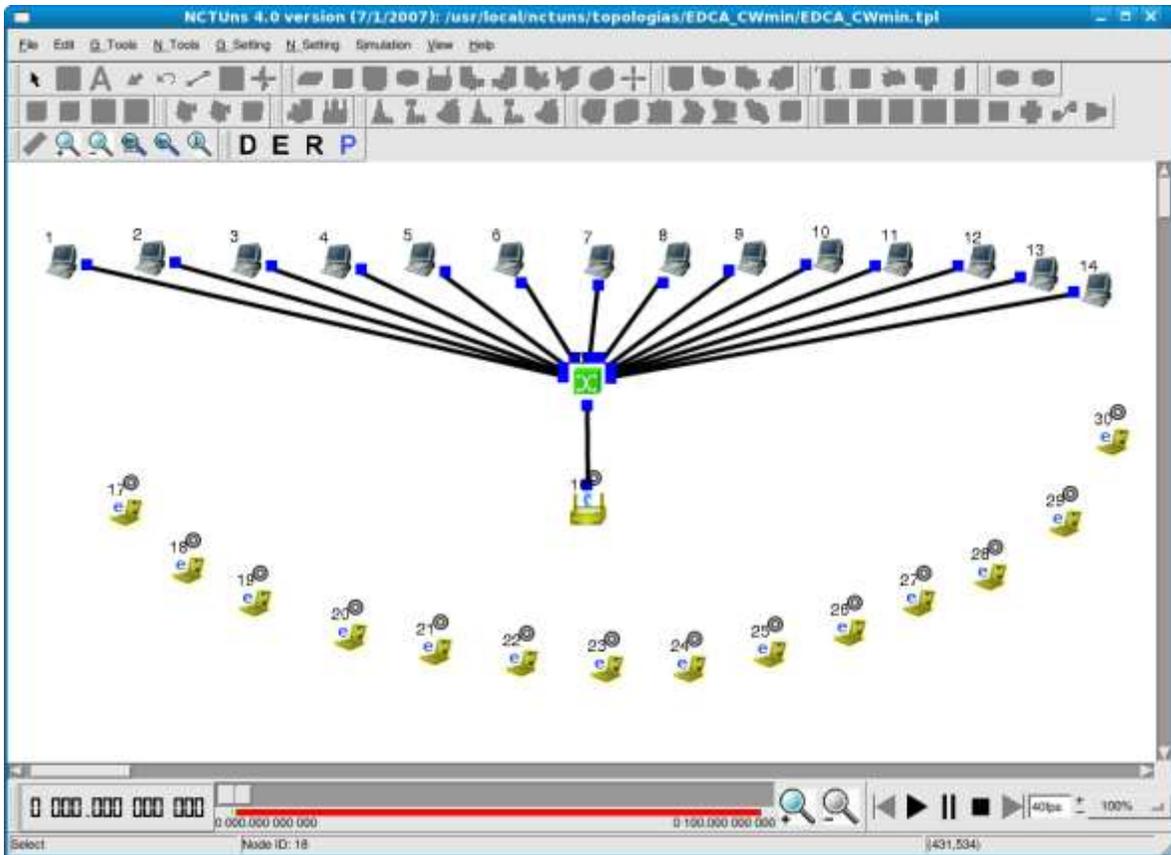


Figura 4.18 Topología de simulación para la variación del  $CW_{max}[AC\_VO]$

En este caso, se partió de la hipótesis que para la categoría de voz al tener una  $CW_{max}$  baja<sup>80</sup> en un escenario con un elevando número de estaciones con flujos de VoIP era muy probable que se presentara una gran cantidad de colisiones.

Precisamente la necesidad de contar con una gran cantidad de estaciones fue lo que impidió obtener resultados prácticos con la experimentación ante la imposibilidad de contar con un la cantidad suficiente de equipos para tal fin, pero para este tipo de situaciones es que este proyecto tenía definido utilizar la simulación.

### Procedimiento

En el simulador se configuró que las estaciones inalámbricas registraran la cantidad de colisiones que se presentaban en cada ejecución de simulación con un  $CW_{max} [AC\_VO]$  diferente.

Dada la relación entre los parámetros  $CW_{max}$  y  $CW_{min}$  donde para las categorías de Voz y Video el  $CW_{max}$  es función del  $CW_{min}$  como puede revisarse en

<sup>80</sup> 7 en el primero intento y luego 15 en el segundo. En experimentación se trabajo inclusive con 3 porque el MadWiFi traía este valor como inicial.

(Alianza WiFi WMM, 2005) y como así mismo lo maneja el Simulador<sup>81</sup>, lo que debió cambiarse fue el valor del CWmin en el archivo de variables respectivo del código fuente del motor de simulación. Este proceso para modificar los valores de los parámetros EDCA en el Simulador NCTUns 4.0 se explica en el Apéndice G en el que como se verá se tuvo la necesidad de invertir una buena cantidad de tiempo para entender el funcionamiento, identificar los archivos fuentes y verificar que las modificaciones eran tomadas.

En cada una de las 14 estaciones inalámbricas se activó el registro de Colisiones, archivos que se generaban con el nombre EDCA\_CWmax.80211\_NX\_P1\_Call.log donde la X identificaba al número de la estación a la que se le iba a hacer el monitoreo

Dada la intención de verificar si se presentaba una disminución en el número de colisiones con el aumento del CWmax se guardó el archivo de registro de una estación piloto, para cada una de las veces que fue recompilado y reinstalado el motor de simulación del NCTUns 4.0

### Resultados:

De la Figura 4.19 a la Figura 4.21 se muestran el número de colisiones de los flujos de VoIP frente a la variación del valor del CWmax[AC\_VO] a partir de la evidencia de los archivos de registro que genera el simulador.

---

<sup>81</sup> A diferencia de MadWiFi en el que puede cambiarse directamente el valor con un comando

Time	Count
1.000	0
2.000	106
3.000	184
4.000	214
5.000	326
6.000	338
7.000	332
8.000	414
9.000	456
10.000	452
11.000	444
12.000	445
13.000	452
14.000	459
15.000	440
16.000	452
17.000	442
18.000	440
19.000	466
20.000	446
21.000	454
22.000	448
23.000	440
24.000	439
25.000	450
26.000	452
27.000	440
28.000	450
29.000	436

Figura 4.19 Archivo de registro de colisiones para  $CW_{max}[AC\_VO] = 4$

Time	Count
1.000	0
2.000	8
3.000	0
4.000	18
5.000	0
6.000	16
7.000	24
8.000	0
9.000	20
10.000	0
11.000	34
12.000	20
13.000	0
14.000	34
15.000	20
16.000	0
17.000	33
18.000	0
19.000	20
20.000	30
21.000	0
22.000	22
23.000	0
24.000	32
25.000	22
26.000	0
27.000	30
28.000	0
29.000	22

Figura 4.20 Archivo de registro de colisiones para  $CW_{max}[AC\_VO] = 15$

Time	Collisions
1.000	0
2.000	4
3.000	0
4.000	12
5.000	0
6.000	8
7.000	12
8.000	0
9.000	8
10.000	0
11.000	14
12.000	8
13.000	0
14.000	14
15.000	8
16.000	0
17.000	12
18.000	0
19.000	8
20.000	12
21.000	0
22.000	8
23.000	0
24.000	14
25.000	8
26.000	0
27.000	12
28.000	0
29.000	8

Figura 4.21 Archivo de registro de colisiones para  $CW_{max}[AC\_VO] = 31$

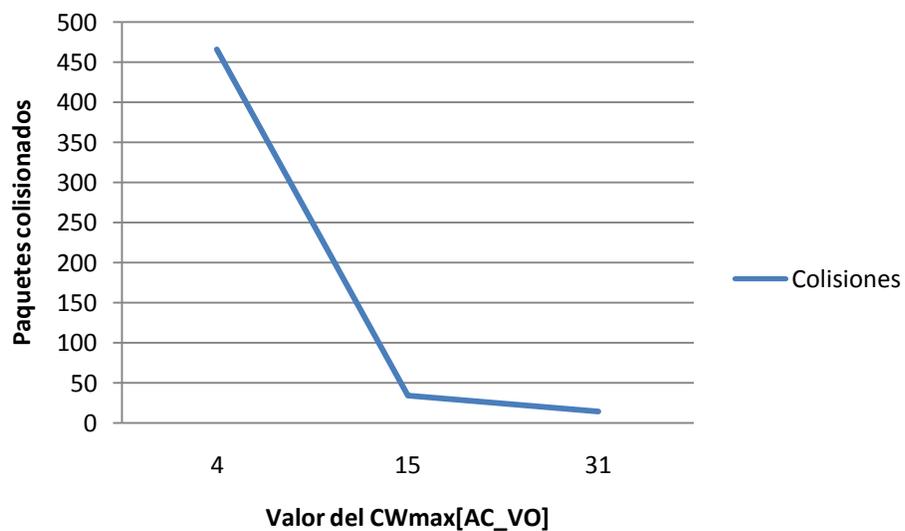


Figura 4.22 Colisiones de flujos de VoIP ante la variación del  $CW_{max}[AC\_VO]$

### Análisis de Resultados:

Se puede apreciar en la Figura 4.22 que en la medida que el CWmax [AC\_VO] se va aumentando, iniciando en un valor de 4<sup>82</sup>, luego pasando a un valor de 15<sup>83</sup> y finalizando en un valor de 31<sup>84</sup> se tiene una disminución notoria en los valores indicativos de la cantidad de paquetes colisionados.

La información generada por el simulador muestra una disminución en la cantidad de paquetes que colisionan, sin embargo, como era de esperarse al ser una red 802.11b con 14 estaciones generando tráfico no se puede esperar que no se vayan a presentar, pero lo que si sucede es que se disminuyen y por ende sus efectos, en especial sobre la pérdida de paquetes que es directamente proporcional a las colisiones.

Por tanto, gracias a la Simulación se puede tener un referente del beneficio que trae a una QBSS con alto tráfico de VoIP aumentar el valor del CWmax de Voz al del Video.

## 4.5. Tercer Escenario: Tráfico Nulo de VoIP

El tercer escenario establecido tenía seis flujos de Datos pertenecientes a la categoría AC\_BE.

Se buscó resolver en específico la siguiente pregunta de investigación:

¿Qué valores de parámetros EDCA pueden de mejor manera<sup>85</sup> contribuir a facilitar el logro de *throughputs* más altos en el tráfico de mejor esfuerzo cuando el nivel de tráfico de VoIP sea nulo?

El procedimiento realizado para resolver esta pregunta se explica a continuación:

Seis STAs con flujo de Datos TCP (HTTP)

*Hipótesis a verificar:* con la disminución del AIFS[AC\_BE] que por defecto está en 3 pasándolo a 2, se logra una mejora en la velocidad efectiva que pueden desarrollar las estaciones de datos.

En otras palabras, si en un QBSS no hay tráfico de VoIP, las estaciones de AC\_BE no deberían mantener la espera fija de AIFS=3 sino disminuirla a 2, esto debe aumentar el *Throughput*.

<sup>82</sup> Pues, así lo trae configurado por defecto MadWiFi y se quiso explorar este valor.

<sup>83</sup> Que es el valor del CWmax por defecto para la [AC\_VO]

<sup>84</sup> Que es el valor del CWmax por defecto para la [AC\_VI] que era al que se quería pasar

<sup>85</sup> Con respecto a los valores por defecto de EDCA

Como corolario, se considera que se mejore el desempeño del tráfico de *Background* si se le baja la espera fija del AIFS=7 que tiene por defecto.

Pruebas a realizar:

La infraestructura utilizada y la distribución de roles entre los equipos se muestra en la Figura 4.23.

Se trata entonces de hacer una experimentación con las 6 STAs enviando datos con el AIFS[AC\_BE]=3 que es el valor por defecto y luego hacer lo mismo pero con el AIFS[AC\_BE]=2 bajando un slot para que no mantengan la espera por defecto. Como adicional fijar el AIFS[AC\_BE]=7 para establecer la ganancia que puede recibir el tráfico de *Background* al procederse de esta forma.

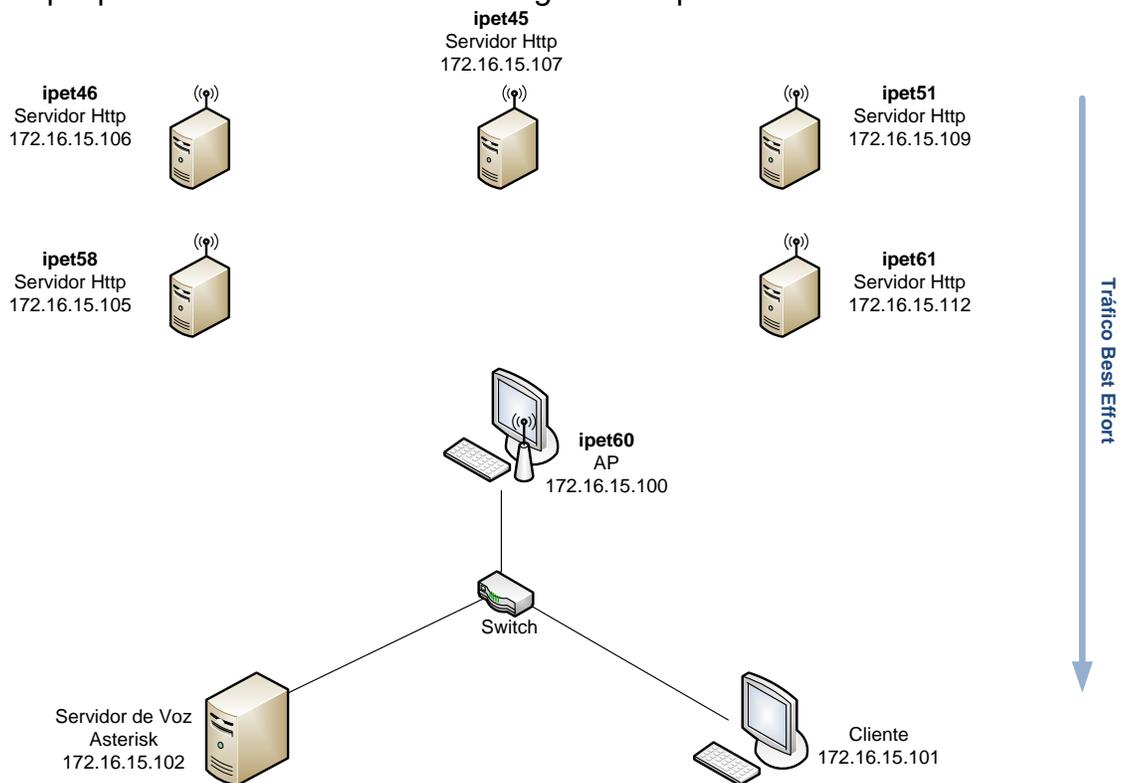


Figura 4.23 Infraestructura para únicamente flujos de Datos

Procedimiento:

1. Realizar el cambio de valor del parámetro AIFS de forma local en cada una de las estaciones con el comando

```
iwpriv ath0 aifs 0 0 X
```

Donde X es el valor deseado de AIFS (3, 2, 7).

2. Iniciar el monitoreo remoto de todas las estaciones incluido el AP con el *script* que lista cada segundo los valores de los parámetros de cada estación que aparece en el Apéndice D. Debe observarse permanentemente el listado de parámetros de cada estación hasta el momento en el que cambie el valor del AIFS al valor fijado en el AP, momento en el que debe concluir la prueba.
3. En el cliente de descargas iniciar todas las descargas simultáneamente por medio del gestor GetRight.
4. Revisar el resultados del impacto sobre el *throughput* de las descargas, que aparece en el archivo .log que genera el software GetRight cada vez que finaliza una descarga.

El procedimiento anterior se repite para los valores de AIFS[2, 3 y 7].

### Resultados:

La Tabla 4.3 resume los resultados de la variación del valor del AIFS[AC\_BE].

AIFS[AC_BE]	ipet45	ipet46	ipet51	ipet55	ipet58	ipet61	Throughput Promedio (Kbytes/sg)
2	182,5	262,8	182,5	177,5	420,2	175,2	233,5
3	176,8	187,5	172,9	172,9	279,5	175,2	197,9
7	172,9	193,2	174,2	156,4	198,5	172,9	179,0

Tabla 4.3. Velocidad promedio<sup>86</sup> global de las descargas al variar el AIFS[AC\_BE]

Una representación gráfica de los datos de la tabla anterior, se encuentran en la Figura 4.24 que permite una conceptualización visual de los resultados.

