



Escola Politècnica Superior

TRABAJO DE FIN DE CARRERA

**TÍTULO DEL TFC: Análisis de Técnicas de Gestión de Recursos Radio
para Sistemas de Comunicaciones Inalámbricas**

**TITULACIÓN: Enginyeria Tècnica de Telecomunicació, especialitat
Sistemes de Telecomunicació**

AUTOR: David Luengo Artero

DIRECTOR: Luis G. Alonso Zárata y Christos Verikoukis

FECHA: 5 de septiembre de 2005

Título: Análisis de Técnicas de Gestión de Recursos Radio para Sistemas de Comunicaciones Inalámbricas

Autor: David Luengo Artero

Director: Luis G. Alonso Zárate y Christos Verikoukis

Fecha: 5 de septiembre de 2005

Resumen

El trabajo de fin de carrera que se presenta a continuación se encuentra en el entorno de las comunicaciones inalámbricas, en concreto en las redes WLAN con estándar IEEE 802.11b. El objetivo de este trabajo es mejorar las comunicaciones en estas redes, optimizando los recursos del sistema mediante técnicas Cross-Layer. Estas técnicas consisten en el intercambio de información entre las capas OSI de un sistema de comunicaciones. En nuestro caso será la capa MAC la que reciba información de la capa física, adaptando las transmisiones en función del estado del canal radio. Hemos utilizado un protocolo de alta eficiencia para la capa MAC llamado DQRUMA (Distributed Queuing Request Update Multiple Access).

Todas las técnicas propuestas se han estudiado y adaptado al protocolo MAC utilizado con una posterior simulación.

Los resultados obtenidos confirman una mejora en los diferentes escenarios de trabajo utilizados, principalmente en términos de *Throughput*, retardo medio y capacidad del sistema en número de usuarios, además de ser de óptimo funcionamiento y fácil implementación.

Después de analizar el estudio realizado, proponemos aprovechar la eficiencia de estas técnicas en el diseño e implementación de futuras WLAN's.

Title: Radio Resource Management Techniques Analysis for Wireless Communication Systems

Author: David Luengo Artero

Director: Luis G. Alonso Zárata y Christos Verikoukis

Date: September, 5th 2005

Overview

The final university studies work presented below is found in the wireless communication environment, specifically WLAN networks based on the IEEE 802.11b Standard. The objective of this work is to improve the communication in these networks, optimising the resources of the system by means of Cross-Layer techniques. These techniques lie on the interchange of information between the OSI layers of a communication system. In this case the MAC layer will be the one which receives information from the physical layer, adapting the transmissions depending on the radio channel status. We have used a high efficient protocol for the MAC layer named DQRUMA (Distributed Queuing Request Update Multiple Access).

All the proposed techniques have been studied and adapted to the used MAC protocol with a subsequent simulation.

The results that we have obtained confirm an improvement in the different used work scenes, principally in terms of *Throughput*, average delay and system capacity in number of users, besides its optimal functioning and easy implementation.

After analysing the realised studies, we propose make use of the efficiency of these techniques in the design and implementation of future WLAN's.

AGRADECIMIENTOS

Quisiera expresar mi agradecimiento por la total atención que me han ofrecido mis directores de trabajo de fin de carrera, Luis G. Alonso Zárate y Christos Verikoukis, ofreciéndome su tiempo y resolviéndome las dudas de forma inmediata en el momento que lo he necesitado.

De la misma manera, agradecer a mi familia y amigos su comprensión y apoyo que me ha dado la fuerza necesaria para llevar a cabo el trabajo.

ÍNDICE

INTRODUCCIÓN	1
CAPÍTULO 1. INTRODUCCIÓN, MOTIVACIONES Y OBJETIVOS	2
1.1. Introducción	2
1.1.1. Introducción a las redes locales inalámbricas.....	2
1.1.2. Introducción a los principales protocolos MAC	3
1.2. Motivación del trabajo.....	3
1.3. Objetivos del trabajo.....	3
CAPÍTULO 2. PROTOCOLOS MAC PARA SISTEMAS WLAN	4
2.1. Introducción	4
2.2. MAC 802.11b.....	4
2.3. DQRUMA (Distributed Queuing Request Update Multiple Access)	5
2.3.1. Funcionamiento básico	8
2.3.2. Características fundamentales de DQRUMA.....	9
2.3.2.1. Acceso múltiple	9
2.3.2.2. Piggy-Backing (PGBK)	11
2.3.3. Algoritmos utilizados en DQRUMA	12
2.3.3.1. Algoritmo de resolución de colisiones.....	12
CAPÍTULO 3. DISEÑO CROSS-LAYER	14
3.1. Introducción	14
3.2. Técnicas Cross-Layer para sistemas WLAN.....	14
3.2.1. Técnica Cross-Layer nº 1 (CL1).....	14
3.2.2. Técnica Cross-Layer nº 2 (CL2).....	17
3.2.3. Técnica Cross-Layer nº 3 (CL3).....	19
CAPÍTULO 4. SIMULADOR DQRUMA	22
4.1. Introducción	22
4.2. Estructura general del simulador.....	22
4.2.1. Estructuras	22
4.2.2. Vectores no estructurados	23
4.2.3. Funciones	23
4.2.4. Pseudocódigo de la estructura general del simulador	25
CAPÍTULO 5. ANÁLISIS DE RESULTADOS	26
5.1. Introducción	26

5.2.	Rendimiento obtenido mediante la técnica CL1	26
5.2.1.	Resultados con tráfico de control ideal	26
5.2.1.1.	<i>Análisis del Throughput</i>	26
5.2.2.	Resultados con tráfico de control real.....	29
5.2.2.1.	<i>Análisis del throughput</i>	29
5.2.2.2.	<i>Análisis del retardo medio</i>	32
5.2.2.3.	<i>Análisis de la desviación típica del retardo medio</i>	33
5.2.3.	Conclusiones acerca del rendimiento de la técnica CL1	34
5.3.	Rendimiento obtenido mediante la técnica CL2	35
5.3.1.	Análisis del throughput.....	35
5.3.2.	Análisis del retardo medio.....	37
5.3.3.	Análisis de la desviación típica del retardo medio	38
5.3.4.	Conclusiones acerca del rendimiento de la técnica CL2	39
5.4.	Rendimiento en presencia de tráfico de voz	40
5.4.1.	Rendimiento de la técnica CL1 en presencia de tráfico de voz.....	40
5.4.2.	Rendimiento de la técnica CL3 en presencia de tráfico de voz.....	41
5.4.3.	Conclusiones acerca del rendimiento de las técnicas CL1 y CL3 en presencia de tráfico de voz.....	43
5.5.	Rendimiento con tráfico heterogéneo.....	43
5.5.1.	Análisis del retardo medio de mensaje y ráfaga	43
5.5.2.	Conclusiones acerca del rendimiento con tráfico heterogéneo	45
5.6.	Conclusiones generales tras el análisis de resultados	46
CAPÍTULO 6. CONCLUSIONES Y LÍNEAS FUTURAS DE INVESTIGACIÓN		47
6.1.	Conclusiones finales.....	47
6.2.	Consideraciones medioambientales.....	48
6.3.	Líneas futuras de investigación	48
7. REFERENCIAS		50

INTRODUCCIÓN

En los últimos años, las comunicaciones inalámbricas han tenido una fuerte influencia en el campo de las telecomunicaciones.

La necesidad de un acceso inalámbrico a redes IP con altos anchos de banda estimuló la inversión de operadores móviles para conseguir estas expectativas. Pero las dificultades que tuvieron para entrar al mercado provocaron que se desarrollaran otros sistemas móviles que ofreciesen esos servicios, como son las WLAN (*Wireless Local Area Network*).

Actualmente, las redes inalámbricas WLAN han logrado llegar a muchos tipos de mercados por su comodidad y la posibilidad de intercambio de información de voz y datos a tiempo real entre varios usuarios. Se ha introducido en la grande, mediana y pequeña empresa, en lugares públicos, en sanidad, en la docencia, e incluso en casas particulares.

Es necesario utilizar nuevos protocolos de acceso al medio más eficientes que el basado en la técnica CSMA/CA del estándar WLAN (IEEE 802.1b) ya que éste ofrece limitaciones, además de las inherentes en las tecnologías inalámbricas. El protocolo de acceso al medio de alta eficiencia utilizado en este estudio es DQRUMA (Distributed Queuing Request Update Multiple Access).

También se ha demostrado que es óptimo relacionar las informaciones de las capas OSI entre sí para obtener una eficiencia más alta en el sistema. Esta relación entre capas se denomina Cross-Layer y en nuestro caso la información de la capa física la recibe la capa de acceso al medio (MAC), permitiendo la obtención continua de información de la situación del canal radio.

El objetivo básico de este trabajo consiste en aumentar la eficiencia de las comunicaciones inalámbricas aprovechando al máximo los recursos radio del sistema. Para ello estudiaremos e implementaremos las técnicas Cross-Layer (CL), analizándolas posteriormente para demostrar el alto grado de optimización que éstas ofrecen al sistema.

CAPÍTULO 1. INTRODUCCIÓN, MOTIVACIONES Y OBJETIVOS

1.1. Introducción

1.1.1. Introducción a las redes locales inalámbricas

Las redes locales inalámbricas, también llamadas WLAN, consisten en un conjunto de aparatos electrónicos como pueden ser ordenadores, impresoras y servidores entre otros, que se comunican entre sí sin la necesidad de cable. Esta característica ofrece gran movilidad al usuario además de una sencilla instalación.

Estas redes requieren un estándar a seguir. El primero fue creado por el organismo IEEE (Instituto de Ingenieros Eléctricos y Electrónicos) en 1997, y el nombre que se le dio fue IEEE 802.11. Que especifica las características de capa física y MAC *Medium Acces Control*.

Debido al próspero desarrollo de estos sistemas, se crearon diferentes estándares en función de la normativa del lugar donde se implantaban. En Estados Unidos las variantes (b, g, a, e, h, etc.) del IEEE 802.11 y en Europa el Hiperlan/2 del organismo ETSI. La tabla 1.1 muestra las características técnicas de las tres tecnologías WLAN más extendidas actualmente.

Tabla 1.1. Características de los estándares WLAN más extendidos

Estándar WLAN	IEEE 802.11b	IEEE 802.11a	IEEE 802.11g
Organismo	IEEE (USA)	IEEE (USA)	IEEE (USA)
Año de finalización	1999	2002	2003
Banda frecuencial	2,4GHz ISM	5 GHz	2,4 GHz
Velocidad máxima	11 Mbps	54 Mbps	54 Mbps
Interfaz aire	DSSS	OFDM	OFDM

El protocolo más extendido actualmente es el IEEE 802.11b y su evolución es IEEE 802.11g. Además Hiperlan/2 tuvo una tardía entrada al mercado que le ha hecho perder puntos en la carrera comercial.

Por estos motivos hemos basado el trabajo en el estándar 802.11b. Pero informamos que las técnicas Cross-Layer y el protocolo DQRUMA utilizado funcionan también con otros estándares WLAN, gracias a su independencia con el tipo de capa física.

1.1.2. Introducción a los principales protocolos MAC

En los sistemas de comunicaciones y en concreto en las redes WLAN suele haber más de un usuario ocupando el canal simultáneamente. Por ese motivo se hace necesario un control de acceso al sistema del conjunto de usuarios.

Para lograr un control estable se utiliza un protocolo de control de acceso al medio o también llamado protocolo MAC (*Médium Acces Control*). Éste se ocupa de: gestionar los accesos de todos los usuarios evitando al máximo las colisiones y tratándolas en el caso que sucedan, dar el permiso de transmisión al usuario en función de la técnica de prioridades que utilice y de enviar los reconocimientos de paquetes de control y datos.

En concreto el protocolo MAC de alta eficiencia que nosotros utilizamos es el DQRUMA (Distributed Queuing Request Update Multiple Access).

En el Anexo hacemos una pequeña explicación de los principales protocolos de acceso al medio para WLAN.

1.2. Motivación del trabajo

El increíble aumento de las comunicaciones inalámbricas tanto en el mundo de la comunicación de datos como de voz sobre IP han provocado muchos intentos de mejorar la capa física de los sistemas, ampliando la gama de estándares para WLAN (IEE 802.11x). Se ha pretendido optimizar las tasas de transmisión *Throughput*, el retardo medio del sistema y la capacidad del sistema en número de usuarios entre otros, aunque siempre manteniendo aislada entre sí cualquier información entre capas OSI. Pero estudios recientes [4], [5] y [6], demuestran que mejorar la capa MAC recibiendo información de la capa física es un buen camino. A este intercambio entre capas se le llama Cross-Layer. Y nosotros propondremos tres técnicas Cross-Layer (o intercapa) que pueden beneficiar al sistema.

1.3. Objetivos del trabajo

El objetivo básico de este trabajo consiste en aumentar la eficiencia de las comunicaciones inalámbricas aprovechando al máximo los recursos radio del sistema. Para ello estudiaremos e implementaremos las técnicas Cross-Layer (CL), analizándolas posteriormente para demostrar el alto grado de optimización que éstas ofrecen al sistema.

Comentados los objetivos y antes de comenzar el desarrollo del estudio, informamos que a lo largo del trabajo deberemos tener en cuenta tres puntos: Primero, el estudio de técnicas Cross-Layer siempre irá adherido al protocolo MAC de alta eficiencia DQRUMA, segundo, las transmisiones se realizarán en modo paquete y tercero, el estándar utilizado es el 802.11b, aunque como antes hemos comentado, es aplicable a los demás estándares.

CAPÍTULO 2. PROTOCOLOS MAC PARA SISTEMAS WLAN

2.1. Introducción

El estándar IEE 802.11b para WLAN propone dos topologías de red diferentes, Red Ad Hoc y Red en modo Infraestructura.

A continuación mostraremos dos protocolos que gestionan el acceso al medio para redes en modo infraestructura. El protocolo estándar base que utilizamos es el MAC 802.11b con una topología de red en modo infraestructura y un acceso al medio equitativo mediante el mecanismo (DCF). En concreto el trabajo lo basaremos en un protocolo de este tipo pero además de alta eficiencia llamado DQRUMA explicado en 2.3.

2.2. MAC 802.11b

El protocolo MAC que utilizamos para el estándar 802.11b proporciona un acceso al medio equitativo utilizando dos mecanismos distintos:

- El *Distributed Coordination Function* (DCF), un protocolo de acceso de uso obligatorio basado en contención. Funciona mediante la técnica CSMA/CA que consiste en que los usuarios traten de evitar las colisiones a partir de la escucha del canal antes de realizar una transmisión. Para realizar las transmisiones de datos usa un tiempo de transmisión virtual (VTT). Este tiempo básicamente incluye:
 - Debido a que trabajamos en (TDD) *Time Division Duplex*, necesitamos dos tiempos (SIFS) para separar uplink de downlink.
 - La duración del backoff que es un valor aleatorio de tiempo que cada usuario debe esperar antes de transmitir. De esta forma se evitan más colisiones.
 - Además de la posibilidad que se produzcan colisiones y recepciones erróneas.

Y este tiempo se puede ampliar con:

- En el uplink la petición de acceso (RTS) *Request To Send*.
 - En el downlink el reconocimiento de acceso (CTS) *Clear to Send*.
-
- El *Point Coordination Function* (PCF), un protocolo basado en sondeo (polling). Este mecanismo raramente se utiliza en los dispositivos

comerciales del mercado actual y es ineficiente para transmisiones en modo paquete.

Podemos ver que este protocolo MAC (DCF) del estándar 802.11b utiliza muchos intervalos de tiempo no destinados a la transmisión de datos y que son dedicados a tráfico de control. Este tráfico de control no se considera una pérdida de tiempo porque es necesario para el correcto funcionamiento del sistema y además utilizado de forma óptima aumenta la eficiencia del sistema.

2.3. DQRUMA (Distributed Queuing Request Update Multiple Access)

DQRUMA es el protocolo de acceso al medio (MAC) de alta eficiencia en el que basamos todo el trabajo. Consiste en una extensión de S-ALOHA y es un sistema centralizado, es decir que la estación base o también llamada punto de acceso (AP) controla totalmente el sistema. Se mantiene estable y eficiente, independientemente del tamaño de paquete de transmisión, del tipo de tráfico generado (datos o voz) y del modelo de canal utilizado. También permite realizar múltiple acceso, evitar colisiones, solucionarlas en el caso que sucedan y conocer el estado del *buffer* de los usuarios durante la transmisión.

Debemos comentar que para adaptar la propuesta original de DQRUMA al estándar 802.11b tuvimos que pasar de un sistema FDD *Frequency Division Duplex*, donde el *uplink* o UL (transmisiones hechas por los usuarios y que van hacia el AP) y *downlink* o DL (transmisiones del AP hacia los usuarios) trabajaban en bandas frecuenciales diferentes, a un sistema TDD *Time Division Duplex* que significa que tanto *uplink* como *downlink* trabajan en la misma banda frecuencial y están separadas temporalmente. Por el *downlink* el AP siempre transmite en modo *broadcast*, es decir, que envía a todos los usuarios el paquete con la identidad del usuario a quien quiere informar pero el propietario de esa identidad será el único que demodule la información.

También cabe informar que nosotros sólo estudiamos la eficiencia del sistema considerando el tráfico de datos en el sentido UL y el tráfico de control en el DL y en modo *broadcast*, pero el protocolo DQRUMA funciona también con datos en DL y control en DL. El motivo por el cual hemos escogido sólo datos en UL es porque nos permite realizar múltiple acceso, en cambio datos en DL, al ser sólo el AP quien transmite, no permite utilizar la idea de *minislots*.

Los tiempos definidos a continuación son los utilizados continuamente para el funcionamiento de DQRUMA.

En el tiempo dedicado a UL se transmiten:

- **RTS (Request To Send):** Es el tiempo en que un usuario envía una petición de acceso a la tabla de accesos donde será elegido para transmitir. En el caso de tráfico de voz, llevará 2 bits de más para informar del número de paquetes de la ráfaga generada por el usuario.

Tenemos bastante con dos bits porque el máximo número de paquetes que puede contener una ráfaga de voz antes de transmitirse o perderse es de 4 paquetes.

- **DATA (paquete de transmisión de datos o voz):** Es el espacio utilizado para transmitir la información del usuario. El tamaño es variable tal como explicaremos en el apartado 1.2.2 del anexo y está estipulado por el AP. Además de transmitir DATA el usuario vía UL, también puede transmitirla la estación base vía DL, pero como hemos comentado ya, este trabajo lo enfocamos a DATA transmitida en UL.
- **PGBK (Piggy Backing):** Tiene el tamaño de un bit y se envía junto a DATA. Se utiliza para que el usuario informe de si le quedan más paquetes en el buffer por transmitir del mensaje actual. En caso positivo vale 1, en caso contrario 0. De esta forma, si es 1 el usuario seguirá transmitiendo trama a trama hasta que logre enviar el mensaje entero. En cambio si es 0 el AP sabe que ya puede eliminar al usuario de la tabla y dar permiso a otro usuario para transmitir en la siguiente trama.

En el tiempo dedicado a DL se transmiten:

- **CTS (Clear To Send):** Durante este intervalo de tiempo la estación base responde al usuario con una confirmación de su RTS.
- **XMT PERM (Permiso de transmisión):** Con este tiempo el AP envía la identidad del usuario que puede transmitir en la trama siguiente. Además de esto, indica al usuario con que tasa podrá transmitir. Para enviar esta información adicional se introducirán 2 bits de más en el XMT PERM, ya que tenemos $2^2 = 4$ tasas diferentes.
- **ACK DATA (Reconocimiento de DATA):** Con este intervalo el AP informa al usuario que ha transmitido que su paquete de información datos o voz, ha sido recibido correctamente por ella.

Y entre UL y DL nos encontramos:

- **SIFS (Short inter frame spaces):** Este espacio se dedica a separar el tiempo de UL del de DL y viceversa.

Los tiempos explicados anteriormente se distribuyen con duplexado TDD formando la trama DQRUMA tal como mostramos en la figura 2.1 y la utilización de los elementos que la componen se explica en el apartado 2.3.1.

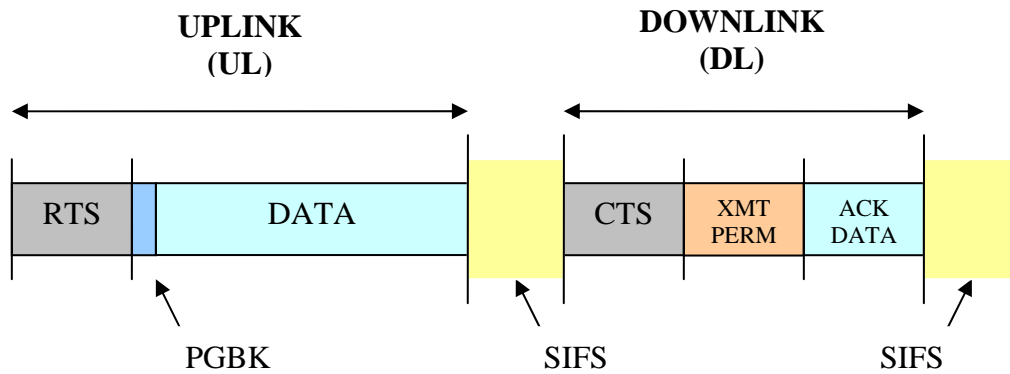


Fig 2.1. Trama de DQRUMA acceso simple

El diagrama de bloques del protocolo DQRUMA lo mostramos en la figura 2.2, en la que podemos ver los elementos del protocolo y las interconexiones entre ellos.

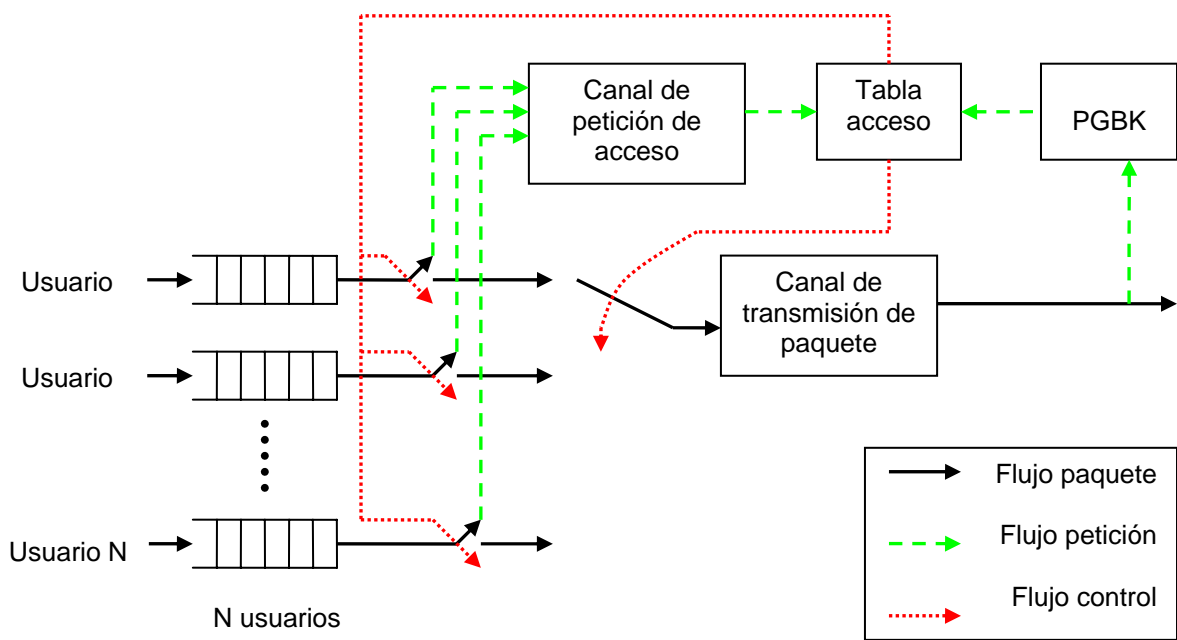


Fig 2.2. Diagrama de bloques del protocolo DQRUMA

Este diagrama consiste en un conjunto de usuarios con sus respectivos *buffers* con mensajes que quieren transmitir. Para lograrlo, primero deben acceder al sistema y ser elegidos de entre los demás usuarios para la transmisión. Por lo tanto, pasan por el canal de petición de acceso donde piden permiso para entrar en la tabla de acceso. Mediante un sistema de control de permiso (*schedulling*), el AP una vez por trama, elige a uno de los usuarios que tienen entrada en la tabla de acceso el cual transmitirá. Se le introduce en el canal de transmisión de paquete y se comprueba su PGBK *Piggy-Backing*. Si éste indica

que quedan paquetes en el *buffer* por transmitir le permitirá quedarse en la tabla de accesos, si no lo eliminará.

El funcionamiento básico de DQRUMA consiste en simple acceso (solo accede un usuario por trama) y en la transmisión de datos. Pero este mecanismo puede variar si en una determinada trama no hay datos a transmitir y estos se utilizan para múltiple acceso (acceden varios usuarios en una misma trama).

2.3.1. Funcionamiento básico

Partimos de un estado como el de la figura 2.1 con varios usuarios en el que cada uno tiene su *buffer* con o sin paquetes, y donde existen usuarios en la tabla de acceso del AP esperando a transmitir. Esto significa que nos encontramos en una situación con acceso simple y de transmisión, ya que accederá sólo uno de todos los *buffers* y transmitirá alguno de los usuarios que se encuentra en la tabla de acceso.

Teniendo esto en cuenta, en un instante determinado varios usuarios generan información que desean transmitir. Éstos envían una petición de acceso al sistema (RTS) vía UL. La petición será aceptada si no se ha producido ninguna colisión entre dos o más usuarios que han accedido a la vez o si no ha habido errores en el canal. Esta petición aceptada por el AP se introducirá en la tabla de accesos donde tiene memorizados todos los usuarios que desean transmitir, y vía DL y en modo *broadcast* avisa al que ha sido aceptado mediante un reconocimiento de petición de acceso (CTS) que indica la identidad de ese usuario. Si la petición no ha sido aceptada debido a alguno de los dos motivos comentados anteriormente, el usuario no recibirá el reconocimiento y sabrá que no ha sido aceptado. Entonces deberá utilizar un algoritmo de resolución de colisiones. Nosotros hemos utilizado el algoritmo *Harmonic Back-off*, el cual asigna una probabilidad u otra en función de la situación del usuario. Ésta se compara con una variable aleatoria de *Poisson*, y la comparación repercutirá en el tiempo que debe esperar dicho usuario antes de volver a intentar acceder, evitando que se produzca otra colisión. Una vez que el usuario se encuentra en la tabla, el AP mediante el proceso de *scheduling* escogerá a quién le toca transmitir. Esta elección puede ser más o menos eficiente en función del algoritmo utilizado. Cuando se elige qué usuario va a transmitir, vía DL y en modo *broadcast*, se transmite un permiso de transmisión (XMT PERM) con su identidad, el cual advertirá al usuario que debe transmitir en la siguiente trama. Entonces transmitirá en el tiempo dedicado a datos (DATA) vía UL. Dentro de DATA se utiliza un tiempo de bit para introducir el *Piggy-Backing* (PGBK) quien informa si el usuario que transmite tiene todavía información pendiente en su *buffer*. En caso positivo, al usuario no se le eliminará de la tabla de accesos, al contrario sí. Finalmente, cuando los datos se han recibido correctamente, se envía al usuario un reconocimiento (ACK DATA) vía DL y en modo *broadcast*, para confirmar la correcta transmisión.

El funcionamiento básico de un modo más visual es el que se muestra en la figura 2.3.