<sup>86</sup> En Kbytes/segundo

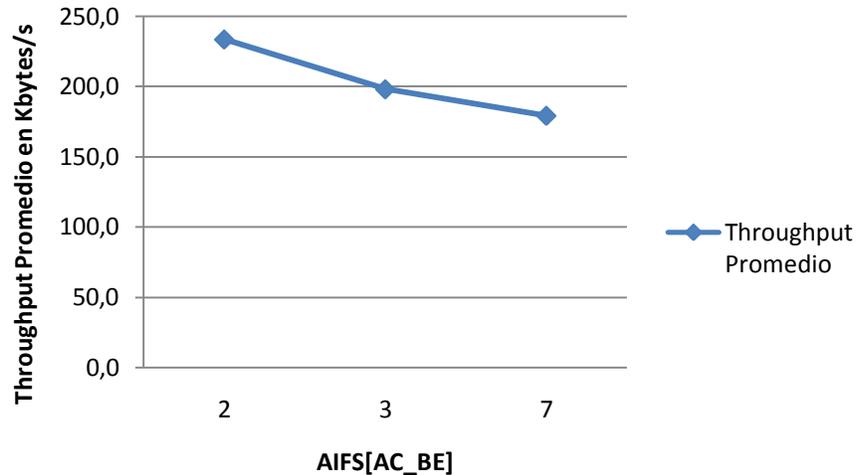


Figura 4.24 Throughput promedio al variar el AIFS[AC\_BE] sin tráfico de VoIP

Análisis de Resultados:

Tal como se esperaba, con tráfico de VoIP nulo, no es necesario que el tráfico de *Best Effort* deba esperar un *slot* adicional siendo justificable disminuirlo para que quede igual al valor del AIFS[AC\_VO] con lo que se puede lograr un mejor desempeño general.

Como adicional, se revisó el desempeño cuando el AIFS de *Best Effort* iguala al AIFS de *Background* encontrando por aplicación inversa que el *Background* también debería bajar su valor de AIFS cuando no se tenga tráfico en la red de las otras tres categorías de acceso.

Por tanto se considera que la red debería estar en capacidad de identificar el tipo de tráfico que cursa y en caso de no existir categoría o categorías de acceso de mayor prioridad bajar la espera fija establecida por el AIFS de las de menor prioridad con la intención de lograr un mejor desempeño.



# Conclusiones y Recomendaciones

El desarrollo de este proyecto de Tesis de Maestría ha permitido concluir que:

En un QBSS que deba manejar tráfico de VoIP es muy importante tener habilitado el soporte para WMM/EDCA. La evaluación realizada muestra que el mecanismo es muy eficiente la mayoría de ocasiones y por tanto estadísticamente genera un acierto para el logro de un mejor desempeño en las comunicaciones de voz.

WMM/EDCA sí beneficia al tráfico de mayor prioridad y por tanto ofrece un mejor desempeño a las aplicaciones de tiempo real como VoIP, sin embargo, se evidencia la necesidad de realizarse un ajuste a los valores de los parámetros para contribuir a satisfacer los requerimientos de QoS.

Es posible obtener mejores desempeños de flujos de VoIP con la modificación y selección de valores de parámetros EDCA, por tanto, se justifica el desarrollo de un proceso para que se haga de forma automática.

Las tecnologías o mecanismos de calidad de servicio que se soportan en marcación y clasificación, requieren que lo primero esté funcionando efectivamente, pues de no ser así, no es efectiva la diferenciación y el tratamiento del flujo de la manera correspondiente.

Modificar el AIFS ofrece una forma constante y confiable de inducir a procesos de optimización para poder lograr que se cumplan los requisitos de VoIP.

Los parámetros CWmin y CWmax, no presentan o facilitan un proceso tan eficiente como el encontrado con el AIFS y requieren de altos volúmenes de tráfico para que se obtenga incidencia en el desempeño.

En redes inalámbricas con calidad de servicio, es conveniente que todos los dispositivos tengan habilitada la calidad, pues, de no serlo, un equipo que no trabaje con el estándar queda al mismo nivel de la categoría de acceso más alta de 802.11e.

El proceso de experimentación aunque demandó un mayor esfuerzo para la puesta en funcionamiento, propició un contacto con la realidad, presentando lo que efectivamente sucedería en una implementación específica.

El proceso de Investigación demanda tiempo, dedicación, concentración y esfuerzo para la obtención de resultados, en especial cuando se quiere que éstos sean un referente para procesos posteriores.

No solamente se pueden publicar resultados finales de un proyecto de investigación, es muy conveniente publicar resultados parciales, que pueden ser de interés para la comunidad o generar un espacio para recibir aportes.

*Se recomienda para trabajos futuros:*

Generar un algoritmo para cambio dinámico del AIFS de acuerdo a las condiciones de tráfico del QBSS, validar por medio de simulación.

Generar una implementación del algoritmo anterior para que pueda ser puesto en funcionamiento en equipos tales como los WRT54GL de Linksys que permiten agregar desarrollos propios a su funcionamiento, gracias al firmware GPL que puede ser instalado en este tipo de dispositivos.

Realizar una evaluación del Protocolo de Enlace Directo, para establecer el impacto de su funcionamiento y las implicaciones.

Realizar una evaluación del uso del Parámetro TXOP en especial cuando deben enviarse varios flujos de Voz y Video por un solo canal, como cuando se tienen enlaces LAN a LAN o cuando son recogidos por el AP para comunicarlos a la red inalámbrica.

Con respecto al driver MadWiFi verificar el funcionamiento del cambio de valores de parámetros para identificar qué produce que se regresen a los valores por defecto luego de transcurrido un tiempo y que este cambio no se produzca de forma simultánea en las estaciones.

## A. Acceso Controlado al Canal HCF (HCCA)

Como se ha mencionado, 802.11e propone una nueva función llamada HCCA la cual se utiliza para proveer calidad de servicio a las estaciones de un QBSS en modo infraestructura y su principal función es la de proveer oportunidades de transmisión (TXOP) a las QSTA de acuerdo a sus necesidades de tráfico. HCCA hace uso de un punto coordinador llamado Coordinador Híbrido (*Hybrid Coordinator*, HC) el cual se coloca por defecto en el AP del QBSS. A diferencia de PCF en el estándar IEEE 802.11, HCCA opera durante los períodos CFP y CP. Durante el CFP, las estaciones no pueden competir por el medio ya que su NAV no está activo y por lo tanto el HC tiene la ventaja de acceder al medio libremente. Durante el CP, el HC también puede tener acceso al medio cuando detecte que está libre, esto lo hace utilizando la prioridad EDCA más alta (AIFS= PIFS, CWmin = CWmax = 0) ya que el mínimo tiempo de escucha para acceder al medio del resto de estaciones es el DIFS, el cual es mayor que el PIFS.

El Coordinador Híbrido asigna oportunidades de transmisión a las estaciones, mediante un mecanismo de sondeo mejorado, las cuales pueden ser asignadas a intervalos que se adaptan a la tasa de transferencia y requerimientos de retardo de flujos de tráfico en particular. Para esto el HC de cada QBSS necesita tener un amplio conocimiento de la cantidad de tráfico acumulado perteneciente a las diferentes categorías de tráfico (con diferentes prioridades) y así poder asignar una TXOP adecuada. Esto aumenta la complejidad.

El mecanismo mediante el cual el HC conoce los requerimientos de las QSTAs es el siguiente: El HC es informado sobre la cantidad de tráfico acumulado relacionado con una categoría específica mediante un proceso llamado *piggybacking*<sup>87</sup>, el cual consiste en enviar tal información en la misma trama de datos que las QSTAs envían al HC. Cada trama de datos contiene un nuevo campo tal como se muestra en la Figura A.1, en el cual la trama indica su Categoría de Tráfico (0-7) o Flujo de Tráfico y el tamaño de la cola (en número de bytes) en la QSTA para esta categoría o flujo.

Octetos: 2	2	6	6	6	2	6 ó 0	2	0-2312	4
Control de Trama	Duración-ID	Dirección 1	Dirección 2	Dirección 3	Control de Secuencia	Dirección 4	Control de QoS	Carga útil	FCS

Figura A.1 Encabezado MAC QoS de una trama de datos

Una QSTA que necesita solicitar una TXOP puede transmitir una trama de datos nula o vacía, con el fin de expresar su necesidad. Además, HCCA protege cada TXOP mediante el uso del mecanismo de Detección Virtual de Portadora al fijar el

<sup>87</sup> Se acostumbra a dejar en inglés.

NAV en las otras estaciones con el valor de la duración de la TXOP en lugar de silenciar a todas las estaciones en el QBSS durante el CFP. De esta manera HCCA genera unas Fases de Acceso Controlado (*Controlled Access Phases, CAP*) durante las cuales el HC tiene el control del medio y brinda al resto de estaciones la posibilidad de transmitir paquetes utilizando las TXOPs.

En particular los bits del campo de Control de QoS permiten identificar si la trama pertenece a una categoría de tráfico (TC) o a un flujo de tráfico (TS), el tipo de reconocimiento que se está utilizando (ACK, No ACK, Block ACK), la duración del TXOP (TXOPlimit) asignado por el HC, la duración del TXOP requerido por una estación que desea transmitir, la cantidad de tráfico en el *buffer* de la estación generadora de tráfico, entre otras configuraciones.

Cabe anotar que al ser aprobado el estándar 802.11e a mediados del año 2005 la Alianza Wi-Fi ratificó una especificación interna denominada Wi-Fi Multimedia (*Wi-Fi Multimedia, WMM*) con el propósito de facilitar la interoperabilidad y garantizar la QoS entre diferentes proveedores de equipos según el tipo de tráfico que exista en la red. Para la clasificación de los paquetes, WMM se basa en cuatro categorías: voz, video, *best effort*, y *background*, es decir, WMM se basa en EDCA. De igual manera el grupo de estudio E tomó en consideración a HCCA como mecanismo de acceso generando un nuevo grupo conocido como Acceso Programado Wi-Fi Multimedia (*Wi-Fi Multimedia Scheduling Access, WMM-SA*). Sin embargo, en mayo de 2006, la junta directiva de la Alianza Wi-Fi decidió dar por terminado el grupo WMM-SA dejando así a WMM (EDCA) como único esquema de QoS que puede ser certificado en equipos de la Alianza Wi-Fi (Epshtein, 2006).

## B. Mecanismos Adicionales de QoS en el Nivel MAC de IEEE 802.11e

Además de los mecanismos EDCA y HCCA descritos anteriormente, el estándar 802.11e provee mejoras adicionales a nivel MAC (Intel, 2004) entre las que se encuentran:

- **Ráfagas Libre de Contienda:** Las ráfagas de libre contienda (CFB, *Contention Free Burst*) se pueden definir como un período de tiempo en el cual las QSTA y los puntos de acceso con QoS pueden transmitir un número determinado de tramas una tras otra (separadas por un SIFS) sin la necesidad de competir por el medio cada vez que quiere transmitir una trama. Como se mencionó anteriormente, el número de tramas que se pueden transmitir depende de la Oportunidad de Transmisión ya sea en EDCA o HCCA. En la Figura B. 1 se observa la transmisión de datos sin<sup>88</sup> y con<sup>89</sup> CFB.

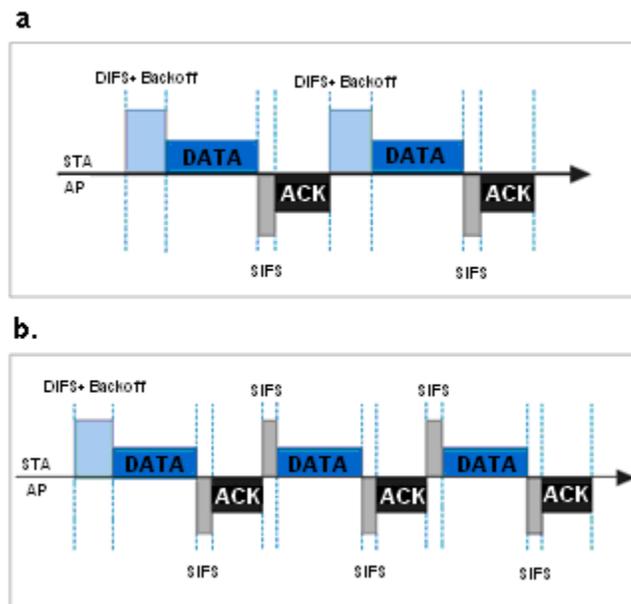


Figura B. 1. Transmisión de datos sin y con CFB

El envío de ráfagas puede mejorar significativamente el desempeño de la red ya que se eliminan los tiempos asociados con un DIFS o con un tiempo de *backoff* aumentando el *throughput* en redes mixtas compuestas por 802.11b y 802.11g. No obstante, la selección de un valor *TXOPLimit*

<sup>88</sup> En este caso una vez la estación haya recibido un ACK del paquete inmediatamente anterior vuelve a competir por acceder el medio esperando un DIFS más un tiempo de *backoff*

<sup>89</sup> Mediante el uso de CFB un cliente puede transmitir múltiples tramas sin la necesidad de generar un DIFS más un tiempo de *backoff* después de recibir un ACK.

apropiado se debe considerar en flujos con baja tolerancia a los retardos. En cualquier instante, cuando se adquiere una TXOP el máximo número de tramas que pueden enviarse depende del número de tramas que se encuentren en el buffer MAC así como de la tasa de transmisión a nivel físico. Por lo tanto, el *TXOPLimit* debe ser directamente proporcional al tamaño del buffer e inversamente proporcional a la tasa de transmisión a nivel físico (Selvakennedy, 2004).

- **Nuevas reglas de Reconocimiento (ACK):** Al igual que en el estándar 802.11, en 802.11e todas las tramas *unicast* requieren de una trama de control ACK inmediata pero existen dos nuevas opciones que se especifican en campo de Control de Qos en las tramas de datos. Estas son:
  - *No ACK:* Incrementa la eficiencia al no enviar ACKs en ciertas aplicaciones. Esta funcionalidad es especialmente útil para aplicaciones con una baja tolerancia al retardo pero que pueden aceptar cierta pérdida de paquetes.
  - *ACK de Bloque (Block ACK):* Mejora la eficiencia al agregar un ACK por múltiples tramas (bloque) a través de una sola respuesta. Este puede ser de dos tipos: inmediato y con retardo. En el primer caso el receptor debe responder inmediatamente con una Trama ACK de Bloque una vez este haya recibido una petición por parte del transmisor. Esta petición se realiza cuando el transmisor termina de enviar múltiples tramas en el CFB durante el TXOP. Para el segundo tipo, la estación receptora responde a la petición del transmisor informándole que la transmisión del ACK de Bloque se retardará un poco. Esto con el fin de realizar un mejor cálculo del ACK en sistemas de bajo desempeño.
- **Protocolo de Enlace Directo:** La especificación general del estándar 802.11 permite que en una red en modo infraestructura el flujo de datos se realice solamente entre las estaciones y el AP. El Protocolo de Enlace Directo (DLP, *Direct Link Protocol*) en el estándar 802.11e permite el intercambio de información entre estaciones de manera directa sin la necesidad de atravesar el punto de acceso tal como se muestra en la Figura B. 2. Esta funcionalidad puede incrementar aproximadamente hasta dos veces el ancho de banda disponible en una comunicación cliente-cliente siempre y cuando ambas se encuentren en el rango de transmisión de cada una de ellas. Además, puede existir una mejora en el desempeño del sistema dado el caso en que las estaciones se encuentren más próximas entre sí y que el estado del enlace sea mejor entre ellas en comparación al enlace con el punto de acceso.