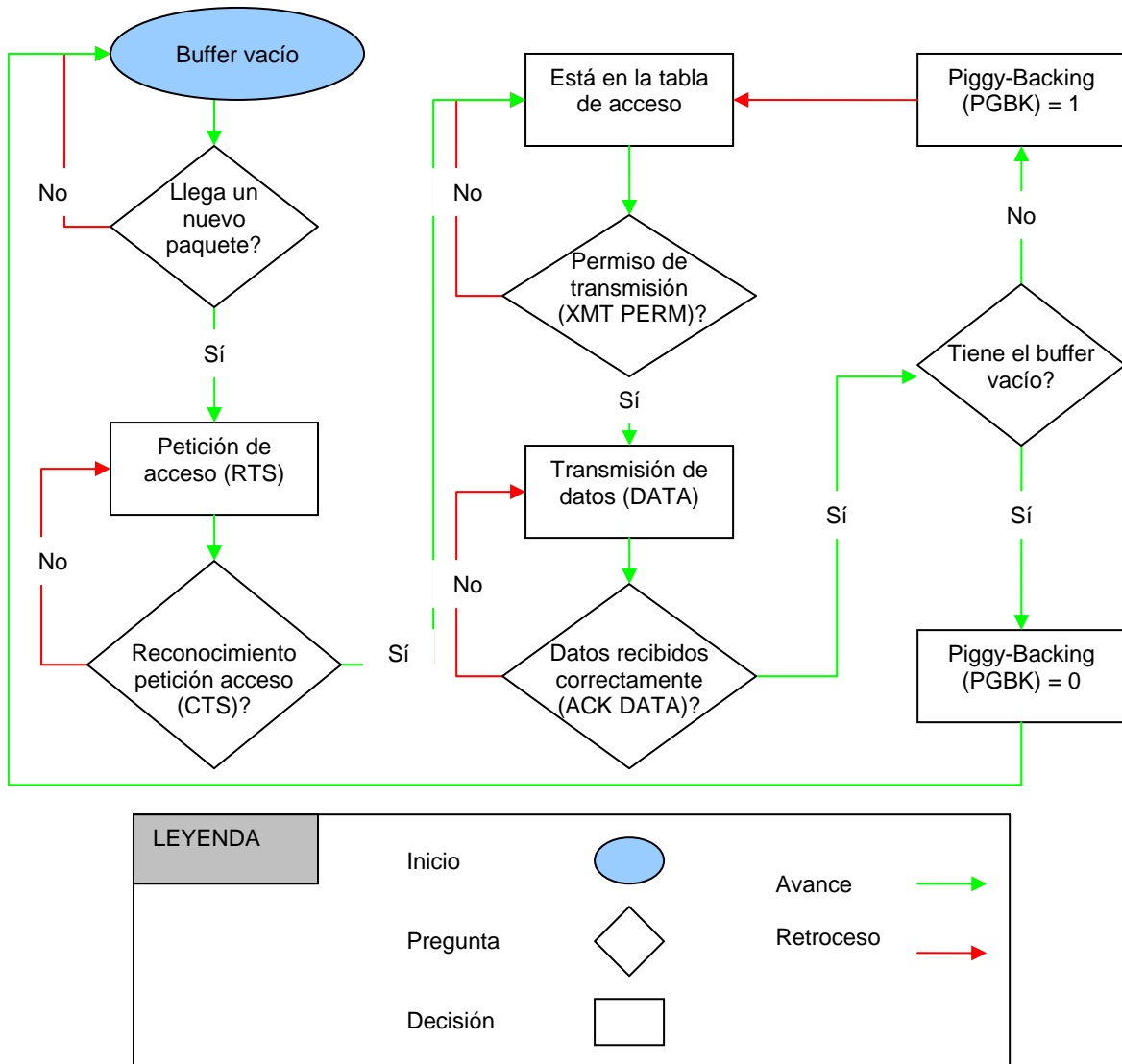


Fig 2.3. Funcionamiento básico DQRUMA

2.3.2. Características fundamentales de DQRUMA

2.3.2.1. Acceso múltiple

Ésta es una característica principal de DQRUMA y consiste en convertir el tiempo dedicado a transmisión en múltiples *minislots* cuando en una trama no se transmite información (DATA) y donde en cada una de estas ranuras se introducirá un RTS de usuarios diferentes. Podemos decir que esta característica ofrece 3 ventajas al sistema:

- Si un tiempo de trama no tiene nada a transmitir, en vez de perderse, se aprovecha con accesos.

- No logra acceder sólo uno de todos los usuarios. Con múltiple acceso lo pueden conseguir tantos usuarios como *minislots* haya. Cada mensaje se divide en paquetes de un tamaño fijado por el AP. Habrá más *minislots* cuanto más grande sea el tamaño fijado del paquete, ya que el tiempo dedicado a transmisión será más grande. El número exacto de *minislots* será el resultado de dividir el tiempo del paquete de transmisión por el tiempo que ocupa un RTS. Estos tiempos se mostrarán en la tabla 1.7 del anexo.
- Puede darse el caso que el tráfico sea bajo y ningún usuario quiera acceder. Pero en el caso que el tráfico sea alto y el acceso simple, cada vez que en una misma trama quieren transmitir varios usuarios, se producen colisiones y sólo uno de todos lo consigue. En cambio, gracias al múltiple acceso, para una misma trama obtenemos varios accesos e igual o menos colisiones. Disminuyendo así el total de colisiones del sistema.

Vistas las características principales, a continuación mostramos un ejemplo del mecanismo del múltiple acceso.

Partimos de un estado con varios usuarios en el que cada uno tiene su buffer con o sin paquetes, pero en este caso no hay usuarios en la tabla de acceso del AP esperando a transmitir. Por lo tanto se podrá dedicar el tiempo de transmisión a realizar múltiples accesos.

Suponemos un instante de $t=1$ en el que el AP informa con el XMT PERM que los dos próximos tiempos de trama se utilizarán para múltiple acceso. En un posterior instante $t=2$, hay varios usuarios con información en sus *buffers* deseando acceder al sistema para transmitir. Los usuarios conocen el número de *minislots* que caben en un tiempo de transmisión. Así que cada usuario calculará una posición aleatoria para escoger uno de los *minislots* en el que realizará su petición de acceso (RTS) vía *uplink*. Si hay dos usuarios que han escogido la misma posición, se producirá una colisión que se tendrá que tratar con el algoritmo *Harmonic Back-off* tal como antes hemos explicado. El tráfico de control en este tiempo de trama ocupa el tiempo de siempre pero no informa de nada. En el siguiente tiempo de trama $t=3$, todo el tiempo de DATA se rellenará de tantos CTS como RTS se hayan aceptado y esta vez sí que se utilizará parte del tráfico de control para avisar con el CTS el RTS que se encuentra antes de DATA y con el XMT PERM para avisar quién transmitirá en la próxima trama. El ACK ocupará el tiempo de siempre pero estará infuncional, ya que nadie ha transmitido en esta trama. Podemos observar el proceso explicado en la figura 2.4.

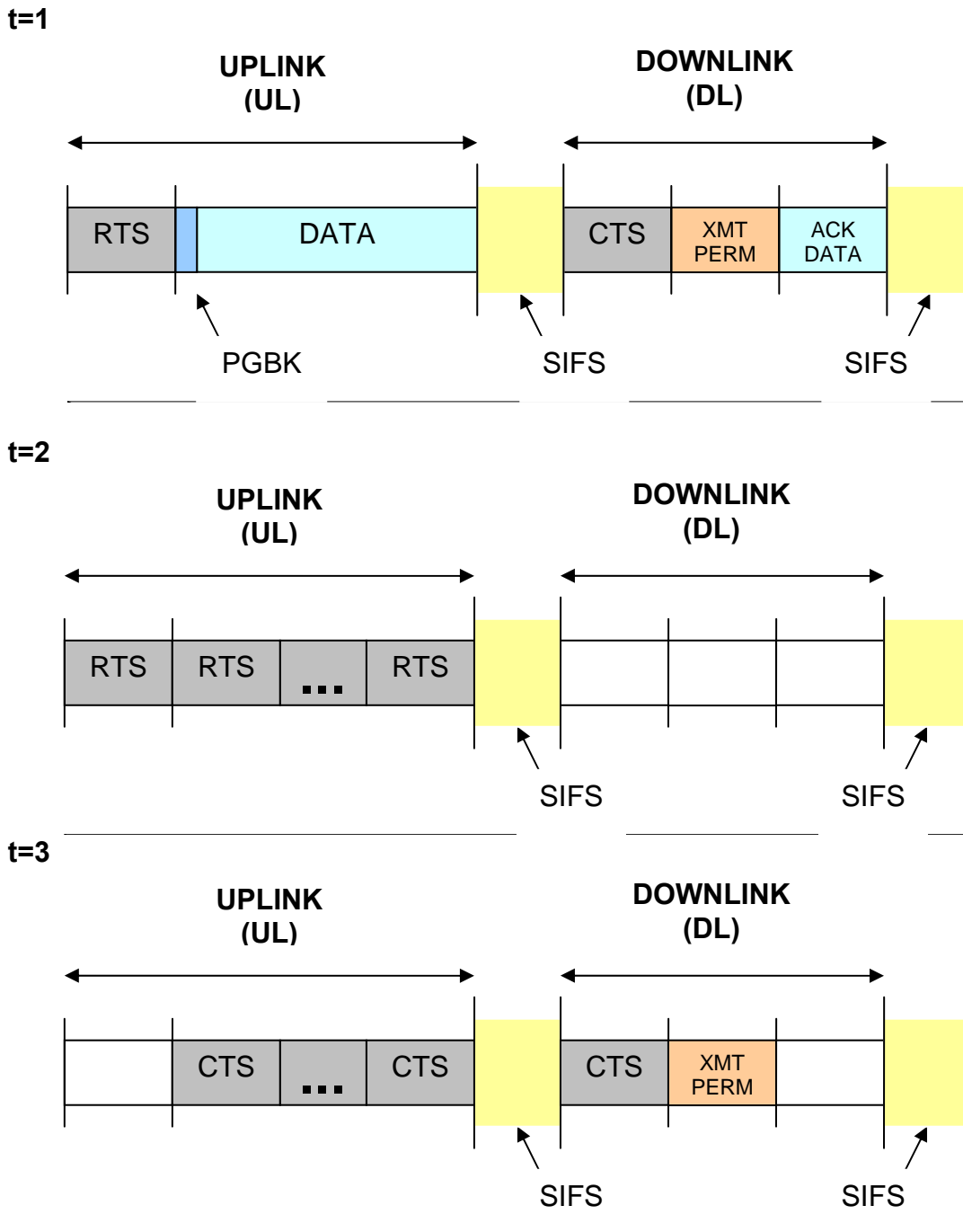


Fig 2.4. Tramas de DQRUMA en acceso múltiple

2.3.2.2. *Piggy-Backing (PGBK)*

Esta es una de las dos características principales de DQRUMA y ya ha sido explicada con detalle al definirla en el apartado 2.3.

2.3.3. Algoritmos utilizados en DQRUMA

Los dos tipos de algoritmos básicos que utiliza DQRUMA son los explicados a continuación. Cabe mencionar que aunque nosotros utilizamos un algoritmo en concreto de cada tipo, DQRUMA soporta cualquier algoritmo que sea de estos dos tipos.

2.3.3.1. Algoritmo de resolución de colisiones

El algoritmo de resolución de colisiones que utilizamos en nuestro trabajo se llama *Harmonic Back-off*. Este algoritmo utiliza cuatro tipos de probabilidades diferentes en función de la situación del usuario, de tal forma que esta probabilidad decidirá el tiempo que el usuario debe esperar antes de acceder al sistema, evitando así colisionar con otros usuarios.

- La probabilidad que se le asigna a un usuario que intenta acceder al sistema cuando la trama DQRUMA está preparada para acceso simple y siempre y cuando este usuario no haya colisionado desde el último acceso exitoso es:

$$P_{nueva} = 1 \quad (2.1)$$

- La probabilidad para aquel usuario que accede en aquella trama que el sistema permite acceso múltiple y siempre y cuando no haya colisionado desde el último acceso exitoso es:

$$P_{nueva} = \frac{1}{MS} \quad (2.2)$$

Donde MS es el número de minislots con posibles accesos que soporta el tiempo de transmisión de datos.

- La probabilidad que se le asigna a un usuario que intenta reaccéder después de haber sufrido una o más colisiones, si la trama DQRUMA está preparada para acceso simple es:

$$P_{nueva} = \frac{P_{anterior}}{1 + P_{anterior}} \quad (2.3)$$

- La probabilidad que se le asigna a un usuario que intenta reaccionar después de haber sufrido una colisión, si la trama DQRUMA está preparada para acceso múltiple es:

$$P_{nueva} = \frac{1}{MS} \cdot \frac{P_{anterior}}{(1 + P_{anterior})} \quad (2.4)$$

2.3.3.2. Algoritmo de 'Schedulling'

Este protocolo necesita un algoritmo que ordene los accesos y decida a quien le toca transmitir o lo que es lo mismo un *Schedulling*. El algoritmo que nosotros utilizamos como básico en DQRUMA para esta tarea es *Round-Robin*. Este mecanismo consiste en ordenar la tabla por orden de entrada. Por lo tanto, el usuario que antes consigue acceder, se coloca primero y será el que transmita en primera posición. El siguiente usuario se colocará detrás de este y transmitirá una vez que el primero acabe la transmisión de su mensaje y se le elimine de la tabla. Y así sucesivamente. Pero en este trabajo pretendemos demostrar que utilizando el *Schedulling* basado en técnicas Cross-Layer, es decir, que ordene teniendo en cuenta las condiciones de la capa física, puede mejorar la eficiencia del sistema. Estas técnicas las veremos en el capítulo 4.

CAPÍTULO 3. DISEÑO CROSS-LAYER

3.1. Introducción

Los sistemas inalámbricos de comunicaciones tienen un canal muy variable. En cambio, en vez de modificar los sistemas para adaptarlos y obtener más eficiencia, desde siempre se ha mantenido una estructura OSI fija con aislamiento entre sus capas. Se ha optimizado la capa física por una parte y la MAC por otra, pero nunca intercambiando información entre ellas. Si la capa MAC pudiese recibir información de la capa física y variar la forma de actuar en función del estado del canal en cada instante, obtendríamos una optimización continua del sistema. Las técnicas basadas en el intercambio de información entre las capas OSI se llama *Cross-Layer* (inter-capas). Estas técnicas también pueden funcionar para relacionar capas no contiguas como se muestra en [6]. En nuestro caso, utilizamos las capas física y MAC que contempla el estándar 802.11 y que se encuentran inmediatamente seguidas en la estructura OSI. Por lo tanto, la capa MAC recibe información de la capa física para adaptar las transmisiones del canal radio.