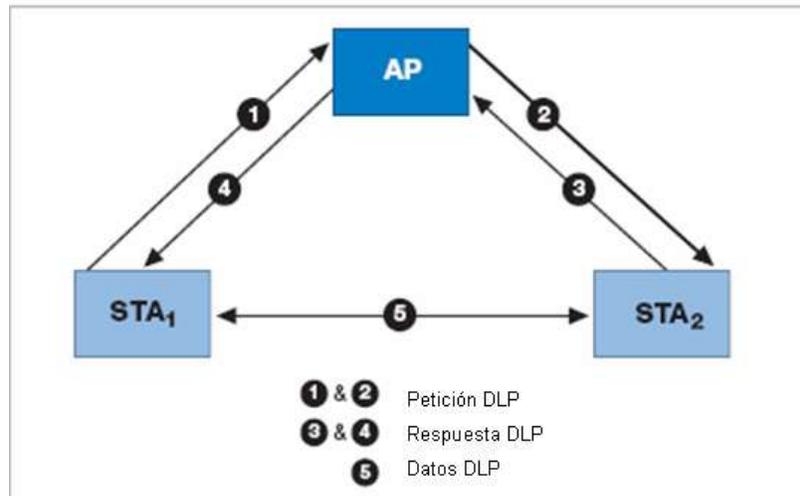


Figura B. 2. Ilustración del Protocolo de Enlace Directo

- *Piggybacking*: Como se mencionó anteriormente, al enviar datos y ACKs sobre la misma trama se reducen los sobre-encabezados (*overhead*) mejorando el desempeño general de la red.
- *Entrega Automática con Ahorro de Energía*: La Entrega Automática con Ahorro de Energía (APSD, Automatic Power-Save Delivery) es una mejora al mecanismo de Ahorro de Energía (*Power Save*) del estándar 802.11 que contribuye con la prolongación de la vida de las baterías permitiendo la opción de apagar los dispositivos durante la mayoría del tiempo mediante la programación de un horario (*schedule*) basado en el patrón de repetición de la trama *Beacon*. Esta funcionalidad es apropiada para aplicaciones VoIP en donde los paquetes se transmiten en intervalos de tiempo periódicos, y para dispositivos como, por ejemplo, los teléfonos VoWLAN/VoWiFi.



## C. Configuración de Asterisk para Generación de Flujos de VoIP con QoS

La instalación del servidor PBX Asterisk se realiza con los siguientes pasos:

1. export http\_proxy=http://10.200.2.173:3128
2. apt-get install g++
3. apt-get install libncurses5-dev
4. apt-get install libstdc++5
5. apt-get install ssh
6. uname -a  
linux servidor **2.6.22-14-generic** #1 SMP Sun Oct 14 23:05:12 GMT 2007 i686 GNU/Linux
7. apt-cache search linux-headers **2.6.22-14-generic**  
linux-headers-**2.6.22-14-generic** - Linux kernel headers for version 2.6.22 on x86/x86\_64
8. apt-get install linux-headers-**2.6.22-14-generic**
9. ln -s /usr/src/linux-headers-**2.6.22-14-generic** /usr/src/linux-2.6
10. Descargar de <http://www.asterisk.org/downloads>:  
asterisk.tar.gz-1.4.17.tar.gz  
libpri-1.4.3.tar.gz  
zaptel-1.4.8.tar.gz
11. cp \*.tar.gz /usr/src/
12. tar -zxvf zaptel-1.4.8.tar.gz  
cd /zaptel  
make clean  
make install
13. tar -zxvf libpri-1.4.3.tar.gz  
cd /libpri  
make clean  
make install
14. tar -zxvf asterisk-1.4.17.tar.gz  
cd /asterisk  
./configure
15. make clean  
make install  
cd ..
16. make samples

Luego se procede a realizar la configuración del plan de marcado (*Dialplan*), en el archivo "extensions.conf".

```
;Se declara la sección general  
;Se declara el contexto que maneja las llamadas
```

```
[tutorial] ;Contexto Ciclo
exten=s,1,BackGround,demo-instruct ;Reproduce el mensaje demo-
instruct que esta ya ;grabado en el servidor
exten=s,2,Goto(tutorial,s,1) ;Crea un ciclo para que se repita el
mensaje ;indefinidamente
exten=101,1,Goto(tutorial,s,1) ;cuando el softphone llama a 101
entra al ciclo
```

Posteriormente se realiza la configuración para los clientes Asterisk en el servidor, en el archivo "iax.conf".

```
[cliente10]
type=friend ;Puede hacer y recibir llamadas
host=172.16.15.110 ;Se conectara; desde esa IP
context=tutorial ;Podra; llamar a los numeros que se declaren en
ese contexto
qualify=yes ;Se monitorea la conexion y el retardo
disallow=all ;No permite un codec diferente a
allow=ulaw ;g711u
```

```
[cliente9]
type=friend ;Puede hacer y recibir llamadas
host=172.16.15.109 ;Se conectara; desde esa IP
context=tutorial ;Podra llamar a los numeros que se declaren en
ese contexto
qualify=yes ;Se monitorea la conexion y el retardo
disallow=all ;No permite un codec diferente a
allow=ulaw ;g711u
```

```
[cliente8]
type=friend ;Puede hacer y recibir llamadas
host=172.16.15.108 ;Se conectara; desde esa IP
context=tutorial ;Podra; llamar a los numeros que se declaren en
ese contexto
qualify=yes ;Se monitorea la conexión y el retardo
disallow=all ;No permite un codec diferente a
allow=ulaw ;g711u
```

```
[cliente7]
type=friend ;Puede hacer y recibir llamadas
host=172.16.15.107 ;Se conectara desde esa IP
context=tutorial ;Podra llamar a los numeros que se declaren en
ese contexto
qualify=yes ;Se monitorea la conexión y el retardo
disallow=all ;No permite un codec diferente a
allow=ulaw ;g711u
```

```

[cliente6]
type=friend          ;Puede hacer y recibir llamadas
host=172.16.15.106  ;Se conectara; desde esa IP
context=tutorial    ;Podra; llamar a los numeros que se declaren en
ese contexto
qualify=yes         ;Se monitorea la conexión y el retardo
disallow=all        ;No permite un codec diferente a
allow=ulaw          ;g711u

[cliente5]
type=friend          ;Puede hacer y recibir llamadas
host=172.16.15.105  ;Se conectara desde esa IP
context=tutorial    ;Podra llamar a los numeros que se declaren en
ese contexto
qualify=yes         ;Se monitorea la conexión y el retardo
disallow=all        ;No permite un codec diferente a
allow=ulaw          ;g711u

jitterbuffer=yes
dropcount=2
maxjitterbuffer=500
maxexcessbuffer=100
minexcessbuffer=50
jittershrinkrate=1

```

Cabe resaltar que las últimas 4 líneas del archivo “`iax.conf`” en el servidor PBX Asterisk se agregaron para activar los comandos `iax2 show channels` y `iax2 show netstats`.

Por último se realiza la configuración de los clientes asterisk, en el archivo “`iax.conf`”.

```

;Se declara la sección general
;Se declara el servidor que le llamaremos ap
[AP]
type=friend
host=172.16.15.102 ;dirección a la que se conectara
context=tutorial
qualify=yes
disallow=all
allow=ulaw
trunk=no ; para que las llamadas se hagan de manera individual

```

Una parte muy importante para resaltar en este proceso de configuración es aquella en la que se habilita la línea `tos=ef` en el archivo “`iax.conf`” y en “`iaxprov.conf`” para habilitar la marcación de los paquetes de voz a nivel 3 de tal forma que puedan luego ser mapeados por el driver MadWiFi a la AC\_VO de WMM/EDCA.

Para realizar un llamada se creó el siguiente *script*, el cual se copia en la dirección */var/spool/asterisk/outgoing* en cada cliente del que se generara una llamada, ya que asterisk esta monitoreando constantemente este directorio, tan pronto se copia este *script* en este directorio con un extensión *\*.call* se genera un llamada de acuerdo a lo en él indicado. A continuación aparece el que fue utilizado para este caso específico:

```
; Archivo sample.call
Channel: IAX2/ap ; llama al servidor 'ap'TM
MaxRetries: 2 ; hasta dos reintentos
RetryTime: 60 ; Reintento cada 60 segundos
WaitTime: 30 ; Esperara 30 segundos la respuesta
Context: tutorial ; La llamada llegara; al contexto tutorial,
Extension: s ; extension s y
Priority: 1 ; prioridad 1
```

# D. Scripts para la Infraestructura de Experimentación

## Configuración de AP virtual y clientes inalámbricos

Para la creación del AP virtual, se generó el siguiente *script* donde también se crea un puente entre la interfaz inalámbrica y la interfaz ethernet.

```
#!/bin/bash
ifconfig ath1 down
wlanconfig ath1 destroy
wlanconfig ath1 create wlandev wifi0 wlanmode ap -bssid
mpstat -P 0 1 1

iwconfig ath1 essid "redwmm51"
ifconfig ath1 up
echo "ap0 Created OK"

brctl addbr br0
brctl addif br0 eth0
brctl addif br0 ath1
ifconfig br0 172.16.15.102 netmask 255.255.0.0 broadcast 172.16.255.255
up
route add default gw 172.16.255.254 br0
route add -net 172.16.0.0 netmask 255.255.0.0 dev br0
echo "BRIDGE Created OK"
```

Para crear clientes que se conecten a la red "redwmm51"<sup>90</sup> se creó el siguiente *script*:

```
#!/bin/bash
sleep 1s
ifconfig ath0 down
sleep 1s
ifconfig eth0 down
sleep 1s
wlanconfig ath0 destroy
sleep 1s
wlanconfig ath0 create wlandev wifi0 wlanmode sta
sleep 1s
iwconfig ath0 essid "redwmm51"
sleep 1s
ifconfig ath0 up
sleep 1s
route add default gw 172.16.255.254 ath0
sleep 1s
route add -net 172.16.0.0 netmask 255.255.0.0 dev ath0
sleep 2s
ifconfig ath0 172.16.15.101 netmask 255.255.0.0 broadcast 172.16.255.255
up
echo "IP,GW, STA Created OK, "
```

---

<sup>90</sup> Que es el SSID especificado para la red en el *script* anterior

El siguiente paso es establecer la velocidad de la conexión inalámbrica a 11Mbps<sup>91</sup>, con el comando:

```
iwconfig ath0 rate 11M
```

Luego se puede activar el soporte para QoS con WMM/EDCA para realizar las pruebas con el comando:

```
iwpriv ath0 wmm 1
```

o para desactivarla con:

```
iwpriv ath0 wmm 0
```

Para verificar el estado de soporte (si está habilitado o no) se utiliza el comando:

```
iwpriv ath0 get_wmm
```

el cual generará un 1 si está activado o un 0 no.

### **Script para listar cada segundo los parámetros de cada estación**

Para el proceso de estar monitoreando los valores de parámetros de las diferentes estaciones del QBSS para asegurar que las pruebas se realizaran antes que regresaran al valor original EDCA en cualquier estación, se creó el siguiente *script*:

```
#!/bin/bash
clear
a=true
i=0
echo "Esto es una prueba"
while (test "$a" = "true")
do

sleep 1s
wlanconfig ath0 list wme
i=i+1;
echo "prueba "
echo i
done
```

---

<sup>91</sup> Para dejarla trabajando con 802.11b

## E. Cambio de Valores de Parámetros con MadWiFi en Linux

En este apéndice se examina lo concernientes a la instalación y uso del Driver Madwifi que permite realizar los cambios en los valores de parámetros de EDCA/WMM para el trabajo experimental en Laboratorio.

Para el presente caso el driver usado fue el [madwifi-ng-r1816-20061118.tar.gz](http://madwifi-ng-r1816-20061118.tar.gz):

```
tar -zxvf madwifi-ng-r1816-20061118.tar.gz
```

Antes de proceder a compilar e instalar el driver es necesario bajar las interfaces de red inalámbricas que existan (si las hay):

```
ifconfig ath0 down.  
Ifconfig wifi0 down.
```

Después de lo anterior se desinstalan los módulos anteriores de MADWifi para ello se entra a la carpeta donde se descomprimió el contenido del archivo y se ejecutan los siguientes comandos:

```
cd madwifi-ng-r1816-20061118  
cd scripts  
./madwifi-unload.bash  
./find-madwifi-modules.sh /lib/modules/
```

A continuación se preguntará si se quieren remover los módulos antiguos de MADWifi, lo más conveniente es responder afirmativamente.

Después de lo anterior se procede a compilar e instalar el driver.

```
cd madwifi-ng-r1816-20061118  
make  
make install
```

Por último para probar la correcta instalación del Driver se ejecuta el siguiente comando:

```
modprobe ath_pci
```

Si no se genera ningún inconveniente el Driver ha quedado adecuadamente instalado.

## USO DEL DRIVER MADWifi.

A través del Driver MADWifi es posible usar la tarjeta inalámbrica en 2 modos como puntos de acceso – AP o como estación – STA. Los comandos para establecer la configuración de la tarjeta inalámbrica son:

```
wlanconfig ath0 destroy.
```

Se usa para bajar la interfaz ath0 en caso de que sea necesaria una re-configuración.

```
wlanconfig ath0 create wlandev wifi0 wlanmode sta/ap
```

Se usa para hacer que la tarjeta inalámbrica opere en modo STA o en modo AP.

```
ifconfig ath0 up
```

Se usa para poner arriba la interfaz ath0.

```
wlanconfig ath0 list scan
```

Se usa para escanear los AP's disponibles y a los cuales una tarjeta en modo estación puede unirse.

```
iwconfig ath0 essid "linksys"
```

Sirve para unirse a un determinado AP si la tarjeta esta en modo cliente o darle un nombre al SSID del AP si la tarjeta esta en modo STA.

## USO DE EDCA/WMM EN LAS TARJETAS INALÁMBRICAS.

```
iwpriv ath0 get_wmm
ath0 get_wmm:1
```

Muestra en qué estado se encuentra EDCA, 1 corresponde a habilitado, 0 corresponde a deshabilitado.

```
wlanconfig ath0 list wme
```

Muestra la lista de categorías EDCA tanto en el AP como en la Estación.

```
iwpriv ath0 get_txoplimit 0 1
```

Muestra la categoría **txoplimit** del tipo de tráfico *Best Effort* en el AP.

```
iwpriv ath0 get_txoplimit 0 1
```

Muestra la categoría **txoplimit** del tipo de tráfico *Best Effort* en la Estación.

```
iwpriv ath0 cwmin 1 0 1
```

El anterior comando fija en 1 el parámetro CWmin del tipo de tráfico *BackGround* en el AP.

```
iwpriv ath0 aifs 1 1 2
```

El anterior comando fija en 1 el parámetro AIFS del tipo de tráfico *Background* en la estación.

El índice de los tipos de tráfico es como sigue:

Índice	Categoría
0	BE Best Effort
1	BK Background
2	VI Video
3	VO Voz

En (MadWiFi, 2007) se encuentra la Guía de Usuario del Driver Madwifi.

## CONFIGURACION DEL ESCENARIO DE PRUEBAS.

Para entrenarse en un escenario básico de laboratorio y comprobar el funcionamiento de WMM/EDCA se realizaron las siguientes pruebas.

### Listado de los valores de parámetros EDCA

Se ejecutó el comando `wlanconfig ath0 list wme` esta instrucción muestra el valor de los parámetros EDCA para cada una de las AC.

En el primer renglón aparecen los valores correspondientes al dispositivo donde se ejecuta el comando (sea AP o STA) y en el renglón que le sigue se ven los valores del dispositivo con el que está asociado (sea STA o AP respectivamente).

En la Figura E.1 se puede apreciar el resultado de la ejecución del comando.