Debemos tener en cuenta que para poder conocer esta información adicional se requerirá añadir nuevo tráfico de control al sistema que aportará los valores de los parámetros de la capa física.

3.2. Técnicas Cross-Layer para sistemas WLAN

A continuación explicamos las tres técnicas CL utilizadas.

3.2.1. Técnica Cross-Layer nº 1 (CL1)

Sabemos que el estándar 802.11b nos limita el número de tasas o velocidades de transmisión a cuatro: 1, 2, 5.5 y 11 Mbps, también la potencia constante a 100 mW y un tamaño de paquete fijo de 0-2312 bytes para todos los usuarios elegido por el AP.

Tal como hemos comentado en la introducción, si pudiésemos obtener información física del sistema podríamos, por ejemplo, saber si un nodo tiene una SNR u otra, y como consecuencia conocer su tasa (R_b). Para saber la SNR de un usuario, el AP la calcularía mediante el RTS. De esta forma, conociendo el estado físico de todos los usuarios, se le podría dar una prioridad superior para transmitir a aquel usuario que tenga disponible una tasa superior. Esto provocaría que cada vez que transmiésemos un paquete, enviaríamos la misma cantidad de datos de paquete fijo en un tiempo menor. Así la eficiencia del sistema se optimizaría, obteniendo un Throughput del sistema superior y un retardo medio de los mensajes menor.

Esto es lo que hace la técnica CL1, obteniendo la información física de los usuarios que piden acceso, puede saber qué usuario tiene la tasa de transmisión más alta y a éste mediante el XMT PERM (vía DL y en modo *broadcast*) se le dará permiso de transmisión para la próxima trama y se le indicará a qué tasa debe transmitir tal y como ya explicamos en el apartado 2.3. En el caso que haya más de un usuario con la velocidad máxima, haremos caso al *scheduling* básico de DQRUMA *Round-Robin*, es decir, transmitirá aquel de esos usuarios que a parte de velocidad máxima, se encuentre en una posición más pequeña de la tabla de acceso, y que por lo tanto lleve más tiempo en la tabla esperando a transmitir. Este método ayudará a que el retardo del paquete del usuario que transmita sea menor, ya que si lleva más tiempo en la tabla esperando, significa que el paquete de su buffer lleva más tiempo creado. La información de quien tiene mejor posición de la tabla la tiene la propia estación base.

Una vez el usuario ha sido elegido por su velocidad y posición, hará uso de tantas tramas DQRUMA como necesite para transmitir el mensaje, con su respectivo ACK cada vez que un paquete sea recibido correctamente por el AP. Al final de cada paquete de transmisión (DATA) el usuario enviará el PGBK que informará de si ese paquete es el último del mensaje o no. Una vez que acaba de transmitir el mensaje, se le eliminará de la tabla de acceso y el AP utilizará el *scheduling* para elegir nuevo usuario a transmitir.

Por lo tanto, los parámetros que necesita conocer el AP para dar la prioridad son:

- Velocidad de transmisión de los nodos que se encuentran en la tabla de acceso (Parámetro físico que se puede conseguir a través del SNR medido)
- Posiciones en la tabla de acceso (Parámetro que tiene el AP)

Para ver más claro el funcionamiento de esta técnica la ejemplificamos a continuación. Suponemos que en la tabla de accesos han conseguido entrar 3 usuarios. En la primera posición de la tabla tenemos al usuario D con una tasa de 5,5 Mbps, en la segunda posición está el usuario L con 2 Mbps y en la tercera posición reside el usuario A con 5,5 Mbps. Utilizando la técnica CL1 nos encontramos dos usuarios que tienen la máxima tasa de entre todos los usuarios de la tabla de acceso, el usuario D y el A, pero el que transmitirá será el usuario D, ya que se encuentra en la posición 1 que es anterior a la posición 3 del usuario A. Este ejemplo lo vemos representado en la figura 3.1.

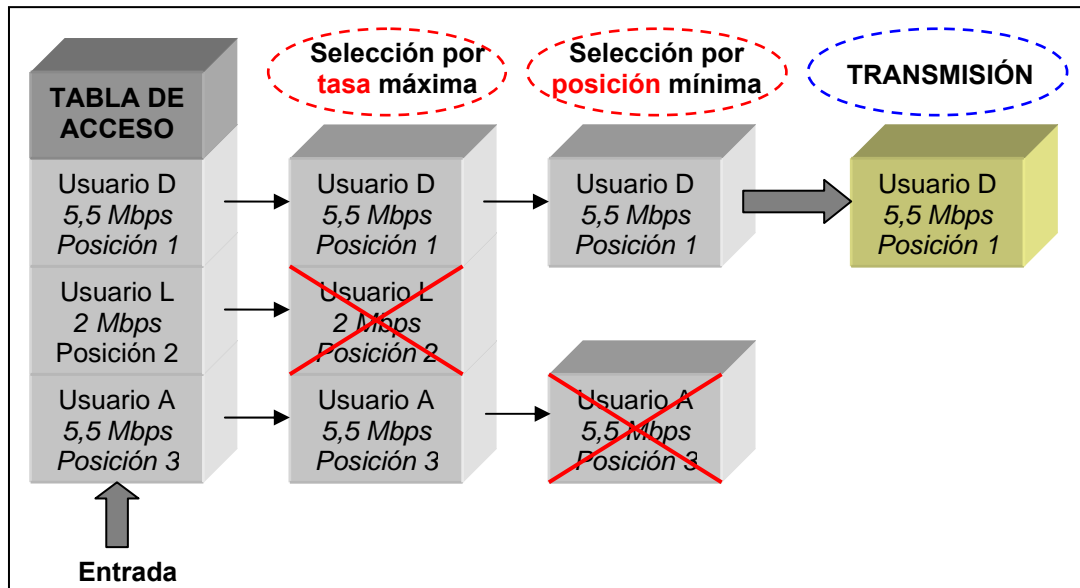


Fig. 3.1 Ejemplo de la técnica CL1

El proceso que sigue la técnica CL1 la mostramos de una forma más resumida en el pseudocódigo siguiente:

```

Para cada trama
{
   $Rb_{max}$  = Tasa máxima disponible en tabla de accesos;

  N = nº usuarios con  $Rb_{max}$ ;

  Si (N = 1) y ( $Rb_{propia} = Rb_{max}$ )
    TRANSMITIR DATOS;

  Si (N > 1) y ( $Rb_{propia} = Rb_{max}$ )
  {
    Pos_tabla_acceso_min = mínima posición de los nodos con  $Rb_{max}$ ;

    Si ( $Pos\_tabla\_acceso_{propia} = Pos\_tabla\_acceso_{min}$ )
      TRANSMITIR DATOS;
  }
}

```

En el capítulo 5 analizamos los resultados de las simulaciones para observar la eficiencia y el rendimiento que ofrece al sistema esta técnica.

3.2.2. Técnica Cross-Layer nº 2 (CL2)

Esta técnica busca un objetivo parecido a la CL1. Obtener máxima eficiencia teniendo en cuenta la velocidad de los usuarios, pero en este caso la velocidad y la posición son igual de importantes. Para obtener la prioridad con CL2 se utiliza la fórmula siguiente:

$$prioridad_i = \frac{Rb_i \text{ [en Mbps]}}{pos_tabla_acceso_i} \quad (3.1)$$

Con esta fórmula se consigue balancear la velocidad con la posición dando la posibilidad de transmitir a un usuario que lleva tiempo esperando en la tabla de acceso y que debido a una SNR baja con CL1 no transmitiría.

Con CL2 transmite el usuario que su prioridad calculada con la fórmula, dé un valor más alto. En el caso que dos usuarios coincidan en valor de prioridad porque la fórmula dé el mismo resultado, transmitirá el que tenga una velocidad más alta. De esta forma se mantiene la idea de obtener la máxima eficiencia del sistema.

Una vez elegido el usuario que transmitirá se seguirán los mismos pasos que en la técnica CL1, un XMT PERM para avisar qué usuario transmitirá y a qué tasa, entonces este usuario utilizará tantas tramas como necesite para transmitir el mensaje, con su respectivo ACK cada paquete. Después de terminar de transmitir el mensaje se le eliminará de la tabla de acceso.

La información que necesita el AP para calcular la prioridad es la misma que en CL1, ya que necesita únicamente la SNR y la tabla de accesos.

Por lo tanto, los parámetros que debe conocer el AP para dar la prioridad son:

- Velocidad de transmisión de los nodos que se encuentran en la tabla de acceso (Lo mismo que en la técnica anterior)
- Posiciones en la tabla de acceso (Parámetro que tiene el AP)

Para ver más claro el funcionamiento de esta técnica la ejemplificamos a continuación. Volvemos a tener 3 usuarios en la tabla de acceso. En la primera posición tenemos al usuario P con una tasa de 5,5 Mbps, en la segunda posición al usuario Y con 11 Mbps y en la tercera al usuario N con 11 Mbps. Con estos datos y aplicando la fórmula obtenemos, $prioridad_{usuario\ P} = 5,5/1 = 5,5$, $prioridad_{usuario\ Y} = 11/2 = 5,5$, $prioridad_{usuario\ N} = 11/3 = 3,666$. Por lo tanto tenemos 2 usuarios con prioridad máxima igual, el usuario P e Y. Pero tal como hemos dicho escogeremos para transmitir al que tenga una velocidad más alta, es decir, al usuario Y. Este ejemplo lo vemos representado en la figura 3.2.

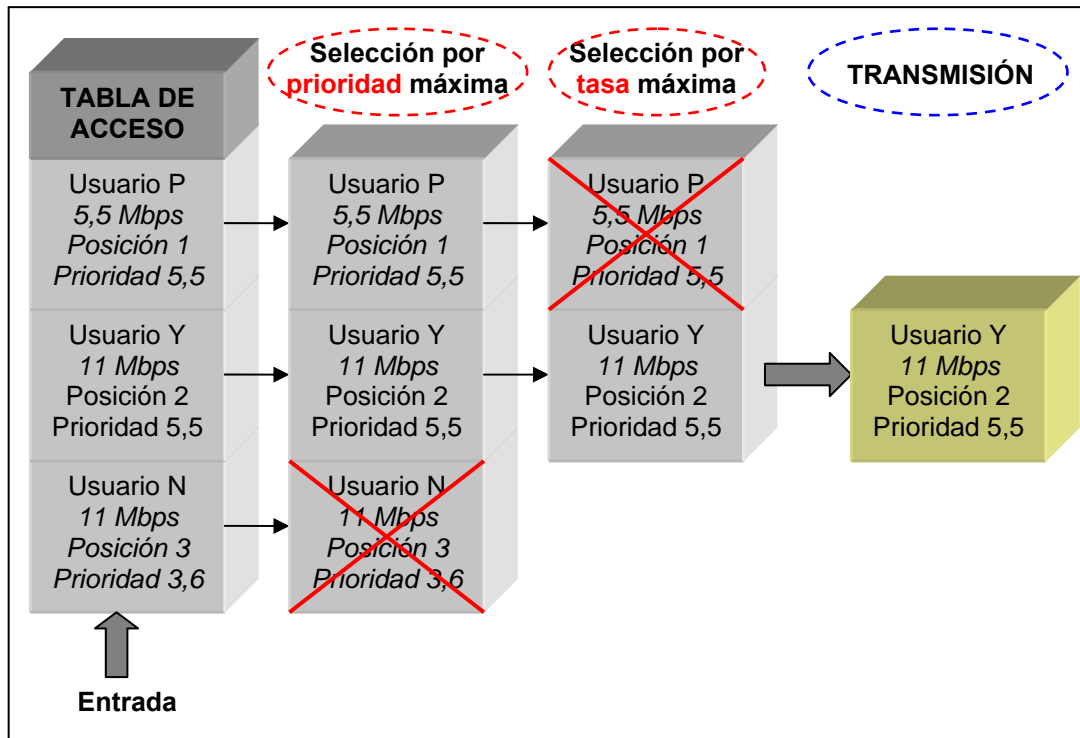


Fig. 3.2 Ejemplo de la técnica CL2

A continuación mostramos un pseudocódigo para ver de otra forma el mecanismo de la técnica CL2:

Para cada trama

```

{
  Prioridadmax = valor máximo de todos los ( $Rb_i / pos\_tabla\_acceso_i$ );

  N = nº usuarios con Prioridadmax;

  Si (N = 1) y (Prioridadpropia = Prioridadmax)
    TRANSMITIR DATOS;

  Si (N > 1) y (Prioridadpropia = Prioridadmax)
  {
    Rbmax = máxima velocidad de los usuarios con Prioridadmax;

    Si (Rbpropia = Rbmax)
      TRANSMITIR DATOS;
  }
}

```

En el capítulo 5 examinamos los resultados de las simulaciones para observar la eficiencia y el rendimiento que ofrece al sistema esta técnica.

3.2.3. Técnica Cross-Layer nº 3 (CL3)

Esta última técnica utilizada se ha diseñado para sistemas de transmisión multimedia en los que se mezcla voz (VoIP 'voz sobre IP') y datos, los dos tipos de tráfico transmitidos por paquete.

En las técnicas anteriores CL1 y CL2 la única información que los usuarios debían enviar al AP era la velocidad, ya que las posiciones las contenía la propia base. Pero para el caso de voz, además se tendrá en cuenta el número de paquetes que tiene cada usuario en su buffer.