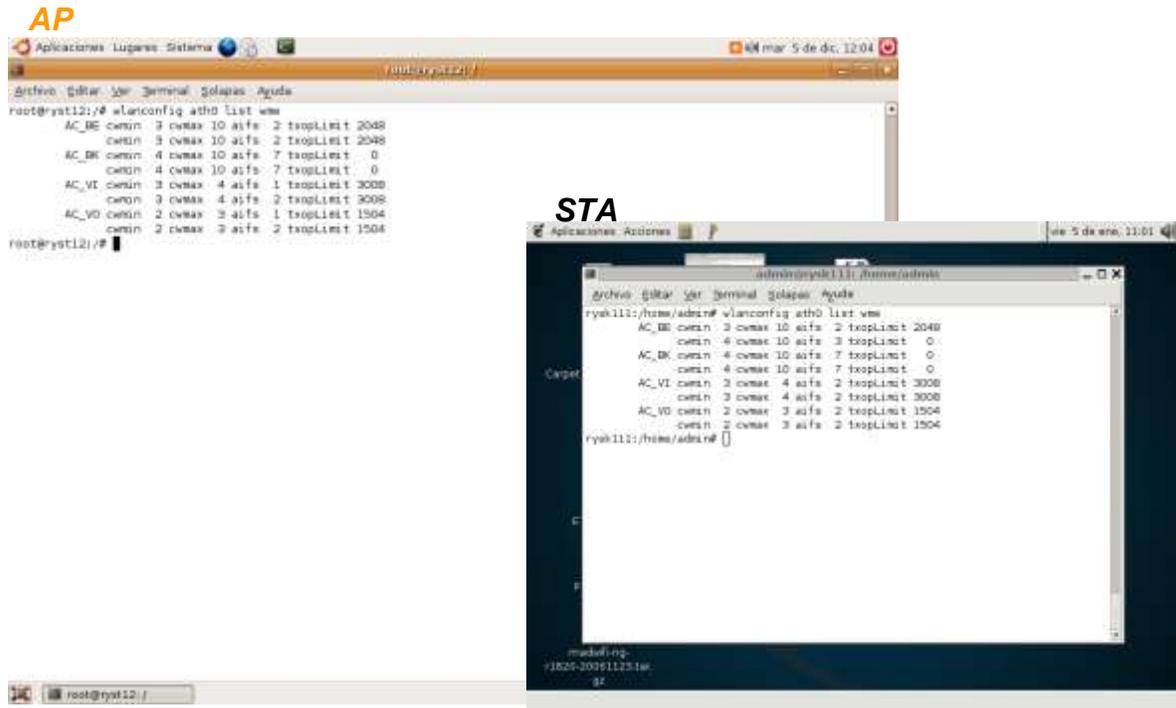


Figura E. 1 Listado de valores de parámetros EDCA en el AP y en la STA

### Cambio del AIFS en el Cliente

Con el comando `iwpriv ath0 aifs 1 1 2` se puede cambiar el parámetro AIFS en los valores EDCA del cliente desde el AP<sup>92</sup>, esto se debe a que desde el AP se puede cambiar los valores de los clientes que estén asociados a él, pero desde el cliente solo se pueden cambiar sus propios parámetros EDCA, comprobándose de esta forma el funcionamiento de la trama *beacon* de 802.11e a través de la que el AP puede inducir modificaciones en los parámetros<sup>93</sup>.

Las dos primeras palabras de la instrucción hacen referencia a la interfaz inalámbrica que se está manejando desde Linux; la tercera palabra hace referencia a el parámetro EDCA que se desea modificar; el primer valor hace referencia al tipo de dato que se desea modificar (0 para BE, 1 para BK, 2 para VI, 3 para VO); el segundo hace referencia, a si se desea modificar en el cliente para lo que debe ser 1 y si se desea modificar en el AP este valor debe ser 0; y el último representa el valor que se le dará al parámetro EDCA.

En la Figura E.2 se observa que en el cliente el valor del parámetro *Background* estaba en 7, y después de la ejecución (en el PC que hace las veces de AP) de los comandos

<sup>92</sup> Se encontró que funciona para todos menos para el AC\_BE

<sup>93</sup> En principio, para ajustarlos de acuerdo a las condiciones de la red

```
iwpriv ath0 aifs 1 0 2
iwpriv ath0 aifs 1 1 2
```

este valor se modificó tanto en el cliente como en el AP a un valor de 2.

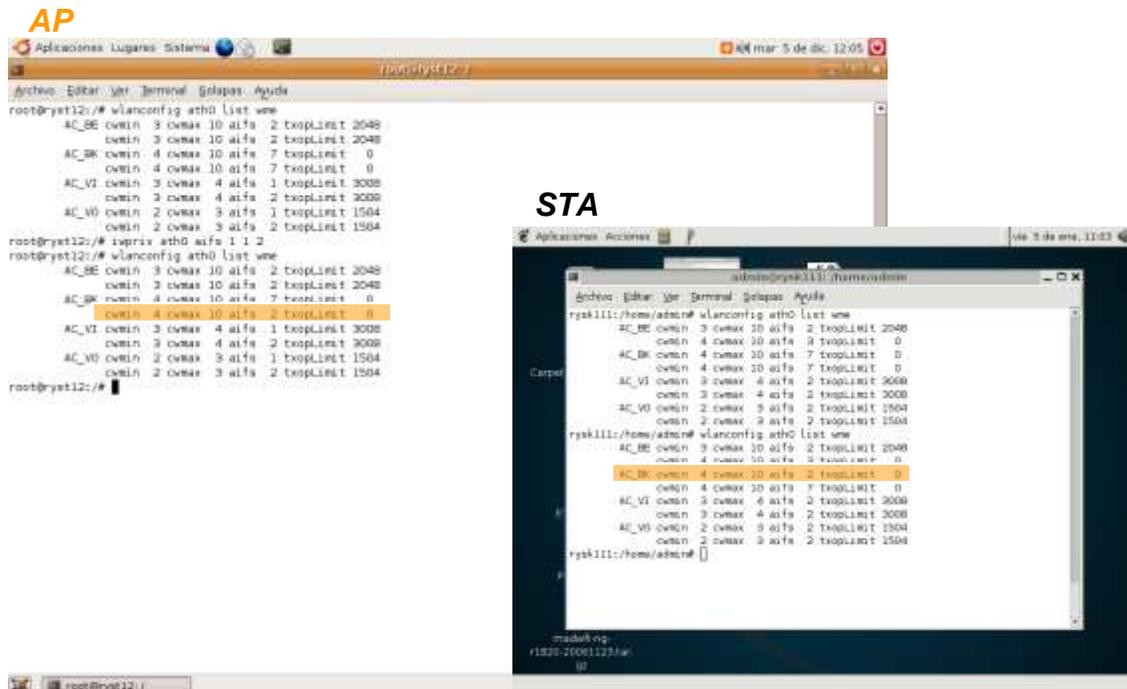


Figura E.2. Variación del AIFS[AC\_BK] de EDCA

### Cambio del valor de CWmin

Con el comando `iwpriv ath0 cwmin 3 1 15`; se puede cambiar el parámetro CWmin de EDCA del cliente desde el AP.

Las dos primeras palabras de la instrucción hacen referencia a la interfaz inalámbrica que se está manejando desde Linux en el AP; la tercera palabra hace referencia a el parámetro EDCA que se desea modificar, para este caso es el de la ventana de contención mínima, CWmin; el primer número hace referencia a la AC que se desea modificar 3 para este caso (AC\_VO); el segundo hace referencia a que se va a modificar en los clientes, por esta razón este número es 1; y el último representa el valor que se le desea dar al parámetro CWmin, para este caso 15.

En la Figura E. 3 se observa que en el cliente el valor del parámetro de Voz estaba en 2, y después de la instrucción este valor se modificó tanto en el cliente como en el AP a un valor de 15. De una manera similar se procede para el cambio de la CWmax.

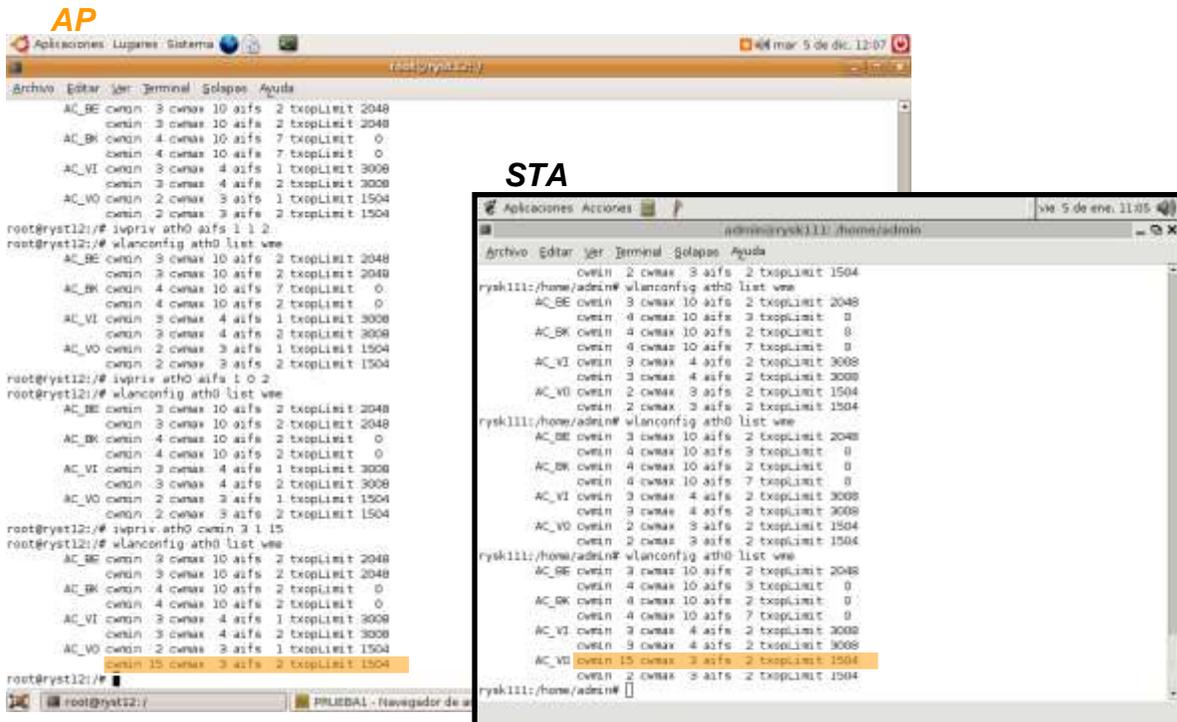


Figura E. 3 Variación del valor del parámetro CWmin de EDCA en la STA

## F. Manejo del Simulador NCTUns 4.0

En este Apéndice se presenta lo concerniente al uso de la herramienta de simulación NCTUns (National Chiao Tung University Network Simulator), creada por el profesor Shie-Yuan Wang. Se expondrá el software necesario para ello y, para facilitar su comprensión, se explicará su modo de uso a través de un ejemplo con EDCA.

Debido a que NCTUns se basa en una metodología de simulación de reingreso de kernel y que sólo funciona en sistemas Linux, no todos los sistemas operativos de Unix cumplen con los requisitos necesarios para ello, en específico para ejecutar el NCTUns 4.0 se debe trabajar con Fedora 8.0 (Wang, *et al*, 2007). Una vez instalado este sistema operativo se deben instalar los paquetes de G++ y deshabilitar el firewall y el SELinux.

A continuación se expondrá paso a paso la simulación de una pequeña red inalámbrica que soporte QoS en la que se evalúe la función de acceso al medio EDCA.

La topología que se utiliza se indica en la Figura F.1 en la que aparece resaltados el icono que permite las conexiones cableadas entre los dispositivos que así lo requieran y enfatiza el modo **Draw Topology** bajo el cual se realiza la inserción de nodos y conexiones.

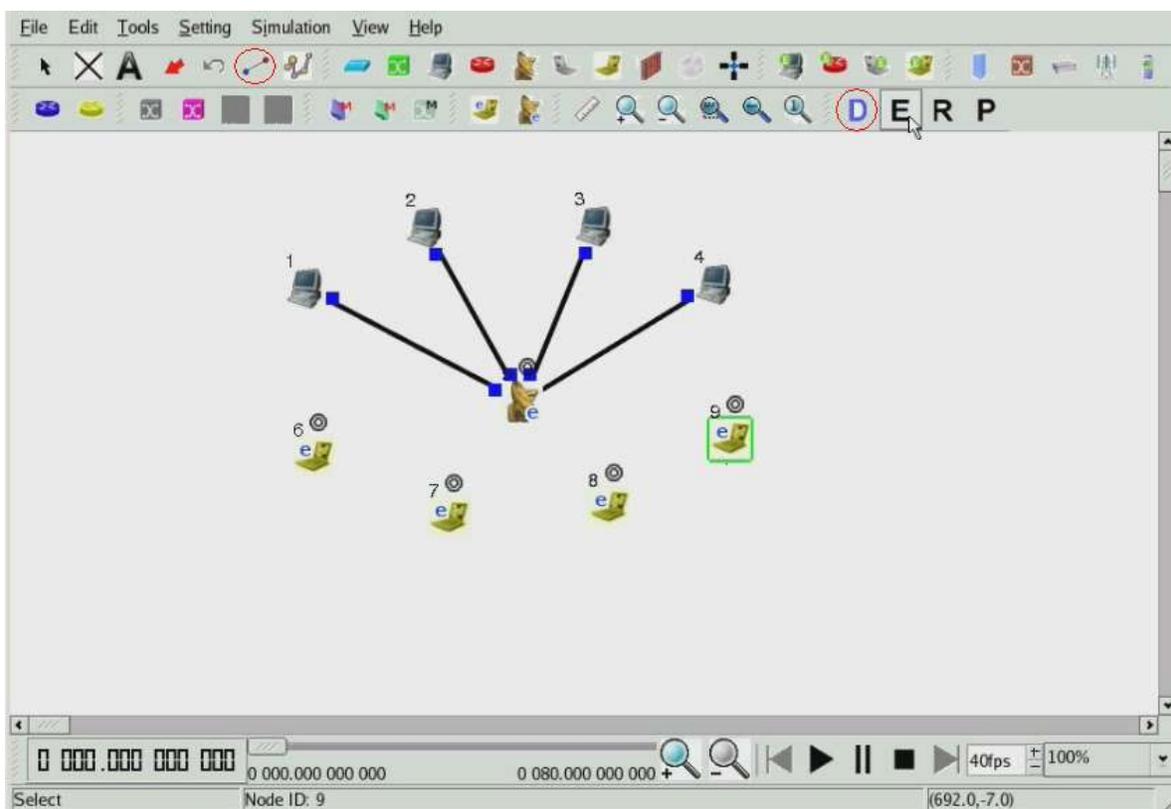


Figura F.1. Topología de red.

Las figuras F.2 y F.3 muestran las ventanas para configuración de los diferentes niveles de una QSTA y un QAP respectivamente.

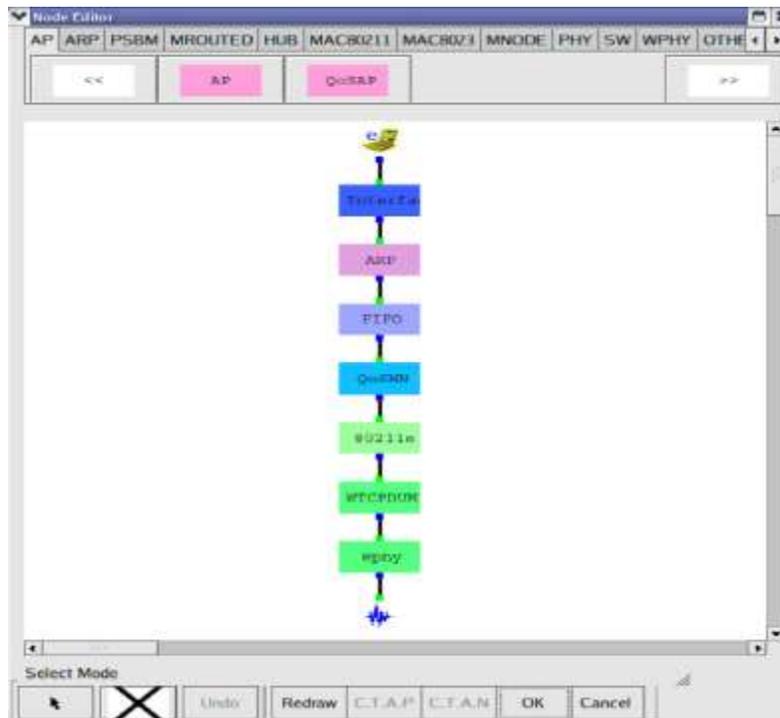


Figura F.2. Ventana de editor de nodo de una estación inalámbrica 802.11e

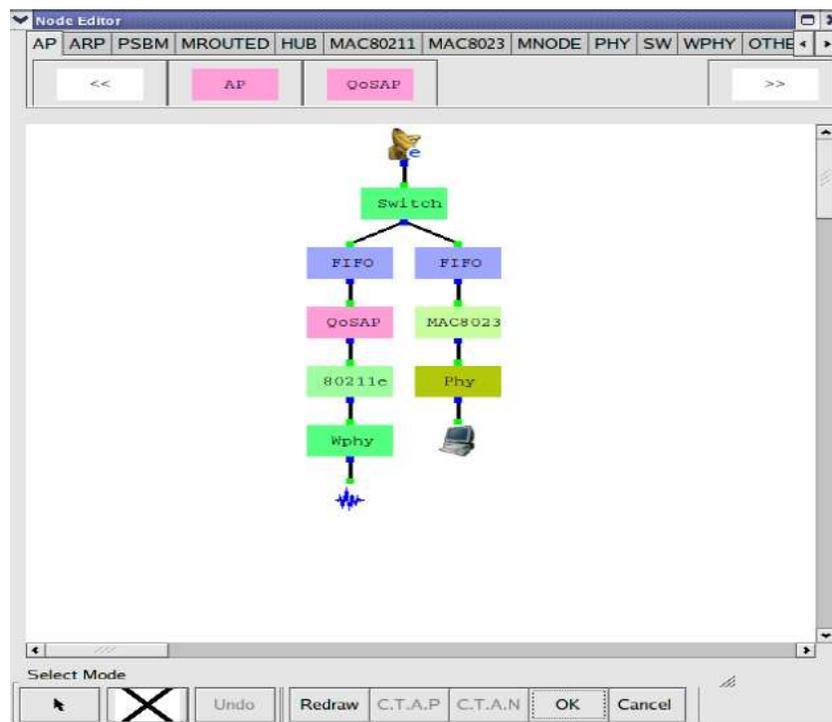


Figura F.3. Ventana de editor de nodo de un AP 802.11e

Cada vez que se inserta un terminal inalámbrico 802.11e debe establecerse su prioridad (ver Figura F.4).