El tráfico de voz exige unos requerimientos de calidad de servicio restringidos por el retardo y su transmisión es por ráfaga de paquetes donde se debe pedir una petición de acceso por cada paquete que forma la ráfaga. Teniendo en cuenta estas dos ideas se puede ver como un sistema con un número alto de usuarios se saturaría rápidamente de peticiones de acceso, con un posterior retardo del sistema que no cumpliría los requerimientos mínimos de calidad exigidos. Por ese motivo, en las simulaciones hemos hecho una pequeña modificación en el protocolo DQRUMA que no permite pedir un acceso por paquete de la ráfaga sino un único acceso por cada primer paquete de ráfaga generada. Es decir, que una vez un usuario ha accedido al sistema y puede transmitir, hará uso de tantas tramas DQRUMA como necesite para transmitir la ráfaga de voz, con su respectivo ACK cada vez que un paquete sea recibido correctamente por el AP. Al final de cada paquete de transmisión enviará el PGBK que informará de si ese paquete es el último de la ráfaga o no. Una vez que acaba de transmitir la ráfaga envía un PGBK igual a 0 y el AP le eliminará de la tabla de acceso.

Para conocer el número de paquetes que contiene cada usuario, se hará con el RTS.

La forma de dar la prioridad de la técnica CL3, inicialmente consiste en que transmita primero el usuario que tenga la velocidad de transmisión más alta. Pero si dos o más usuarios tienen la misma tasa, transmitirá aquel que tenga más número de paquetes, y si dos o más usuarios tienen la misma cantidad de paquetes, transmitirá aquel que se encuentre en una posición menor de la tabla de acceso. El porqué transmite el que tenga más velocidad, ya lo hemos comentado en las 2 técnicas anteriores, pero esta técnica contempla algo nuevo, el número de paquetes. El hecho que un usuario con máxima velocidad tenga un número de paquetes más alto que otro significa que este usuario podrá transmitir durante más tiempo a máxima velocidad, ya que transmitirá todos sus paquetes a tasa máxima. De esta forma el sistema asegura una alta eficiencia más prolongada. Y si dos usuarios tienen la misma velocidad y número de paquetes, parece evidente que será mejor que transmita el que lleva más tiempo esperando en la tabla de acceso para mejorar el retardo.

Por lo tanto, los parámetros que requiere conocer el AP para dar la prioridad son:

- Velocidad de transmisión de los nodos que se encuentran en la tabla de acceso (Lo mismo que en la técnica anterior)

- Número de paquetes de los nodos que se encuentran en la tabla de acceso (Parámetro físico que cada usuario debe dar a conocer al AP)
- Posiciones en la tabla de acceso (Lo mismo que en la técnica anterior)

Para ver más claro el funcionamiento de esta técnica la ejemplificamos a continuación. Suponemos cuatro usuarios en la tabla de acceso. En la primera posición de la tabla tenemos al usuario A con 2 Mbps y 12 paquetes en su *buffer*, en la segunda al usuario M con 11 Mbps y 7 paquetes, en la tercera el usuario V con 11 Mbps y 7 paquetes y en la cuarta al usuario T con 11 Mbps y 5 paquetes. Podemos observar como si priorizamos primero por velocidad, tenemos tres usuarios iguales, el M, el V y el T. Así que ahora seleccionamos por número de paquetes, de esta forma descartamos al usuario T que tiene 4 paquetes y seleccionamos a M y V que tienen 7 paquetes por igual. A continuación miramos cual de ellos tiene una posición menor en la tabla de accesos seleccionando al usuario M que se encuentra en la segunda posición que es inferior que la tercera posición de V. Por lo tanto el usuario que transmitirá en la siguiente trama será M. Este ejemplo lo vemos representado en la figura 3.3.

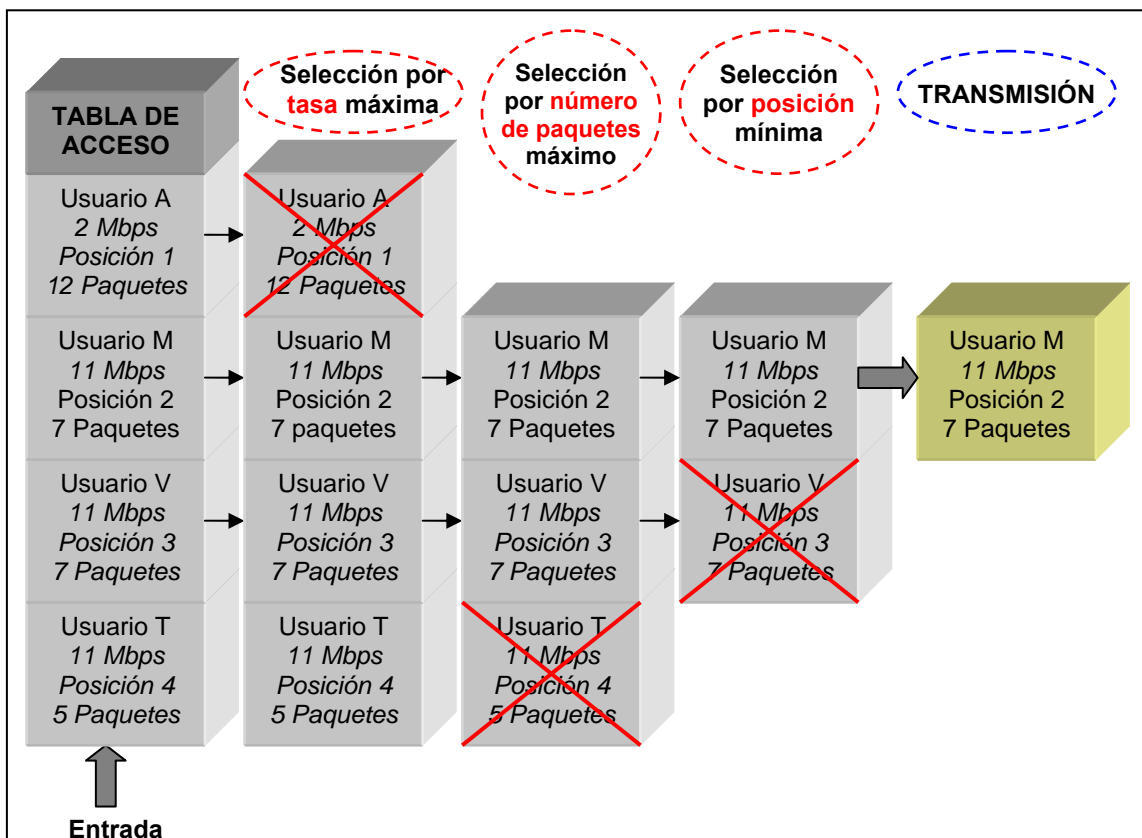


Fig. 3.3 Ejemplo de la técnica CL3

A continuación mostramos en forma de pseudocódigo el mecanismo de esta técnica:

```
Para cada trama
{
     $Rb_{max}$  = Tasa máxima disponible en tabla de accesos;

    N = nº usuarios con  $Rb_{max}$ ;

    Si (N = 1) y ( $Rb_{propia} = Rb_{max}$ )
        TRANSMITIR DATOS;

    Si (N > 1) y ( $Rb_{propia} = Rb_{max}$ )
    {
        Num_paquetes_max = Max núm de paquetes de usuarios con  $Rb_{max}$ ;

        M = nº usuarios con  $Rb_{max}$  y Num_paquetes_max;

        Si (M = 1) y (Num_paquetes_propio = Num_paquetes_max)
            TRANSMITIR DATOS;

        Si (M > 1) y (Num_paquetes_propio = Num_paquetes_max)
        {
            Pos_tabla_acceso_min = Min posición de los nodos con  $Rb_{max}$ ;
            y Num_paquetes_max;

            Si (Pos_tabla_acceso_propia = Pos_tabla_acceso_min)
                TRANSMITIR DATOS;
        }
    }
}
```

En el capítulo 5 estudiamos los resultados de las simulaciones para observar la eficiencia y el rendimiento que ofrece al sistema esta técnica.

CAPÍTULO 4. SIMULADOR DQRUMA

4.1. Introducción

En el presente capítulo describimos la estructura y principales características del simulador utilizado durante todo el estudio para evaluar la eficiencia de las técnicas CL en sistemas WLAN.

4.2. Estructura general del simulador

Para el desarrollo del programa que soporta el simulador utilizamos como inicio una versión básica del protocolo DQRUMA en C. Esta versión únicamente implementaba un sistema WATM donde un número variable de nodos comparten el canal radio mediante el protocolo básico DQRUMA. El trabajo de programación ha consistido en adaptar el protocolo básico al estándar 802.11b, prepararlo para que soporte las técnicas CL e implementar dichas técnicas.

El simulador está formado por el programa básico DQRUMA del apartado 2.3, por diferentes estructuras y funciones que implementan el escenario de trabajo descrito en el apartado 1.2 del anexo (canal, tráfico, tramas...), por las técnicas CL (CL1, CL2 y CL3) del capítulo 3 y por la extracción de resultados midiendo los parámetros relevantes mostrados en el apartado 1.3 del anexo.

El simulador realiza un número de iteraciones de simulación, cada una de las cuales corresponde a una trama DQRUMA. El número de iteraciones de simulación es variable pero se ha ajustado de tal forma que el tiempo total de simulación $T_{simulación}$ garantice que se alcanza el régimen permanente durante un tiempo significativo y los resultados obtenidos sean fiables.

4.2.1. Estructuras

- **'buff'**: Es una cola FIFO con 200 posiciones de la cual dispone cada usuario. Cada una de estas posiciones corresponderá al tamaño de un mensaje, y habrá tantas posiciones ocupadas como mensajes generados. Este tamaño informará al 'AC_RACH' del 'RACH_est'(estado de acceso) y del 'RACH_p' (probabilidad de acceso) del usuario y a la 'tabla_SNR' del estado de su *buffer* mediante el 'PGBK'. En el caso de tráfico de datos también informará del instante de generación de cada mensaje para poder calcular el retardo total del sistema. En cambio en tráfico de voz, el instante de generación se mide por paquete, ya que las restricciones en término de retardo son más altas que en datos.
- **'tabla_SNR'**: Es la estructura que contiene todos los usuarios que han logrado acceder al sistema y sus respectivos datos, es decir, su identidad, la SNR actual y el número de paquetes de su *buffer*. En

función de estos parámetros se hará la elección del usuario que puede transmitir mediante 'prioridad_tx' que utiliza la técnica de selección que toque, sea *Round-Robin* o *Cross-Layer*. Entonces se elegirá la tasa correspondiente de transmisión 'v_tx' y se le comunicará al usuario.

- **'AC_RACH'**: Controla los accesos que realizan los usuarios tanto en simple como en múltiple acceso recibiendo información de sus *buffers* tal como hemos comentado en 'buff'.
- **'on_off'**: Esta estructura es exclusiva para tráfico de voz y contiene la información necesaria para poder generar dicho tráfico. Almacena el estado actual (on u off), el instante de inicio de ese estado y el tiempo que lleva activo.

4.2.2. Vectores no estructurados

Estos vectores contienen tantas posiciones como usuarios hay e informan de los diferentes estados en los que se pueden encontrar ellos:

- **'PGBK'**: Es una variable booleana e indica si al transmitir el usuario tiene el *buffer* vacío o no. De esta forma decide si lo elimina de la 'tabla_SNR'.
- **'buffsize'**: Es una variable entera que informa del número de mensajes que tiene cada usuario en su *buffer*.
- **'RACH_est'**: También es booleana y nos indica el estado de acceso, es decir, si el usuario está en espera de acceder o no. Si el estado es negativo, puede significar o que el usuario no quiera transmitir información o que sí que quiera pero ya se encuentre en la 'tabla_SNR'.
- **'RACH_p'**: Nos indica la probabilidad que el usuario utiliza para calcular el tiempo de *Back-off* que deberá esperar antes de enviar el RTS. Es una variable decimal, ya que las probabilidades de acceso asignadas por el algoritmo de *Harmonic Back-off* pueden ser diferentes valores no enteros.

4.2.3. Funciones

- **'modelo_canal'**: Ejecuta el tipo de canal que se quiera utilizar, ya sea *Good-Bad* o *Markov*, devolviendo el valor de SNR que tiene en ese momento un usuario.
- **'prioridad_tx'**: Utiliza la técnica de selección para elegir a qué usuario de la 'tabla_SNR' le tocará transmitir, devolviendo su posición en la tabla y consecuentemente su identidad. La técnica puede ser *Round-Robin*, *CL1*, *CL2* o *CL3*.

- **'poisson'**: Nos da la probabilidad de una variable aleatoria de Poisson que nos servirá para controlar la cantidad de tráfico de datos generado y también hacer decisiones de acceso.
- **'selrach'**: Es la variable aleatoria que utilizan los usuarios para escoger en qué minislot introducirán su petición para entrar en la 'tabla_SNR' cuando el acceso es múltiple, y así evitar colisionar con otros usuarios.
- **'v_tx'**: Realiza la conversión de un valor de SNR a su correspondiente velocidad de transmisión o tasa, tal y como podemos ver en la tabla 3.3 ya mostrada.

En la figura 4.1 mostramos la interacción entre las diferentes estructuras, vectores y funciones del simulador.