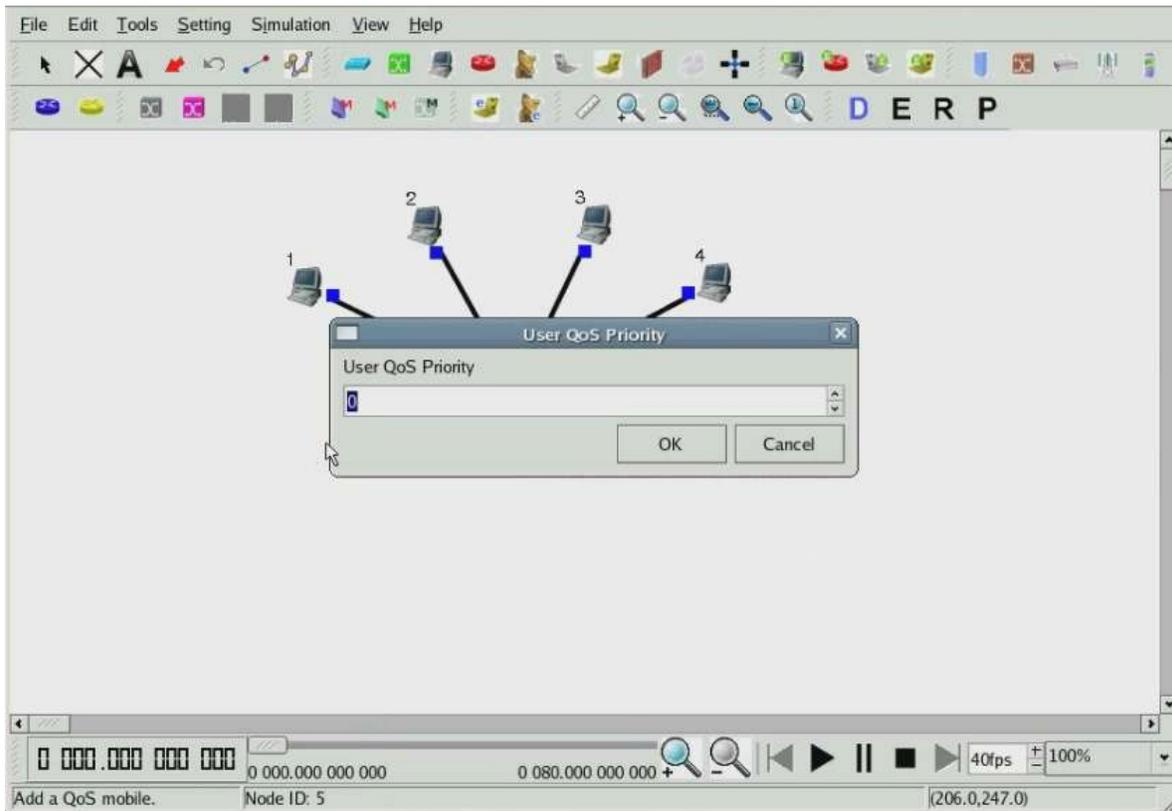


Figura F.4. Prioridad QoS de terminales 802.11e.

Para este caso los nodos 6, 7, 8 y 9 tendrán prioridad 1, 3, 5 y 7 respectivamente. Como se sabe, las prioridades de estos tipos de terminales varían de 0 a 7 representando el tipo de tráfico del cual se trata según se indica en la Tabla F.1.

Priority	UP (Same as 802.1D user priority)	802.1D designation	AC	Designation (informative)
Lowest ↓ Highest	1	BK	AC_BK	Background
	2	—	AC_BK	Background
	0	BE	AC_BE	Best Effort
	3	EE	AC_BE	Best Effort
	4	CL	AC_VI	Video
	5	VI	AC_VI	Video
	6	VO	AC_VO	Voice
	7	NC	AC_VO	Voice

Tabla F.1. Tipos de tráfico según 802.1p y su AC de 802.11e respectiva

Volviendo al ejemplo original, para pasar al modo **Edit Property** se debe guardar el archivo de simulación tal como se aprecia en la Figura F.7.

Luego, se va al menú **Edit** -> **Select All 802.11(e) Mobile Nodes** para seleccionar todos los nodos 802.11e y asignar direcciones IP a los mismos a través del menú **Tools** -> **802.11(e) Mobile Nodes** -> **Generate 802.11(e) Mobile Nodes' IP and MAC Address** (Need to select some Duch nodes in the Edit Property). Ahora pueden agregarse los comandos pertinentes de cada nodo por lo que se da doble click sobre el dispositivo de interés, dando a su paso la aparición de una nueva ventana donde podrá modificarse su configuración.

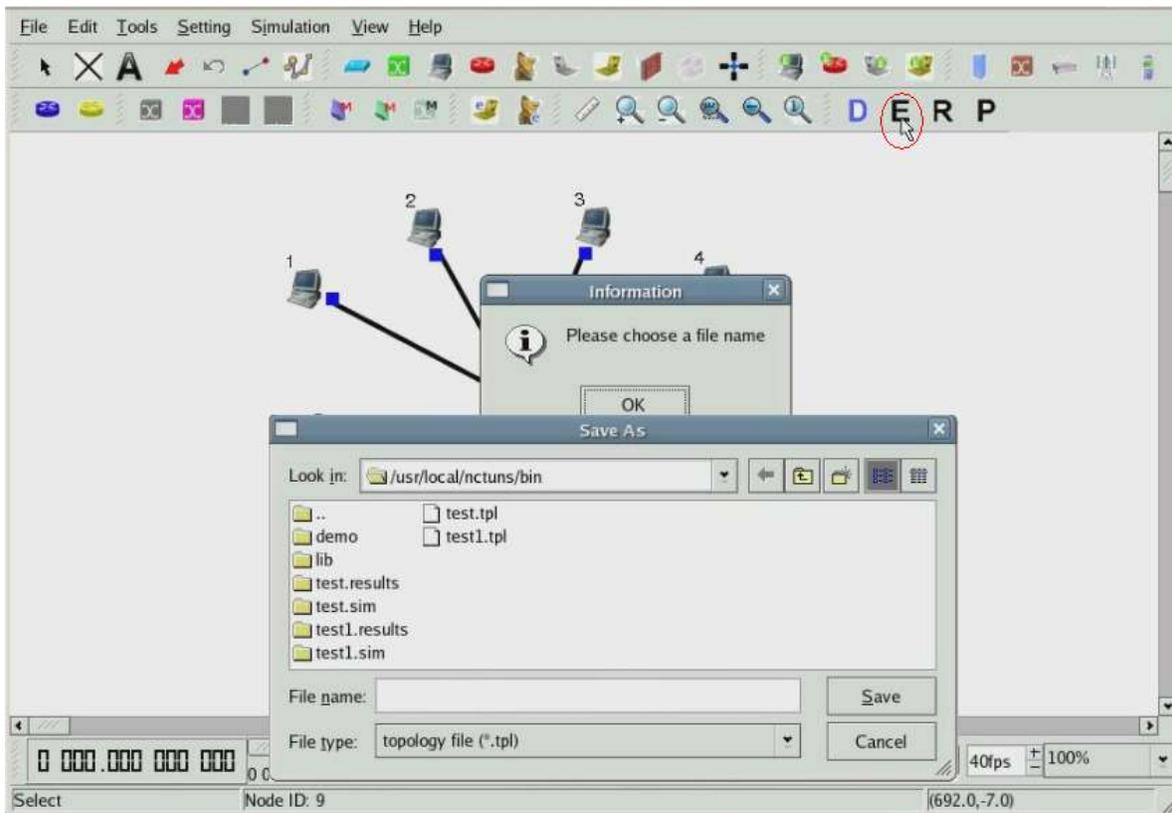


Figura F.7. Modo Edit Property.

Luego de lo anterior se establece el tráfico que se va a generar como se muestra en la Figura F.8 para lo que se señala en la pestaña de "Applications" sobre el botón "Add" el comando "`tcp -t -u -s -p 8000 1.0.1.1`" asignado al nodo 6, donde la dirección IP al final del comando corresponde a la del nodo 1. Así mismo, se configura a los nodos 8, 9 y 10 con las direcciones IP de los nodos 2, 3 y 4 respectivamente. Con "`tcp -t`" se indica al simulador que tales nodos tendrán la tarea de emitir paquetes hacia las direcciones IP señaladas a través de los puertos indicados con "`-p # de puerto`". Para mayor información se puede remitir al botón "App. Usage" el cual contiene el significado de los comandos y su apropiado uso.

Para los nodos 1, 2, 3 y 4 se realiza de manera similar, sólo que estos serán configurados de modo que reciban la información que fue enviada hacia ellos ("*ttcp -r*") y por el número de puerto asignado ("*-p # de puerto*"), tal como se muestra en la Figura F.9.

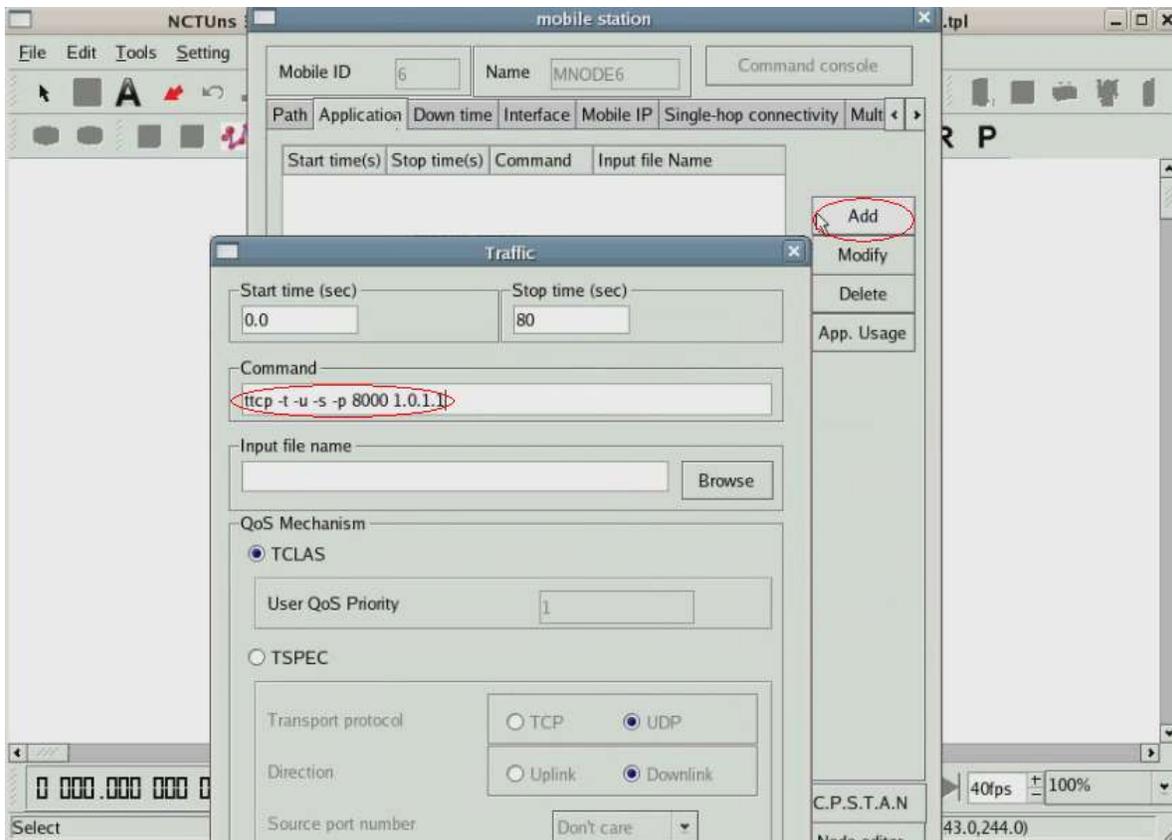


Figura F.8. Inserción de comandos en los nodos inalámbricos de red.

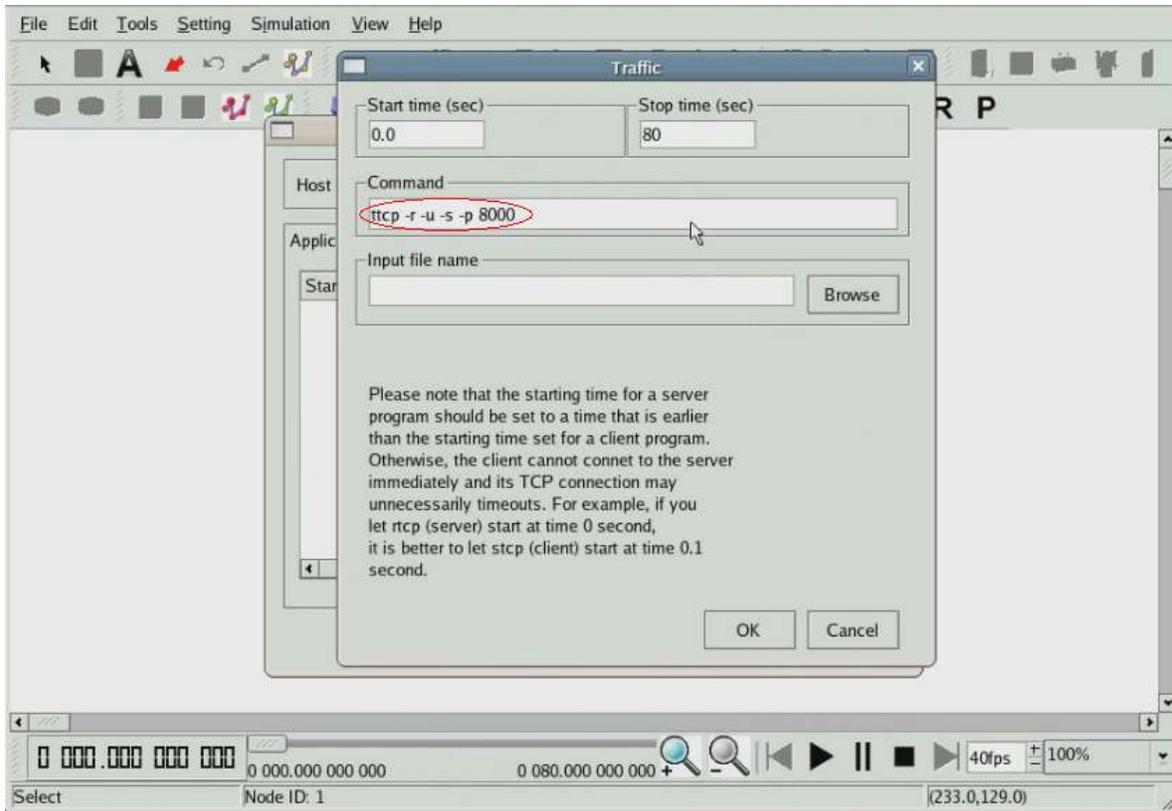


Figura F.9. Inserción de comandos en los nodos cableados de red.

En particular, la herramienta de simulación NCTUns en su versión 4.0 trae consigo un conjunto de opciones para generar tráfico a través de la red en cuestión, para conocer las aplicaciones que se pueden utilizar se puede hacer uso en el botón “App. Usage” de la Figura F.8, con lo que se presenta una explicación bastante resumida de la sintaxis para el uso de las mismas, esto se ve en la Figura F.10.

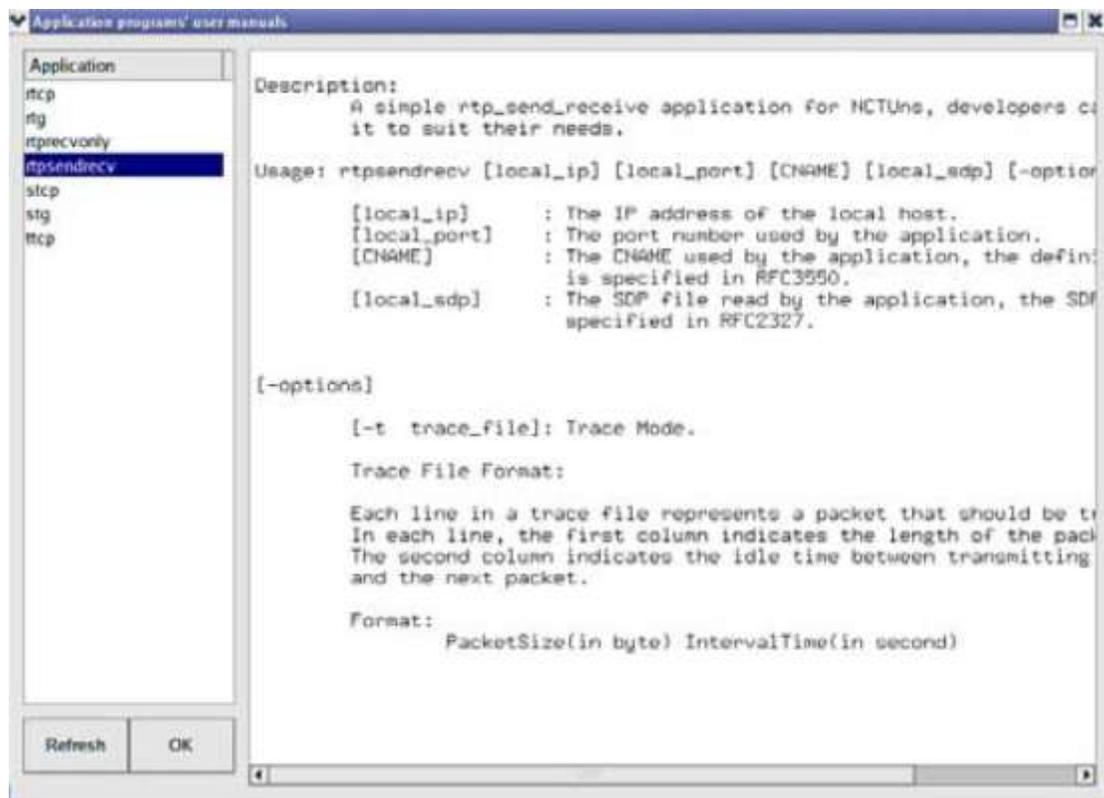


Figura F.10. Ventana de manual de usuario sobre aplicaciones programadas.

Para introducir alguno de los comandos que llaman las aplicaciones se presiona el botón “Add” lo cual dará lugar a la ventana indicada en la siguiente Figura F.11. Su uso se basa en el ingreso de un comando o programa de aplicación en el campo “Command” en donde adicionalmente el usuario está en capacidad de modificar los tiempos de inicio y fin de la generación de tráfico. Esto se muestra en la Figura F.11. Si en algún momento el usuario desea agregar un nuevo comando, modificar o borrar uno ya existente sólo es necesario hacer click sobre los botones “Add”, “Modify” o “Delete” respectivamente.

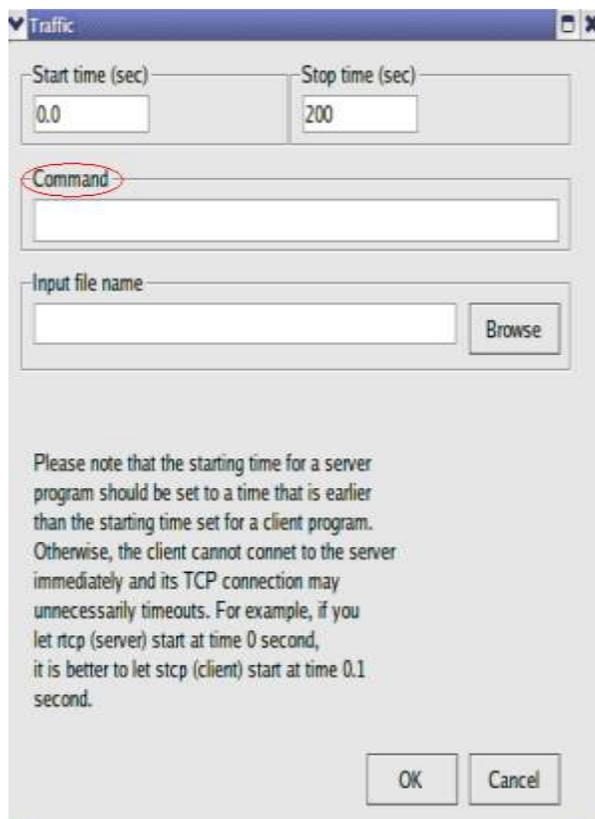


Figura F.11. Ventana de comandos de tráfico.

Entre los comandos utilizados en esta Tesis de Maestría para generar tráfico se encuentran:

#### Comandos:

- **ttcp**

*Sintaxis:*

**ttcp -t [-opciones] dirección IP host destino:** Genera tráfico

**ttcp -r [-opciones ]:** Recibe tráfico

Opciones comunes a ambos comandos:

- u            hace uso de UDP (por defecto TCP)
- p *YYY*      envía o recibe tráfico por el puerto *YYY*

Opciones específicas de la opción -r:

- w *XXX.log*    almacena en un archivo de nombre “*XXX.log*” un registro del throughput por segundo

**Ejemplo 1:**

`ttcp -t -p 8000 1.0.1.2` : Este comando es característico de una estación que genera tráfico encapsulado sobre TCP en el puerto 8000 con destino la estación con dirección IP 1.0.1.2. Nótese que al no incluir la opción `-u`, se genera automáticamente tráfico TCP.

`ttcp -r -p 8000`: La estación que contenga este comando registrará todo el tráfico TCP que reciba por el puerto 8000 proveniente de cualquier estación.

Estos comandos se utilizaron para generar los flujos de Datos (AC\_BE) cuando fue necesario dentro del desarrollo de la Tesis.

- **stg (generador de tráfico), rtg (receptor de tráfico)**

*Sintaxis:*

**stg [-modos] Dirección IP del destino [-opciones]**

[-modos]:

- t Duración (seg) Genera tráfico TCP con una duración en segundos determinada por el usuario
- u Tamaños de la carga útil de datos (bytes) Duración (seg): Se transmite tráfico UDP con una carga útil dada por un tiempo específico
- i XXX.cfg Genera tráfico de acuerdo a las características del archivo de configuración XXX.cfg

[-opciones]

- p YYY Envía tráfico por el puerto YYY

*Formato del archivo XXX.cfg:*

- 
- type:           ###: Protocolo ### ya sea TCP o UDP
  - start\_time:   ###: Tiempo ### expresado en segundos desde el cual se comienza la transmisión del primer paquete.
  - on-off:       ###: Número de líneas entre la línea actual y la línea que contiene la cadena de caracteres "end" que será repetida ### veces.

on: [time: XXX o packet YYY] [distribución de tiempo entre transmisiones consecutivas] [distribución de tamaño de paquetes]: Indica que el tráfico debe ser generado basándose en un intervalo de tiempo específico con una determinada distribución de tiempo y de tamaño de paquetes.

end

---

[time:XXX] indica que se debe generar tráfico durante XXX segundos

[packet:YYY] indica que se deben generar YYY paquetes únicamente

De las anteriores configuraciones solo se debe escoger una.

La distribución de tiempo entre transmisiones consecutivas puede ser:

[const XXX]: indica que el tiempo que transcurre entre el envío de paquetes consecutivos es un valor fijo de XXX segundos.

[uniform XXX, YYY]: el tiempo transcurrido entre el envío de paquetes consecutivos varía entre una valor mínimo igual a XXX segundos y un máximo igual a YYY segundos.

[exponential XXX, YYY, ZZZ]: el tiempo de envío entre paquetes consecutivos es una función exponencial con una media igual a XXX, un valor mínimo de YYY segundos y un máximo de ZZZ segundos.

La distribución del tamaño del paquete puede ser:

[const XXX]: indica que el tamaño de los paquetes es un valor XXX fijo.

[uniform XXX, YYY]: indica que el tamaño de los paquetes generados es función de una distribución uniforme con un valor mínimo de XXX bytes y un máximo de YYY bytes.

[exponential XXX, YYY, ZZZ]: el tamaño de los paquetes generados es función de una distribución exponencial con una media de XXX bytes, un valor mínimo de YYY bytes y un máximo de ZZZ bytes.

off: time: ###: el generador de tráfico deja de transmitir por un período de ### segundos.

Nota para tráfico UDP:

stg soporta cuatro tipos de distribuciones de tiempo para el tráfico UDP:

(1) const XXX: Tasa Constante de Bits (*Constant Bit Rate*, CBR) con distribución de tiempo de XXX segundos

(2) uniform XXX YYY: Distribución con un tiempo mínimo de XXX bytes y un máximo de YYY bytes.

(3) exponential XXX YYY ZZZ: Distribución con promedio XXX, mínimo YYY y máximo ZZZ (Flujos de paquetes con una distribución de Poisson)

(4) greedy: tiempo aleatorio

stg soporta tres tipos de tamaños de paquetes (en bytes) para el tráfico UDP:

- (1) const XXX: constante
- (2) uniform XXX YYY: valor mínimo XXX y máximo YYY
- (3) exponential XXX YYY ZZZ: promedio XXX, valor mínimo YYY y máximo ZZZ.

Nota: El máximo tamaño del paquete no puede ser mayor a los 1500 bytes de la Máxima Unidad de Transferencia (*Maximun Transfer Unit*, MTU) Ethernet.

### Ejemplo 2:

Un ejemplo de generación del *códec* G.711<sup>94</sup> haciendo uso de la opción.

G711.cfg

```
type: udp
start_time: 1
on-off: 1
on: time: 90 const 0.02 length: const 160
end
```

En este caso se genera tráfico con una longitud de trama de 160 bytes con una distribución de tiempo de 0.02 segundos es decir 20ms (const 0.02) por un tiempo de 90 segundos (time) a partir del primer segundo de haber iniciado la simulación (start\_time).

Es así como el comando `stg -i G711.cfg -p 5004 1.0.1.5` envía tráfico por el puerto 5004 hacia la estación con dirección IP 1.0.1.5 de acuerdo a la configuración especificada en el archivo G711.cfg

- **rtg**

*Sintaxis:*

**rtg [-tipo] [-opciones]**

[-tipo]            -t            Conexión TCP  
                   -u            Conexión UDP

[-opciones]

**-p ###**            La estación escucha por el puerto número ### (por defecto 3000)

<sup>94</sup> *Códec* con carga útil de 160 bytes y tamaño de muestra de 20 ms.

-o XXX.log Registra estadísticas de *delay* y pérdida de paquetes en un archivo de nombre XXX.log.

-w YYY.log Registra el *throughput* en un archivo de nombre YYY.log

### Ejemplo 3:

```
rtg -u -o delay_lost.log
```

En este caso la estación receptora generará archivos con nombres delay\_lost.log que contendrán las estadísticas relacionadas con el tráfico UDP entrante por cualquier puerto (no se ha especificado el número del puerto).

En especial este comando fue clave para la Tesis porque permitió revisar el desempeño de los flujos de VoIP (AC\_VO) en los dos aspectos claves anotados: retardo (delay) y pérdidas (lost).

De esta manera se armaron pares entre stg (para enviar) y rtg (para recibir y generar registros) para los flujos de VoIP que se necesitaron en las Simulaciones.

Luego se pasa la modo de **Run Simulation** por lo que debe irse al menú *Simulation* -> *Run* (ver Figura F.12).

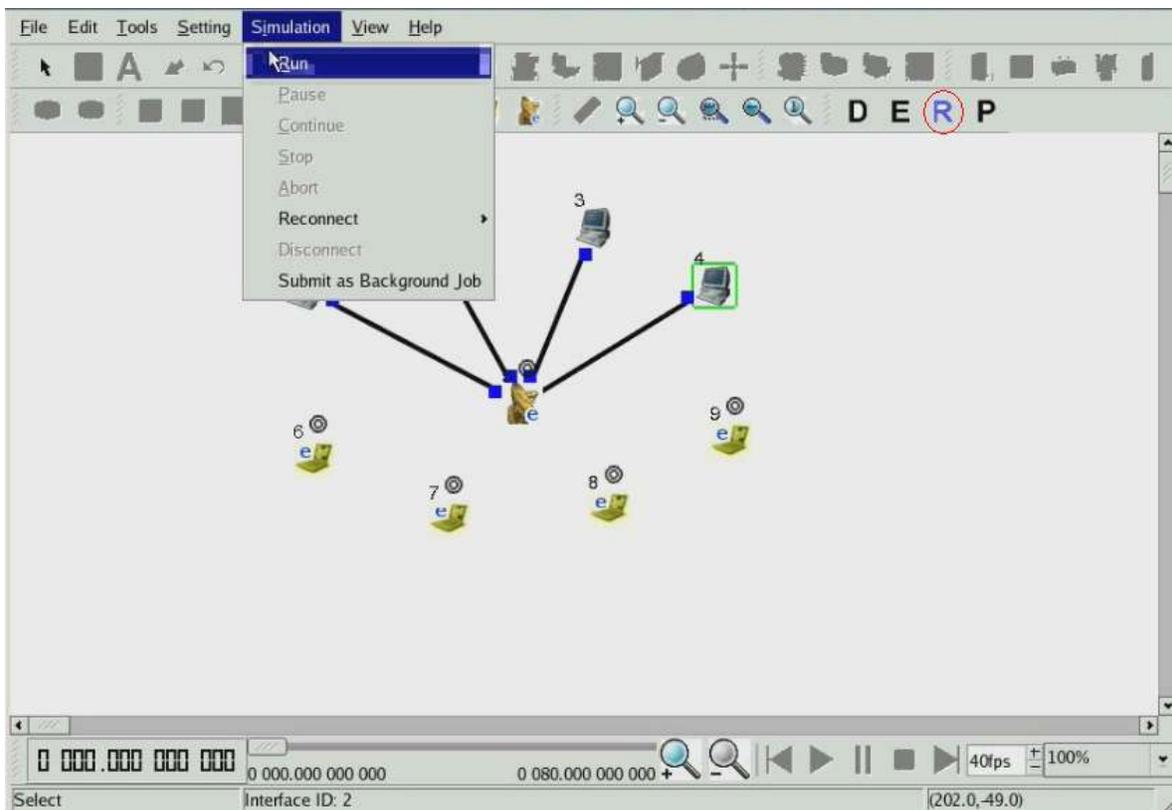


Figura F.12. Simulación de la topología de red.

Finalmente, en el modo **Play Back** puede observarse la animación creada como resultado de la simulación donde podrá observarse de hecho que aquel nodo con categoría más alta ganará el acceso al medio, tal como lo estipula EDCA, mientras que las demás estaciones se verán obligadas a retrasar sus transmisiones (ver Figura F.13).