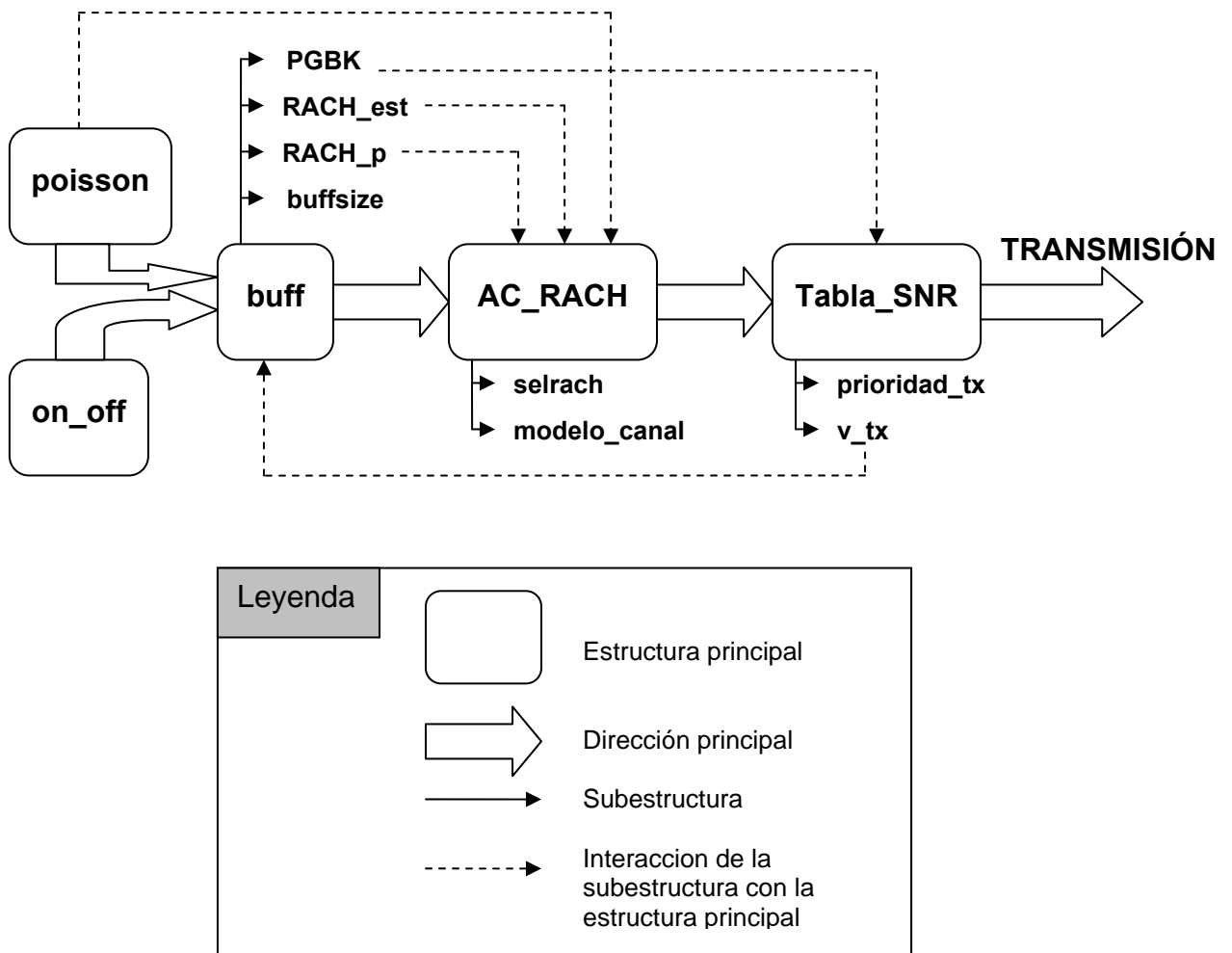


Fig. 4.1 Interacción entre los elementos del simulador

4.2.4. Pseudocódigo de la estructura general del simulador

A continuación mostraremos en forma de pseudocódigo la estructura principal del simulador donde incluimos tanto acceso simple como múltiple. Queremos informar que aunque hemos implementado tres tipos de programa que pueden trabajar de forma totalmente independiente (datos, voz y datos+voz), cada uno de ellos sigue la estructura fundamental mostrada en el pseudocódigo, pero con sus respectivas adaptaciones que le caracterizan. Para complementar con este pseudocódigo aconsejamos mirar las figuras 2.2 y 2.4 y los apartados 2.3.1 y 2.3.2. Entre paréntesis remarcamos la estructura, vector o función que utiliza cada acción. Para que se vea más claro, las estructuras las ponemos de azul, los vectores en naranja y las funciones de verde.

```
Por iteración o trama
{
  Por usuario
  {
    GENERACIÓN TRÁFICO (poisson y on_off);
    PETICIÓN ACCESO SIMPLE O MÚLTIPLE (AC_RACH, RACH_est, selrach y poisson);
    RESOLUCIÓN COLISIONES (poisson y RACH_p);
    Si (tiempo_transcurrido > tiempo_coherencia)
    {
      ACTUALIZACIÓN CANAL (modelo_canal);
    }
  }
  Si (acceso simple)
  {
    Si (trama anterior = acceso simple)
    {
      SELECCIÓN USUARIO TRANSMITE (tabla_SNR y prioridad_tx);
      CALCULO VELOCIDAD TRANSMISIÓN (v_tx);
    }
    TRANSMISIÓN;
    REORDENACIÓN BUFFER USUARIO (buff y buffsize);
    Si (PGBK=0)
    {
      ELIMINACIÓN TABLA_SNR (tabla_SNR);
      REORDENACIÓN TABLA_SNR (tabla_SNR);
    }
  }
  Si (acceso múltiple)
  {
    RECONOCIMIENTO ACCESOS;
    SELECCIÓN USUARIO TRANSMITE (tabla_SNR y prioridad_tx);
    CALCULO VELOCIDAD TRANSMISIÓN (v_tx);
  }
}
```

CAPÍTULO 5. ANÁLISIS DE RESULTADOS

5.1. Introducción

A lo largo de este capítulo mostraremos los resultados de las simulaciones realizadas con las tres técnicas Cross-Layer. El entorno de trabajo y los parámetros utilizados para valorar la eficiencia de estas técnicas está explicado en el anexo 1.

5.2. Rendimiento obtenido mediante la técnica CL1

5.2.1. Resultados con tráfico de control ideal

La idea de tráfico ideal consiste en transmitir el tráfico de control a la misma velocidad que se transmite el tráfico de datos en cada trama, es decir, que dejará de ser fijo a la mínima velocidad (1 Mbps) y ahora podrá transmitirse a (1, 2, 5.5 o 11 Mbps). Este caso es ideal, porque normalmente se transmite a la mínima velocidad para asegurar que la información destinada a control se reciba siempre sin problemas. Pero es una buena forma inicial de valorar el potencial de la técnica CL1.

5.2.1.1. *Análisis del Throughput*

Para valorar la eficiencia de esta técnica utilizando un tráfico de control ideal, mostraremos los resultados obtenidos en términos de throughput, explicado en el apartado 1.3 del anexo. Lo evaluaremos tanto en el canal Good-Bad como en Marcov y con diferentes tamaños de paquete de transmisión (100, 570 y 2312 bytes). Debemos observar un aumento del throughput cuanto más grande sea el tamaño del paquete y cuando utilicemos la técnica CL1.

Debemos tener en cuenta que a partir de ahora cuando en una gráfica mostremos que se utiliza DQRUMA sin ningún complemento posterior, estamos hablando del DQRUMA básico con *schedulling Round-Robin*, es decir, sin ninguna técnica Cross-Layer aplicada.

A continuación mostramos el throughput obtenido en un sistema de 20 usuarios de datos con el Canal Good-Bad en la figura 5.1 y con el canal Marcov en la figura 5.2.

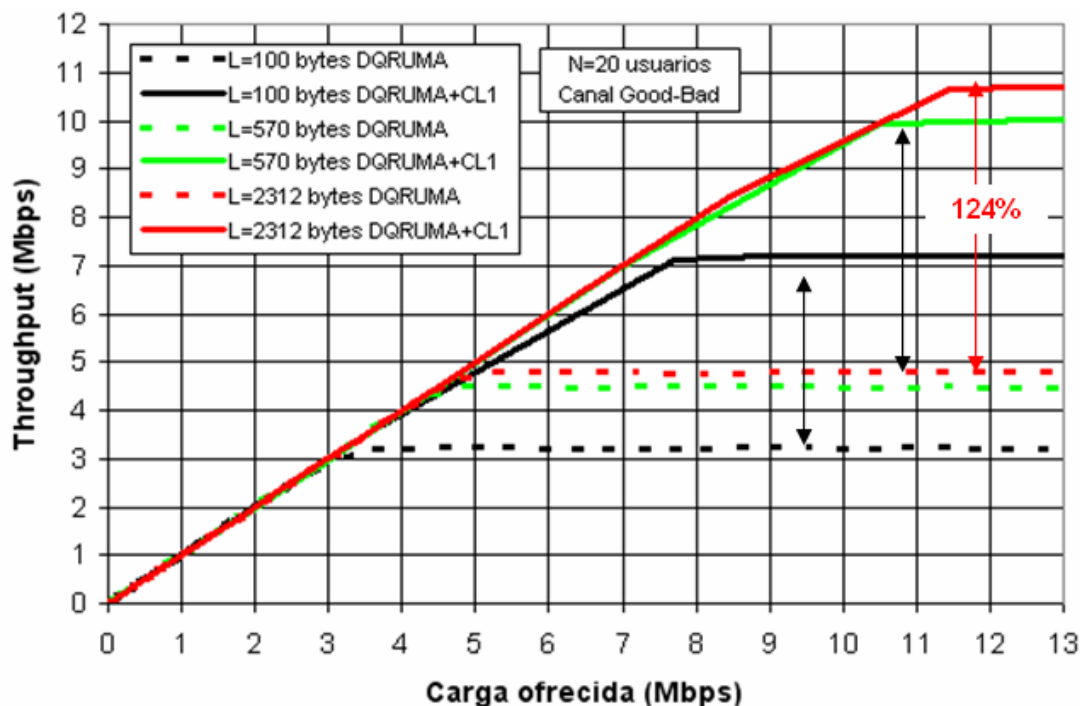


Fig. 5.1 Throughput alcanzable con tráfico de control ideal y canal Good-Bad

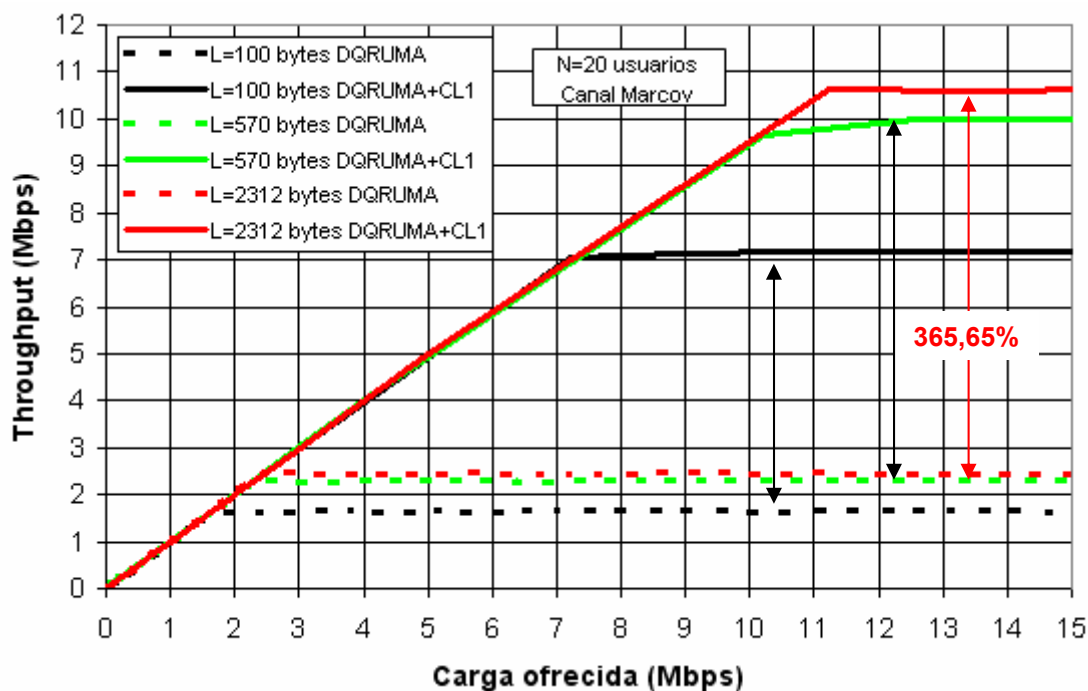


Fig. 5.2 Throughput alcanzable con tráfico de control ideal y canal Markov

Antes de comenzar el análisis, comentamos que las flechas muestran cualquier tipo de ganancia ofrecida al sistema en términos de throughput, retardo medio y desviación típica del retardo medio. En concreto la flecha roja indica la ganancia máxima de la gráfica.

Inicialmente podemos observar como en ambas gráficas con modelos de canal diferentes, al aumentar el tamaño de paquete, el throughput es más alto y la ganancia obtenida con CL1 también. El throughput aumenta debido a que el tiempo dedicado a datos (variable) es mayor respecto al de control (fijo) cuanto mayor sea el paquete de transmisión de datos, y esto se resume en que la eficiencia de las tramas ρ_i (expresión 1.1 del anexo) es mayor. Y la ganancia obtenida mediante CL1 aumenta porque si el tamaño de paquete es más grande, el tiempo dedicado a datos también lo será (como acabamos de comentar), y de este modo una vez escogida la tasa de transmisión más elevada, aseguraremos durante un tiempo superior esta velocidad máxima.

A continuación podemos ver que la diferencia entre el protocolo DQRUMA con y sin CL, es muy alta. En el caso del tamaño de paquete $L=2312$ bytes en el canal Marcov pasamos de 2,44 Mbps de máximo throughput alcanzable con DQRUMA básico a 10,63 Mbps con CL1. Por lo tanto se consigue una ganancia del **335,65%** sobre el throughput máximo, gracias a esta técnica. Esta misma observación para el caso de Good-Bad se convierte en una ganancia del 124%.

Por otra parte la diferencia entre los dos gráficos nos muestra que para un canal más hostil como es Marcov, donde los valores de DQRUMA básico son más bajos que en Good-Bad, la técnica CL1 responde con una ganancia superior.

Vistas las observaciones de los gráficos debemos comentar que al utilizar un modelo de canal u otro, cambia el porcentaje de utilización de las 4 tasas disponibles. Es decir que en función de si un canal es capaz de utilizar más o menos veces la velocidad máxima, o lo que es lo mismo, si la utilización de las tasas de transmisión σ_i (expresión 1.1 del anexo) es superior para la tasa de 11 Mbps que para las otras tres o no, la técnica CL1 será más o menos óptima. Los porcentajes de utilización de cada velocidad en función del canal utilizado los veremos cuando tratemos el tráfico de control real en el apartado (5.2.2), porque consideramos más clara su demostración allí.

Debemos tener en cuenta que la eficiencia de la técnica CL1 depende del número de usuarios simultáneos en el canal. Es decir que cuantos más usuarios haya mejor funcionará la técnica, ya que el AP mediante el *schedulling* tendrá más posibilidades de encontrar en la tabla de acceso una tasa mayor. Cuando hay pocos usuarios no tienen porque tener alguno de ellos la tasa máxima disponible, y esto haría que el throughput máximo alcanzable del sistema disminuyese. A esta ganancia se le llama 'ganancia por diversidad multiusuario'. En la figura 5.3 mostramos esta ganancia valorada para $L=2312$ y para el canal Marcov.

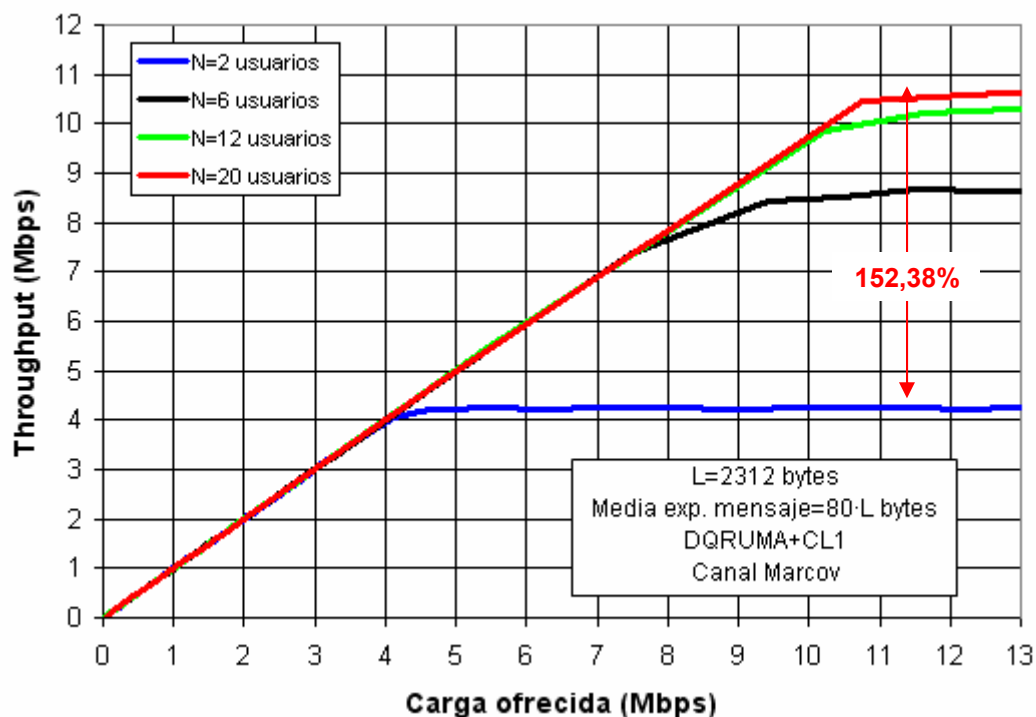


Fig. 5.3 Ganancia por diversidad multiusuario con tráfico de control ideal y canal Marcov

El hecho que la media exponencial del mensaje sea de $80 \cdot L$ bytes en vez de $10 \cdot L$ bytes como se estipuló en la tabla 1.5 del anexo, es debido a que al utilizar en la gráfica 2 usuarios, si no aumentábamos el número de paquetes por mensaje generado, la carga del sistema era muy baja y no se podía observar el valor para estos 2 usuarios.

Podemos observar la ganancia obtenida al pasar de 2 a 20 usuarios (**152,38%**).

5.2.2. Resultados con tráfico de control real

En el apartado anterior suponíamos que la velocidad de transmisión que utilizaba el tráfico de control era la misma que el de datos, es decir, caso ideal. Pero a partir de ahora, en los próximos apartados, tanto cuando hablemos de tráfico de datos como de voz, siempre utilizaremos la velocidad mínima (1 Mbps) para transmitir el tráfico de control que se corresponde con el caso real.

5.2.2.1. Análisis del throughput

A continuación mostramos el throughput con tráfico real y con los dos tipos de canales vistos anteriormente. El canal Good-Bad en la figura 5.4 y Marcov en la 5.5.

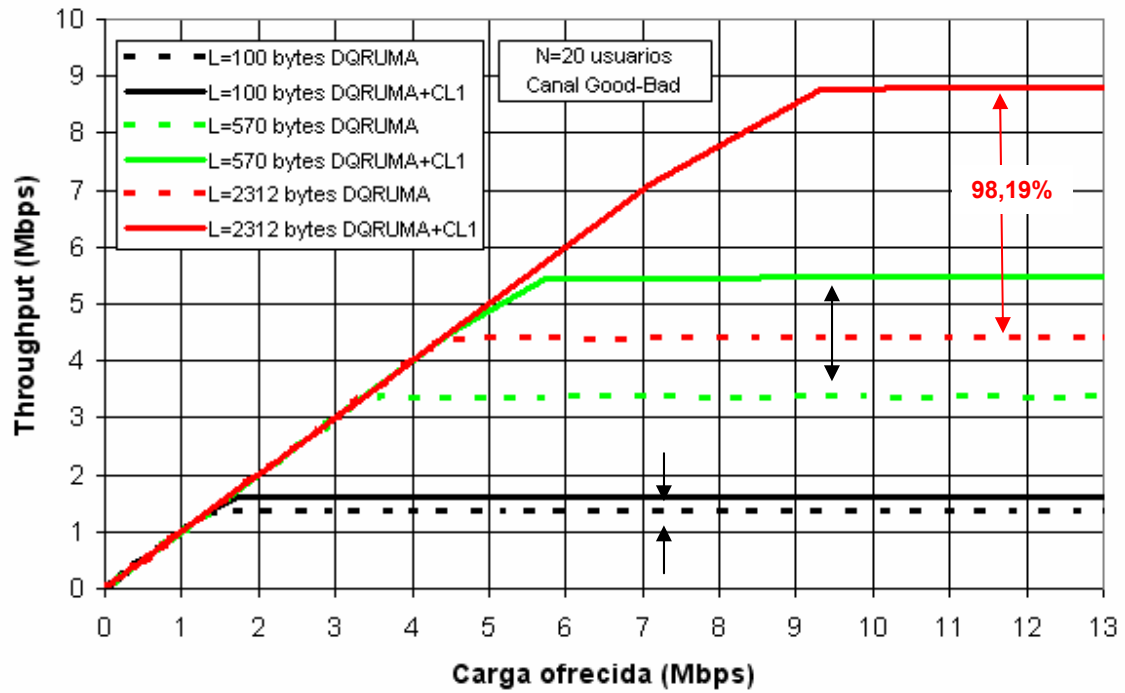


Fig. 5.4 Throughput alcanzable con tráfico de control real y canal Good-Bad

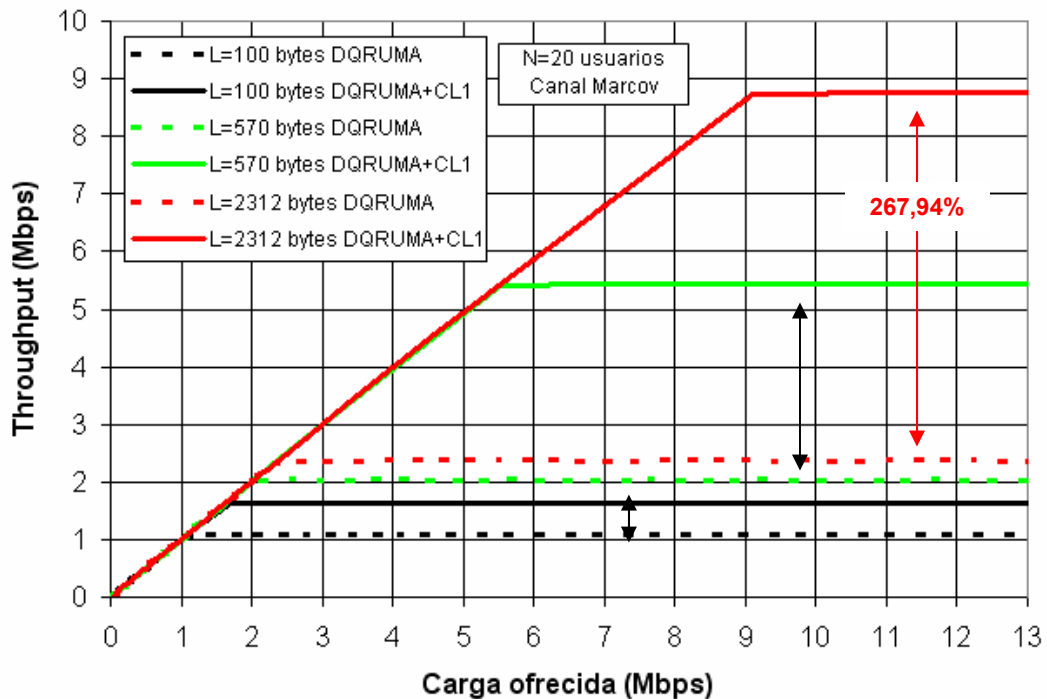


Fig. 5.5 Throughput alcanzable con tráfico de control real y canal Marcov

Por una parte podemos observar que las dos gráficas mantienen respectivamente mucha similitud en características con las dos calculadas para tráfico de control ideal. Es decir que con un mayor tamaño de paquete obtenemos un mayor throughput y que con CL1 se obtiene también una cota

superior, llegando a obtener por ejemplo en el canal Marcov y para L=2312 bytes una ganancia del **267,94%** con CL1 sobre el máximo alcanzable con DQRUMA básico. Con el canal Good-Bad obtenemos una ganancia máxima de un 98,19%.

Pero por otra parte vemos que existen dos diferencias fundamentales. Primero, para L=100 bytes en Good-Bad vemos que la ganancia mediante CL1 es mínima, esto es debido a dos factores: la poca eficiencia que representan 100 bytes de datos respecto al total de tráfico y la utilización de un tráfico real. Segundo, que el throughput máximo alcanzable disminuye respecto al tráfico ideal, esto es debido a que todo el tráfico de control se transmite a mínima tasa (1 Mbps), por lo tanto, aunque el porcentaje de utilización de la tasa máxima (11 Mbps) en tráfico de datos sea del 100%, no obtendremos nunca un throughput máximo total (control+datos) de 11 Mbps. Pero cabe decir que cuanto más alto sea el porcentaje de utilización de la tasa máxima, más throughput obtendremos.

A continuación mostramos en la figura 5.6 los diferentes porcentajes de utilización de cada tipo de tasa en función del modelo de canal y de la técnica *scheduling* utilizada (DQRUMA básico o CL1). Debemos decir que los porcentajes de utilización de las tasas son independientes del tamaño de paquete de transmisión, ya que el tamaño no influye en si un usuario en el momento de transmitir tiene un canal radio mejor o peor y por consiguiente una tasa superior o inferior. Sólo dependen del canal radio y lo asiduo que sea la elección de tasas superiores mediante el *scheduling*. Informamos que estos son los porcentajes máximos adquiridos por cada técnica, es decir, han sido calculados cuando el throughput es el máximo alcanzable para cada canal y técnica mostradas en el gráfico.

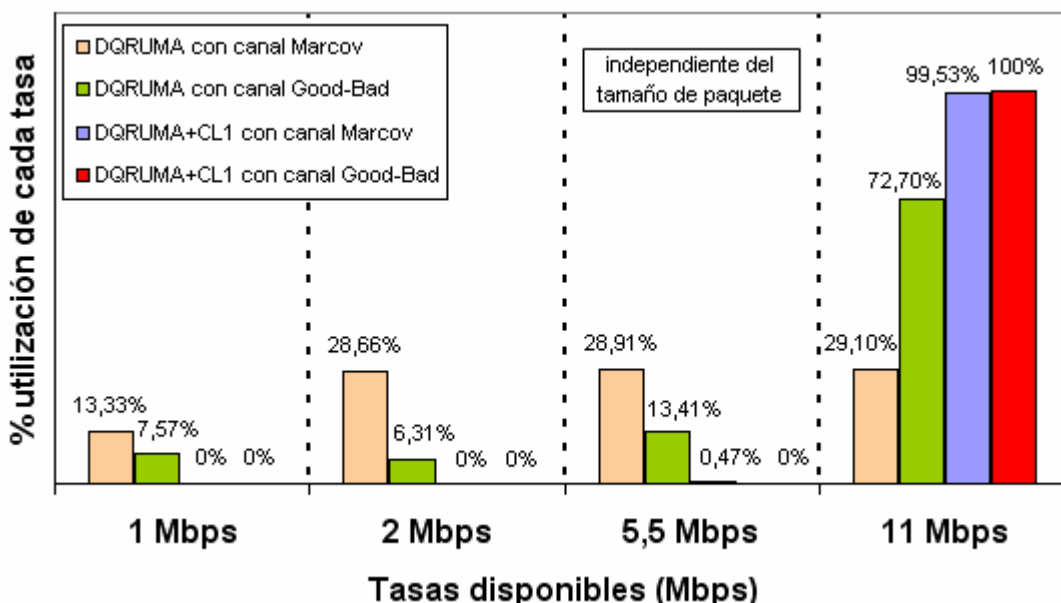


Fig. 5.6 Porcentaje de utilización de cada tipo de tasa para cada modelo de canal y técnica *scheduling*, con throughput máximo y tráfico real

Vemos que los valores de los porcentajes se corresponden en proporción con el throughput de las figuras 5.4 y 5.5. Esto es debido a la relación directa entre el porcentaje de utilización de cada trama con el throughput. Podemos ver como los resultados para el caso de Good-Bad con CL1 son prometedores, pudiendo obtener a pleno rendimiento, un porcentaje de utilización de la tasa máxima de hasta el 100%.

Finalmente en las figuras 5.4 y 5.5 podemos ver como en ambos canales y para los diferentes tamaños de paquete, los valores de DQRUMA básico obtenidos en ambas gráficas son diferentes. En cambio la versatilidad de la técnica CL1 permite obtener un valor máximo de throughput igual en ambos modelos de canal y para los diferentes tamaños de paquete, aplicando una ganancia inferior o superior en función del canal y tamaño utilizado.