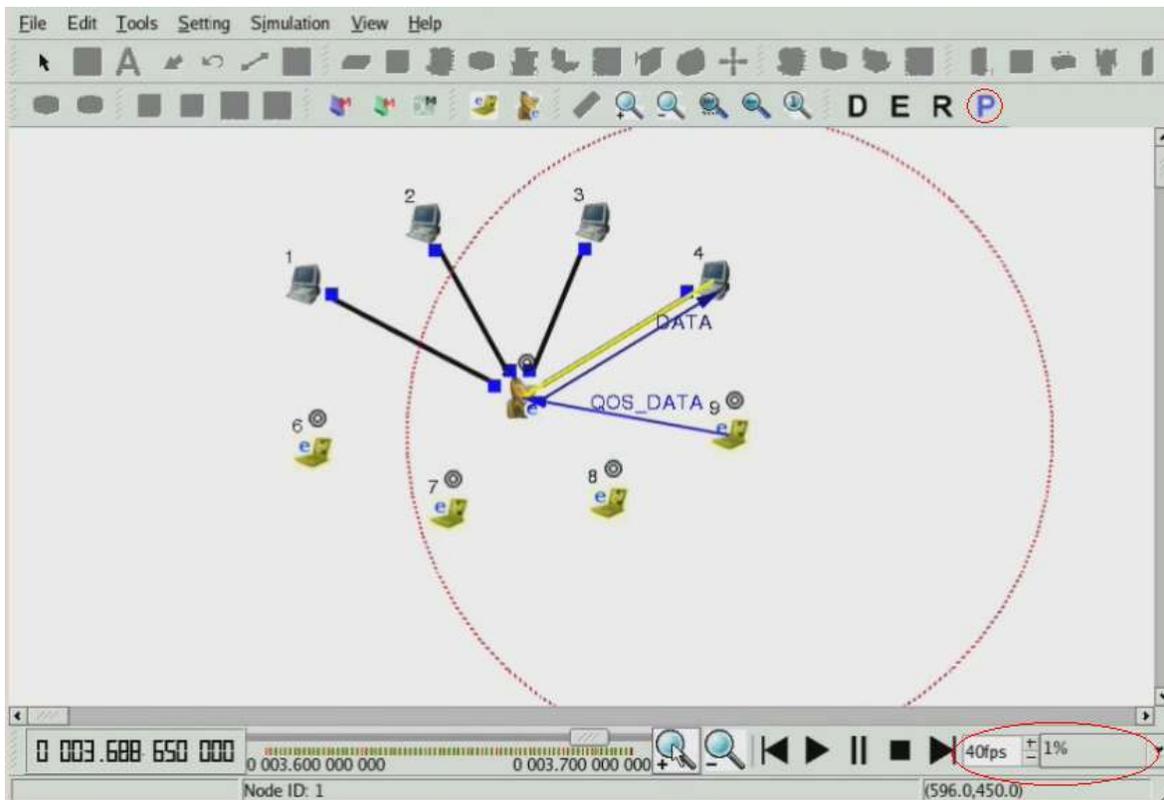


Figura F.13. Resultados vistos mediante el Play Back de NCTUns 4.0



## G. Modificación de Valores de Parámetros EDCA en NCTUns 4.0

Con el desarrollo del proyecto se necesitó descubrir la forma de cambiar los valores de los parámetros WMM/EDCA en el Simulador NCTUns 4.0 dado que esto no se permite por medio de la Interfaz gráfica.

Partiendo, entonces, del hecho que este Simulador suministra el código fuente de su motor de simulación, se encontró que accediendo a la ruta `/NCTUns-4.0/src/nctuns/module/mac` y abriendo los archivos `mac-802_11e.cc` y `mac-802_11e.h` modificando y recompilando se podía realizar el cambio en el valor de los parámetros. Este fue un proceso dispendioso de que necesitó entender la estructura de archivos de las fuentes y encontrar el rol de directorios y archivos, porque no existe documentación al respecto, pero afortunadamente se reitera, fue clave contar con el Código Fuente.

El archivo `mac-802_11e.cc` contiene la lógica de funcionamiento de 802.11e y en especial los valores de los parámetros, mientras que el archivo `mac-802_11e.h` es un archivo de cabecera donde están definidas las variables utilizadas en el primero. Una vez entendido lo anterior, en el `mac-802_11e.cc` se procedió a modificar los valores por defecto que trae el simulador para los valores de AIFSN (como se ve en la Figura G. 1), CWMIN, CWMAX y TXOP (como se muestra en la Figura G.2), para posteriormente realizar varias pruebas de *throughput* para verificar si se producían cambios en los resultados, y de esta forma determinar que el procedimiento era válido, para luego poder ser utilizado dependiendo del escenario específico que necesitara el proyecto.

Después de realizar los cambios, se necesita recompilar el simulador con el fin de que las modificaciones se incluyan en el funcionamiento del mismo. Para ello se debe en primera instancia borrar del sistema los archivos de la compilación previa del simulador, que generalmente quedan ubicados en la ruta `/usr/local/nctuns` para la instalación por defecto. Luego se procede a acceder a la ruta donde se encuentra la carpeta descomprimida de simulador y se recompila y re-instala digitando el comando `./install.sh`.

Finalmente, si los cambios realizados en el código fuente del motor de simulación son correctos, en el momento de finalizar la instalación no se debe obtener ningún mensaje de fallo en el reporte que entrega el programa al final del proceso.

Una vez se han realizado los anteriores pasos, se abre el archivo con la topología creada previamente, se guarda y luego se simula para obtener los resultados correspondientes y poder evidenciar el cambio en los resultados.

```

mac-802_11e.cc *
    return;
}

int mac802_11e::calcul_IFS() {
    aifsn[AC_BE] = 7;
    aifsn[AC_BK] = 3;
    aifsn[AC_VI] = 2;
    aifsn[AC_VO] = 2;

    sifs = phymib->aSIFSTime;
    pifs = sifs + phymib->aSlotTime;
    difs = sifs + 2 * phymib->aSlotTime;
    eifs = sifs + difs + TX_Time(MAC80211E_ACK_LEN);
    for(int i = AC_VO ; i >= AC_BE; i--)
        aifs[i] = sifs + aifsn[i] * phymib->aSlotTime;

    return(1);
}

int mac802_11e::init() {

```

Figura G. 1. Parte del código fuente del archivo mac802\_11e.cc donde se modifica el parámetro AIFSN.

```

mac-802_11e.cc *
    tx_state      = MAC_IDLE,
    tx_active     = 0;

    /* init EDCA 4 AC parameters */
    CWmin[AC_BE] = phymib->aCWmin;
    CWmin[AC_BK] = phymib->aCWmin;
    CWmin[AC_VI] = (phymib->aCWmin+1)/2 - 1;
    CWmin[AC_VO] = (phymib->aCWmin+1)/4 - 1;

    CWmax[AC_BE] = phymib->aCWmax;
    CWmax[AC_BK] = phymib->aCWmax;
    CWmax[AC_VI] = phymib->aCWmin;
    CWmax[AC_VO] = (phymib->aCWmin+1)/2 - 1;

    /* based on WME spec. */
    edca_txop = 0;
    txopLimit[AC_BE] = 0;
    txopLimit[AC_BK] = 0;
    txopLimit[AC_VI] = 188; /* 188 << 5, unit is 32-usec */
    txopLimit[AC_VO] = 102; /* 102 << 5, unit is 32-usec */

    /* initialize buffers */
    epktRx      = 0;
    epktTx      = 0;
    epktBuf     = 0;
    epktCTRL    = 0;

```

Figura G. 2. Parte del código fuente del archivo mac802\_11e.cc donde se modifican los parámetros CWmin, CWmax y la TXOP.

## H. Metodología de Simulación

A lo largo de este documento se describirán las etapas que deben seguirse como parte de una metodología de simulación aplicada para encontrar solución a sistemas reales. La metodología de simulación escogida fue desarrollada por el Comité Técnico de Metodologías de Simulación de la Universidad de Michigan- Dearborn<sup>95</sup>, la cual abarca aspectos de otras clases de metodologías, como las de Law (1993), Musselman (1992), Sadowski (1989) y Ulgen (1991), todas estas dirigidas hacia una perspectiva práctica. Dicha perspectiva es el rasgo de mayor interés puesto que a través de ella es posible obtener una mayor aproximación entre sistemas reales y modelos de simulación, lo que permite a su vez la obtención de datos más confiables, su posterior análisis y la generación de conclusiones más certeras. Lo anterior se fundamenta en cinco etapas definidas a lo largo de la metodología de simulación, las cuales son: Descripción del Problema, Diseño de Estudio, Diseño del Modelo Conceptual, Formulación de Entradas, Suposiciones, Definición y Validación de Procesos y Experimentación con el Modelo de Simulación. Cada una de ellas será explicada a continuación, dejando en claro el rol que juegan a la hora de simular sistemas reales.

### DESCRIPCIÓN DEL PROBLEMA

Es quizás la fase de mayor impacto en la totalidad del estudio de simulación puesto que una definición incorrecta incurrirá en pérdidas de tiempo y dinero. Por ello se requiere la adquisición del material de estudio necesario, así como la estimación del periodo de tiempo que abarcará dicho estudio y su respectiva documentación. Además, debe tenerse muy presente la relación costo-beneficio del análisis que se planea realizar, puesto que resulta un método muy apropiado que permite la justificación del mismo aun en una etapa tan temprana como lo es la Descripción del Problema.

Es por eso que dentro de esta fase de simulación se deben identificar todos los objetivos de estudio, así como los aspectos específicos que serán analizados y ubicados dentro de un dominio de estudio o bajo ciertos límites establecidos con miras a obtener la información requerida. Para ello deben identificarse todos los componentes del sistema real que serán plasmados en el modelo de simulación y determinar el nivel de detalle de acuerdo a las capacidades de la herramienta de simulación y a los objetivos de estudio. También es necesario estimar el tiempo que tomará realizar la simulación y los recursos necesarios para la misma teniendo en mente la relación costo-beneficio de los mismos. Para ello se debe identificar y definir tanto el ámbito del proyecto, las entidades y procesos a simular, los elementos de simulación y sus funcionalidades.

---

<sup>95</sup> "Simulation Methodology – A Practitioner's Perspective", desarrollada por Ulgen, Onur M., Black, John J., Johnsonbaugh, Betty., Klunge, Roger..

El ámbito del proyecto define los elementos de simulación y sus propiedades por defecto, que están disponible en un determinado sector. La posibilidad de disponer de diferentes ámbitos disminuye el tiempo de desarrollo de los modelos de simulación al acortar las fases de representación y configuración del modelo. Una vez se selecciona es necesario identificar las entidades (por ejemplo, estaciones inalámbricas) y procesos que se desean simular (por ejemplo, asignar prioridades distintas a tipos de tráfico distintos, etc.). La configuración de cada una de las entidades y procesos se realiza mediante la interacción, por parte del usuario, con la ventana de edición de los mismos que posee el programa. Se deben reconocer los diferentes elementos de simulación que se observan en el modelo a simular, para lo cual se identifican las funcionalidades como pueden ser la creación de entidades que entran en el sistema, destrucción de las que abandonan el sistema, o petición de las mismas, especificación de atributos y de operaciones, etc. Se han de definir los diferentes procesos donde intervienen las entidades, ya que éstas se mueven o definen su comportamiento en el sistema según los procesos en que intervienen, entendiendo por proceso el conjunto de las fases sucesivas de un fenómeno natural o de una operación asociada a una entidad. La configuración general del modelo permite definir qué elementos pueden ser modificados en tiempo de experimentación, a través del motor de simulación propio de cada simulador, y el tiempo total de la simulación. Finalmente, en esta etapa se establece un cronograma de actividades con el cual se regulará el avance del proyecto.

## **DISEÑO DE ESTUDIO**

El Diseño de Estudio es la fase en que se investigan muchos de los aspectos tratados anteriormente pero con mayor detalle, donde los aspectos técnicos y los problemas que ellos repercuten adquieren mayor peso, y se plantean todas las premisas del caso. De acuerdo a ello se elige la herramienta de simulación que resulte más favorable, y se determinan que datos serán recolectados para su eventual análisis.

Debido a que muchos aspectos del estudio o proyecto a realizar dependen de la herramienta de simulación, su selección es muy importante y deberá sujetarse a los requerimientos del mismo, entre ellos las premisas del caso, así como los requerimientos de animación -lo cual resulta muy útil como método de verificación y validación de resultados lo cual se explicará más adelante-, el nivel de detalle y sistema considerado para su estudio.

## **DISEÑO DEL MODELO CONCEPTUAL**

La estrategia de modelamiento a utilizar hace parte de la tercera fase en el proceso de simulación. Esta estrategia está relacionada con la toma de decisiones que involucran la manera en que el sistema debe ser representado en términos de las características provistas por la herramienta de simulación. De manera general, la estrategia debe enfocarse en la búsqueda de un modelo conceptual que reduzca el esfuerzo en el proceso de la simulación asegurando al

mismo tiempo que se cumplan todos los objetivos de la investigación. Esta fase consta de las siguientes etapas:

### **Decidir si se hace uso de un modelo continuo, discreto o combinado**

En un modelo continuo, las variables del sistema cambian continuamente con el tiempo y es necesario de ecuaciones diferenciales para describir su comportamiento. Por otra parte, en un modelo de eventos discretos las variables cambian de valor en puntos discretos y este se mantiene constante entre eventos diferentes. En el caso de que se haga uso de un modelo combinado se hace uso de ambos tipos de variables y es muy frecuente hacer uso de modelos discretos en sistemas continuos y viceversa. La persona encargada de realizar el modelo debe escoger el mejor modelo para el sistema de acuerdo a los objetivos del proyecto.

- **Determinar los elementos que controlan al sistema**

Algunos de los elementos (partes, gente) que hacen parte de un sistema son dinámicos, es decir, se mueven a través del sistema causando así que otras entidades o recursos reaccionen y actúen sobre ellos como respuesta de alguna señal. Otros elementos son estáticos en el sentido en que estos esperan a que las entidades o recursos dinámicos realicen alguna acción sobre ellos. El modelador debe decidir que elementos son importantes en el sistema, dependiendo de la complejidad, el tamaño, y el nivel de detalle del modelo

- **Determinar los requerimientos gráficos del sistema**

Los requerimientos de animación se discuten con detalle en esta etapa y se consideran los factores como la velocidad de ejecución de la simulación, la animación como una herramienta de verificación, la construcción de componentes estáticos y dinámicos de animación, facilidad de enlace de componentes y la disponibilidad de íconos en las librerías gráficas.

- **Determinar cómo recolectar estadísticas en el modelo**

Todas las herramientas de simulación producen reportes de salida estándar que pueden ser o no de utilidad para el usuario. El modelador debe estar en la capacidad de definir las variables, los contadores, histogramas, series de tiempo, diagramas de pastel, etc. con el fin de recolectar los datos de una manera que muestre los resultados de una manera clara y comprensible. Es muy frecuente que el modelador se vea en la necesidad de hacer uso de interfaces adicionales a la herramienta de simulación para poder recopilar toda la información de salida.

El modelo conceptual describe la representación lógico-matemática del sistema que se quiere simular y se refiere a las estrategias de modelamiento las cuales involucran las decisiones tomadas de acuerdo a la manera en que el sistema debe

representarse en términos de capacidad y de los elementos provistos por la herramienta de simulación. Debe enfocarse en asegurar que todos los objetivos del proyecto se logren y que los aspectos específicos del mismo sean investigados, a la vez que se procura obtener el mejor rendimiento de la herramienta de simulación en lo que se refiere al consumo de recursos hardware y el tiempo que tardarán las distintas simulaciones.

Dicho modelo puede ser tanto discreto como continuo, e inclusive, una combinación de ambos. En un modelo continuo el estado de las variables del sistema varía continuamente en el tiempo, mientras que en un modelo discreto sus estados variables cambian a valores discretos que permanecen constantes entre distintos eventos. Por otra parte, un modelo combinado puede utilizar las variables de ambos tipos para describir el funcionamiento del sistema de acuerdo a los objetivos del estudio y del nivel de detalle requerido. En esta fase se elige el tipo de simulación a llevar a cabo, se determinan los requerimientos gráficos del modelo que se utilizarán como herramienta de verificación y como elemento de presentación. Además, se necesita definir variables, contadores, histogramas, etc., a través de los cuales puedan apreciarse los reportes obtenidos de la simulación de manera tal que expliquen el comportamiento del modelo de simulación.

También se requiere determinar el nivel de detalle necesario con el cual se describen los componentes del sistema, lo cual depende principalmente de los objetivos del estudio, por lo que no siempre resulta sencillo la eliminación de detalles innecesarios.

## **FORMULACIÓN DE ENTRADAS, SUPOSICIONES, DEFINICIÓN Y VALIDACIÓN DE PROCESOS**

En esta fase el modelador describe en detalle la lógica operacional del sistema y realiza la recolección de datos y las tareas de análisis. Esta fase se compone de los siguientes pasos:

1. Especificar la filosofía de operación del sistema.
2. Describir las limitaciones del sistema.
3. Describir la creación y terminación de elementos dinámicos.
4. Describir el proceso en detalle.
5. Obtener las especificaciones de operación.
6. Realizar una lista con todas las suposiciones iniciales del sistema.
7. Analizar los datos de entrada.
8. Validar el modelo conceptual.

## **CONSTRUIR, VERIFICAR Y VALIDAR EL MODELO DE SIMULACIÓN**

En esta fase, el modelador hace uso de técnicas software para la construcción, verificación y validación del modelo. Del treinta al cuarenta por ciento del proceso

de la simulación debe ser realizado en esta fase, proceso que debe repetirse cuantas veces sea necesario con el fin de obtener los resultados deseados. Entre los aspectos a considerar se encuentran:

1. Limitaciones de la herramienta de simulación.
2. Construcción de diagramas de flujo en caso de ser necesarios.
3. Reutilizar el código existente al máximo.
4. Realizar pruebas de verificación del funcionamiento del sistema y de validación.
5. Hacer uso de convenciones adecuadas para la identificación de procesos y variables.
6. Utilizar macros tanto como sea posible.
7. Hacer uso de técnicas de programación estructurada.
8. Documentar el código utilizado.
9. Realizar una validación de las variables de salida y los parámetros de entrada.

## **EXPERIMENTACIÓN CON EL MODELO DE SIMULACIÓN**

En esta etapa el equipo de simulación debe decidir si investigar posibles alternativas para la simulación y analizar la posibilidad de realizar un replanteamiento de la metodología adoptada. En esta etapa el modelador identifica las variables de mayor importancia y elimina aquellas cuyo aporte a la simulación es prácticamente insignificante. Una vez hecho esto se debe realizar un estudio detallado de la experimentación que se piensa realizar, especialmente si se cuenta con un ciclo de vida largo. Entre los pasos a seguir son:

1. Identificar las variables de mayor importancia para la experimentación, es decir para la simulación.
2. Desarrollar un diseño de experimentación si es necesario.
3. Construir intervalos de referencia de los datos de salida.
4. Aplicar métodos de reducción de varianza si es requerido.
5. Análisis de resultados e identificar las relaciones de causa y efecto entre variables de entrada y de salida.

## **DOCUMENTACIÓN Y PRESENTACIÓN**

Una documentación apropiada juega un papel fundamental en el éxito de una simulación. Las tareas en esta fase se desarrollan de manera paralela al resto de fases del proceso. Entre los elementos que hacen parte de esta fase se encuentran:

1. Libro con el proyecto de simulación.
2. Documentación sobre las entradas, el código y la salida.

3. Manual de usuario.
4. Manual de mantenimiento.
5. Discusión y explicación de los resultados.
6. Recomendaciones para posibles áreas de estudio.
7. Informe final y presentación.

## Referencias Bibliográficas

- (Agredo y Gaviria, 2006) **G. Agredo** y J. Gaviria, J. “*Evaluación Experimental de la Capacidad de de IEEE 802.11b para soporte de VoIP*” Workshop I2Comm 2006. Cali, 2006.
- (Agredo y López, 2004) **G. Agredo** y G. López. “*Redes Inalámbricas y Celulares como soporte a las aplicaciones de Telemedicina*”. Telecomunicaciones & Sociedad. Agosto - 2004: Volumen 2: pág. 55 – 60
- (Alianza WiFi WMM, 2005) Wi-Fi Alliance Technical Committee Quality of Service (QoS) Task Group. “WMM™ (including WMM™ Power Save Specification)” WiFi Alliance. Versión 1.1. 2005
- (Alianza WiFi WMM, 2004) Wi-Fi CERTIFIED™ for WMM™ - *Support for Multimedia Applications with Quality of Service in Wi-Fi®* Alianza WiFi. 2004. [http://www.wi-fi.org/files/wp\\_1\\_WMM%20QoS%20In%20Wi-Fi\\_9-1-04.pdf](http://www.wi-fi.org/files/wp_1_WMM%20QoS%20In%20Wi-Fi_9-1-04.pdf)
- (Avaya, 2005) Avaya. “Preparing the WLAN for Voice” White Paper, 2005.  
<http://www.ctimanagementservices.com/documents/Preparing%20the%20WLAN%20for%20Voice.pdf>
- (Bianchi, et al, 2005) G. Bianchi, I. Tinnirello y L. Scalia. “*Understanding 802.11e Contention-Based Prioritization Mechanisms and Their Coexistence with Legacy 802.11 Stations*” IEEE Network, special issue on Wireless Local Area Networking: QoS provision & Resource Management, vol. 19, no. 4, Jul. 2005
- (Branchs, et al, 2005) A. Branchs, A. Azcorra, C. García y R. Cuevas. “*Applications and Challenges of the EDCA Mechanism*”. IEEE Network, special issue on Wireless Local Area Networking: QoS provision & Resource Management, vol. 19, no. 4, Jul. 2005
- (Camp, 2002) Ken, Camp. “IP Telephony Demystified” 1ra. Edición. McGraw-Hill Professional. 2002.
- (Choi, et al, 2003) S. Choi, J. del Prado, S. Shankar y S. Mangold. “*IEEE 802.11e Contention-Based Channel Access (EDCF) Performance Evaluation*” en Proc. IEEE Internacional Conference Computing, May. 2003.
- (Chou, 2005) J. Chou. “*Design a successful VoWLAN system*”. Wireless Net DesignLine. Disponible en: <http://www.wirelessnetdesignline.com/howto/170101775>

- (Clark, 1999) K. Clark "Cisco® LAN Switching (CCIE Professional Development)" Cisco Press. 1999
- (Coleman, 2006) D.D. Coleman y D. A. Westcott. "CWNA® Certified Wireless Network Administrator™ Study Guide" Wiley Publishing, Inc. 2006 .
- (Christos y Vergados, 2005). D. Christos y D. Vergados, "Guest Editorial" IEEE Network Special Issues on Wireless Local Area Networking: QoS Provisioning and Resource Management, vol. 19, no.4, Jul. 2005.
- (Dangerfield, et al, 2006) Ian Dangerfield, David Malone, Douglas J. Leith. "Experimental Evaluation of 802.11e EDCA for Enhanced Voice over WLAN Performance" 2nd International Workshop On Wireless Network Measurement (WinMee 2006) Abril 3, 2006. Disponible en: <http://www.winmee.org/papers/02-05.pdf>.
- (Epshtein, 2006) G. Epshtein,. "How to assure QoS over WLAN for consumer electronics", 2006. Página web disponible en: <http://www.networksystemsdesignline.com/howto/metronetworking/190500325>
- (EuroNGI, 2004) EuroNGI, "QoS in multi-service wireless networks - State of the art", 2004. Disponible en: <http://eurongi.enst.fr/archive/127/DWPJRA241.pdf>
- (Gao, et al, 2005) D. Gao, J. Cai y K. Ngi. "Admission Control in IEEE 802.11e Wireless LANs" IEEE Network, special issue on Wireless Local Area Networking: QoS provision & Resource Management, vol. 19, no. 4. Jul. 2005.
- (Garg, et al, 2003) P. Garg, R. Doshi, R. Greene M. Baker, M. Malek y X. Cheng. "Using IEEE 802.11e MAC for QoS over Wireless". En Proc. International Performance Computing and Communications Conference, 2003.
- (Gast, 2005) M. Gast. "802.11® Wireless Networks The Definitive Guide" 2<sup>nd</sup>. Ed. O'Reilly. 2005.
- (Gomez, et al, 2008) J. Gómez, S. Benavides y **G. Agredo**. "Estimación de capacidad en una red IEEE 802.11e para entornos outdoor multicelda" Workshop I2Comm 2008.
- (Guchhait, et al, 2006) A. Guchhait, R. Muralidhar y A. Bakre. "Performance of Real-Time Traffic in EDCA-Based IEEE 802.11b/g Wlans" en Proc. 1st International Conference on Communication System Software and Middleware, 2006. Comsware 2006.
- (Hurtado, et al, 2008) J. Hurtado, D. Lozano y **G. Agredo**. "Capacidad de una Red Inalámbrica en Malla, para el Transporte de Voz sobre IP y Datos en un Entorno Rural" Workshop I2Comm 2008.

- (IEEE 802.11, 1999) Estándar IEEE 802.11, "*IEEE Standard for Information technology-Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*", 1999. Disponible en: <http://standards.ieee.org/getieee802/download/802.11-1999.pdf>
- (IEEE 802.11a) Estándar IEEE 802.11a, "*Supplement to IEEE Standard for Information technology-Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-speed Physical Layer Extension in the 5 Ghz Band*", 2003. Disponible en: <http://standards.ieee.org/getieee802/download/802.11a-1999.pdf>
- (IEEE 802.11b) Estándar IEEE 802.11b, "*Supplement to IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Further Higher Data Rate Extension in the 2.4 Ghz Band*", 2003. Disponible en: <http://standards.ieee.org/getieee802/download/802.11b-1999.pdf>
- (IEEE 802.11e, 2005) Estándar IEEE 802.11e, "*IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment: Medium Access Method (MAC) Quality of Service Enhancements*", Noviembre 2005. Disponible en: <http://standards.ieee.org/getieee802/download/802.11e-2005.pdf>
- (IEEE 802.11e-Anexo D, 2005) Estándar IEEE 802.11e, "*Annex D: ASN.1 encoding of the MAC and PHY MIB*". Noviembre 2005. Disponible en: <http://standards.ieee.org/getieee802/download/802.11e-2005.pdf>
- (IEEE 802.11g) Estándar IEEE 802.11g, "*Supplement to IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Further Higher Data Rate Extension in the 2.4 Ghz Band. Amendment 4: Further Higher Data Rate Extension in the 2.4 GHz Band*", 2003. Disponible en: <http://standards.ieee.org/getieee802/download/802.11g-2003.pdf>
- (Intel, 2005) Intel. "Overcoming Barriers to High-Quality Voice over IP Deployments" Disponible en <http://www.intel.com/network/csp/pdf/8539.pdf>

- (Intel, 2004) Intel White Paper, “*Providing QoS in WLANs: How the IEEE 802.11e Standard QoS Enhancements Will Affect the Performance of WLANs*”, 2004. Disponible en: [http://www.intel.com/network/connectivity/resources/doc\\_library/white\\_papers/303762.htm](http://www.intel.com/network/connectivity/resources/doc_library/white_papers/303762.htm)
- (Jojoa, 2007) P. Jojoa. “*Guía para la elaboración del documento de Tesis de la Maestría en Ingeniería, área Electrónica y Telecomunicaciones*”. 2007
- (Kong, et al, 2004) Z. Kong, D. Tsang, B. Bensaou y D. Gao. “*Performance Analysis of IEEE 802.11e Contention-Based Channel Access*”. IEEE Journal on Selected Areas in Communications, vol. 22, no. 10, 2004.
- (Lera, et al, 2005) A. Iera, A. Molinaro, G. Ruggeri, y D. Tripodi. “*Improving QoS and Throughput in Single- and Multihop WLANs through Dynamic Traffic Prioritization*” IEEE Network, special issue on Wireless Local Area Networking: QoS provision & Resource Management vol. 19, no. 4. Jul. 2005
- (Madwifi, 2008) Multiband Atheros Driver for WiFi – MADWifi <http://madwifi.org/>
- (MadWiFi, 2007) Madwifi “*Atheros Wireless Linux Driver Users Guide*” Disponible en: <http://www.magicbox.pl/pub/magicbox/manual/madwifi-users-guide.pdf>
- (Mangold, et al, 2003) S. Mangold, S. Choi, G. Hiertz, O. Klein y B. Walke. “*Analysis of IEEE 802.11e for QoS support in Wireless Lans*” IEEE Wireless Communications, Vol. 10, no. 6, Dic. 2003.
- (Martell, 2007) J. Martell. “*Certification Shelved for Voice QoS over Wireless - Is WMM-SA Dead?*” Network Computing. 2007. Disponible en: <http://www.networkcomputing.com/channels/wireless/showArticle.jhtml?articleID=199700589>
- (Mlinarsky, 2005) F. Mlinarsky. “*Metrics and Methods Bring VoWLAN Success*”. Wireless Systems Design. Disponible en: <http://www.wsdmag.com/Articles/ArticleID/10003/10003.html>
- (Moon y Aghvami, 2003) B. Moon, H. Aghvami, “*DiffServ Extensions for QoS Provisioning in IP Mobility Environments*”, IEEE Wireless Communications, October 2003
- (NCTUns, 2008) “*NCTUns 4.0 Network Simulator and Emulator*”, 2008. <http://nsl.csie.nctu.edu.tw/nctuns.html>
- (Pattara, et al, 2004) W. Pattara-Atikom, S. Banerjee y P. Krishnamurthy. “*ADRAFT: An Adaptive QoS Mechanism to Support Absolute and Relative Throughput in 802.11 Wireless LANs*” en Proc. ACM MSWiM, 2004.

- (Prasad, 2005) A.R. Prasad y N.R. Prasad. "802.11 WLANs and IP Networking - Security, QoS, and Mobility" Artech House. 2005.
- (Selvakennedy, 2004) Selvakennedy, S. "The Impact of Transmit Buffer on EDCA with Frame-Bursting Option for Wireless Networks", 2004. Documento PDF disponible en: <http://ieeexplore.ieee.org/iel5/9433/29935/01367309.pdf>
- (Sinreich, et al. 2006) Sinreich, Henry y Johnston, Alan B. "Internet Communications Using SIP: Delivering VoIP and Multimedia Services with Session Initiation Protocol" 2da. Edición. Wiley. 2006.
- (Szigeti y Hattingh, 2005) Szigeti T, Hattingh C. "Quality of Service Design Overview". Disponible en <http://www.informit.com/articles/article.asp?p=357102&rl=1>
- (Tao, et al, 2006) Y. Tao, J. Bonnin y G. Straub "QoS issues in Wi-Fi-WMM based triple play home networks" en Proc. IEEE CNCC 2006.
- (UIT-T, G.114) Recomendación G.114 de la Unión Internacional de Telecomunicaciones, en su rama de Telecomunicaciones (*Internacional Telecommunications Union – Telecommunication, ITU-T*)
- (Ulgen, et al, 2004) Ulgen, Onur M., Black, John J., Johnsonbaugh, Betty. y Klunge, Roger. "Simulation Methodology – A Practitioner's Perspective", University of Michigan-Dearborn.
- (Wang, et al, 2007) Wang, Shie-Yuan, Chou, Chih-Liang y Lin, Chih-Che. "The GUI User Manual for the NCTUns 4.0 Network Simulator and Emulator" 2007. En: [http://nsl10.cs.nctu.edu.tw/support/documentation/NCTUns\\_4.0\\_GUI\\_Manual\\_07132007.pdf](http://nsl10.cs.nctu.edu.tw/support/documentation/NCTUns_4.0_GUI_Manual_07132007.pdf)
- (Xia y Li, 2004) Y. Xiao y H. Li. "Voice and Video Transmissions with Global Data Parameter Control for the IEEE 802.11e Enhanced Distributed Channel Access" IEEE Transactions in Parallel and Distributed Systems, vol. 15, no. 11, 2004.
- (Zhang y Zeadally, 2004) L. Zhang y S. Zeadally. "HARMONICA: Enhanced QoS Support with Admission Control for IEEE 802.11 Contention-Based Access" en Proc. IEEE RTAS, 2004.