5.2.2.2. Análisis del retardo medio

La técnica CL1 consigue que el sistema transmita la misma cantidad de paquetes en menos tiempo y por ese motivo, tanto los mensajes de los usuarios con alta tasa como los de baja esperan menos en el *buffer* antes de ser transmitidos, y de esta forma el retardo medio total del sistema baja. En las dos figuras siguientes mostramos el retardo en los dos tipos de modelo de canal, en la figura 5.7 con el canal Good-Bad y en la 5.8 con Marcov.

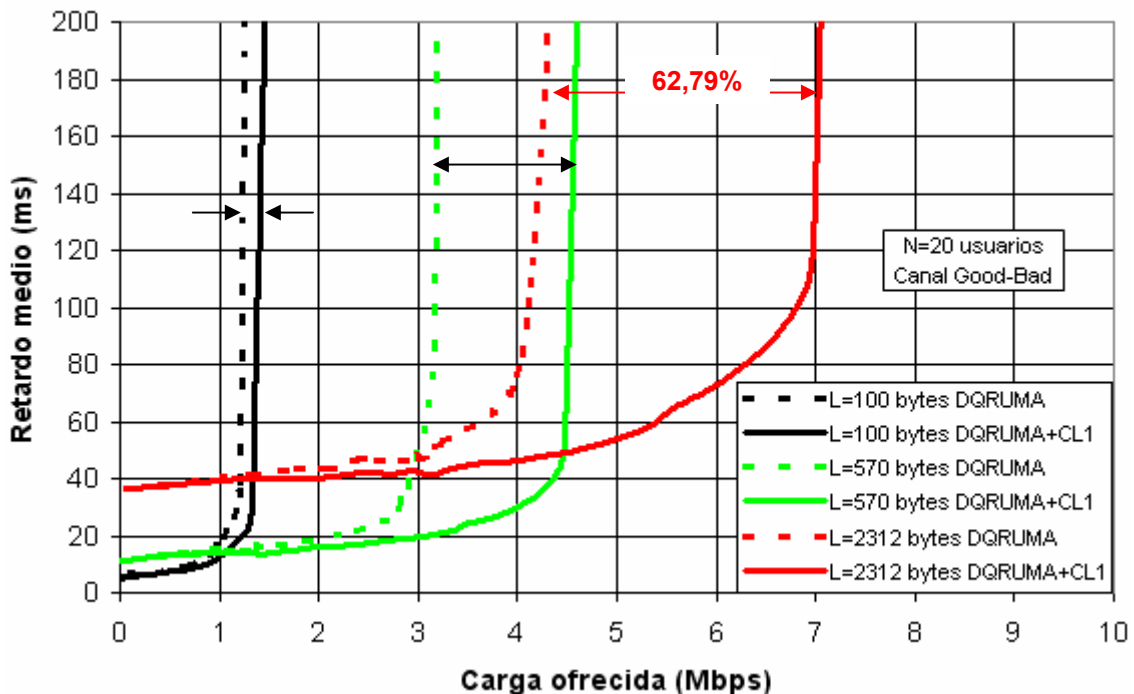


Fig. 5.7 Retardo medio con tráfico de control real y canal Good-Bad

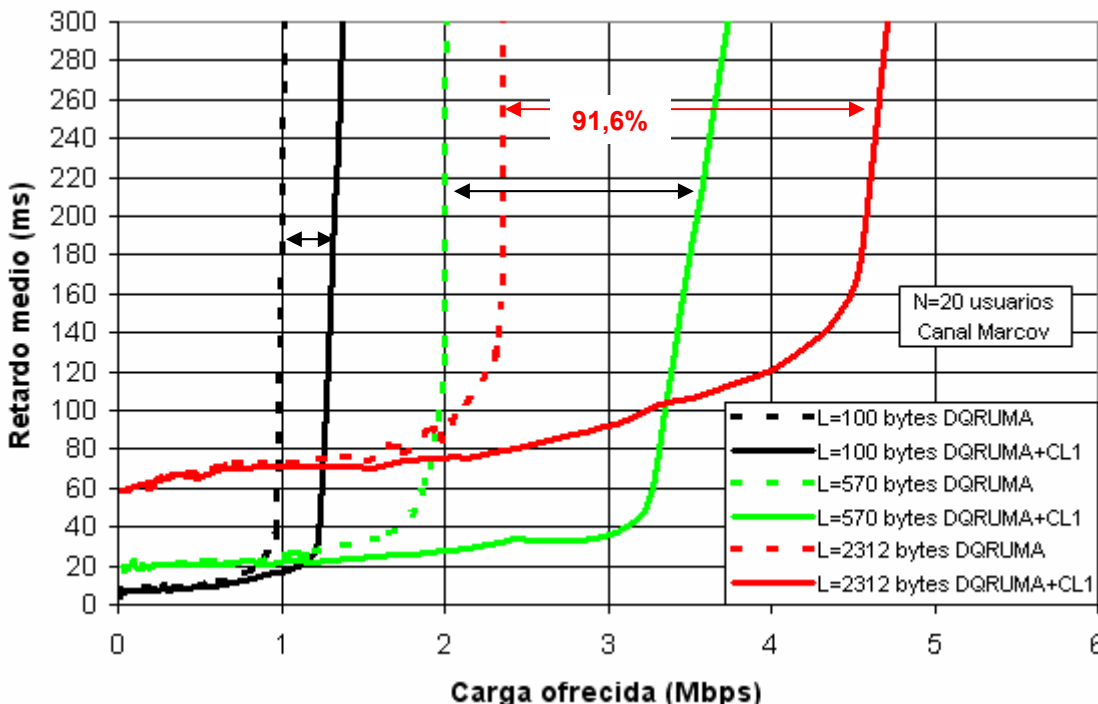


Fig. 5.8 Retardo medio con tráfico de control real y canal Marcov

Se puede observar como el retardo medio es menor en el canal Good-Bad, por lo tanto corroboramos que el canal Marcov es más hostil, como sucedía con el throughput.

Además vemos que el retardo de DQRUMA básico aumenta con el tamaño de paquete. Es decir que cuanto mayor sea el paquete, peores resultados de retardo obtendremos con esta técnica. Esto es debido a que los usuarios deben esperar más tiempo porque los paquetes a transmitir de todos los usuarios son más largos y tardan más en ser recibidos

También se observa que los valores de las dos gráficas con CL1 son mejores que en DQRUMA básico, además de soportar más carga para un mismo valor de retardo medio. En concreto el canal Good-Bad es capaz de soportar para un mismo valor de retardo medio, un 62,79% más de carga mediante CL1 y el canal Marcov un **91,6%**

5.2.2.3. *Análisis de la desviación típica del retardo medio*

En el análisis de este parámetro observaremos como la técnica CL1 vuelve a dar un buen resultado, es decir que podemos considerar que esta técnica estabiliza el retardo consiguiendo que se desvíe menos y que además para un mismo valor de desviación CL1 soporte cargas más altas que DQRUMA básico. Podemos ver todo esto en la figura 5.9.

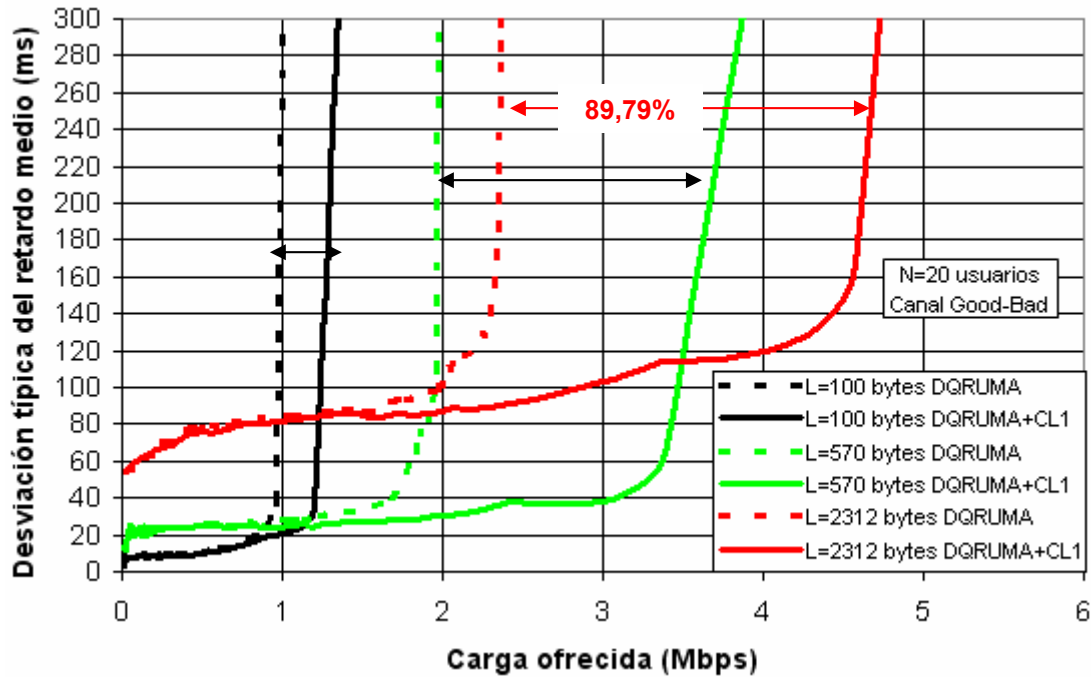


Fig. 5.9 Desviación típica del retardo medio con tráfico de control real y canal Good-Bad

Efectivamente, los resultados confirman las premisas comentadas en el párrafo anterior soportando un 89,79% más de carga que DQRUMA básico. Además podemos agregar que cuanto mayor es el tamaño de paquete mayor es la ganancia de CL1 respecto a DQRUMA básico.

5.2.3. Conclusiones acerca del rendimiento de la técnica CL1

Después de analizar todos estos parámetros de la primera técnica Cross-Layer, podemos decir que se han cumplido los objetivos marcados. Ya que el sistema obtiene, gracias a esta técnica, una mayor eficiencia y un mejor aprovechamiento de los recursos de éste. A continuación resumimos los resultados analizados:

- Se consiguen ganancias de throughput para ambos canales analizados y para cualquier tipo de tamaño de paquete de transmisión.
- Incluso cuando el sistema utiliza el tráfico de control real, consigue obtener valores mucho más altos que con el DQRUMA básico (incrementos de hasta un 267,94% para el canal Marcov).
- CL1 también mejora el retardo medio de los mensajes respecto DQRUMA básico en ambos canales analizados y para cualquier tipo de paquete, disminuyendo el tiempo de transmisión de cada paquete y por

lo tanto el retardo total del sistema. Se consigue soportar un 91,6% más de carga mediante CL1 en el canal Marcov.

- Se obtienen unos mejores resultados de desviación típica del retardo medio de los paquetes mediante esta técnica respecto a DQRUMA básico. Aumentos de carga del 89,79%.

Después de analizar estas cuatro mejoras concluimos que la técnica CL1 mejora de forma significativa la eficiencia de los sistemas WLAN y por ese motivo proponemos utilizar esta técnica en futuros diseños de redes inalámbricas.

5.3. Rendimiento obtenido mediante la técnica CL2

En este apartado mostraremos el análisis hecho sobre la técnica CL2 en los mismos dos modelos de canal utilizados para CL1 (Good-Bad y Marcov) y para dos tipos de tamaño de paquete (1000 y 2312 bytes).

5.3.1. Análisis del throughput

A continuación mostramos el throughput con tráfico real, con los dos tamaños de paquete comentados y con los dos tipos de canales vistos anteriormente. El canal Good-Bad en la figura 5.10 y Marcov en la 5.11.

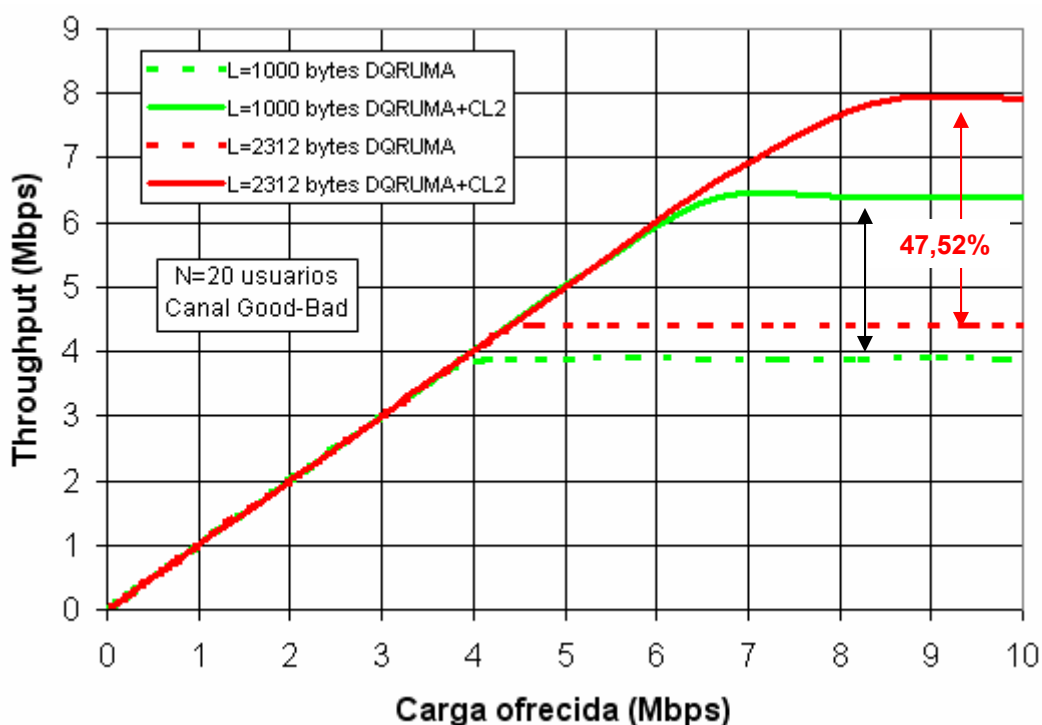


Fig. 5.10 Throughput alcanzable con tráfico de control real y canal Good-Bad

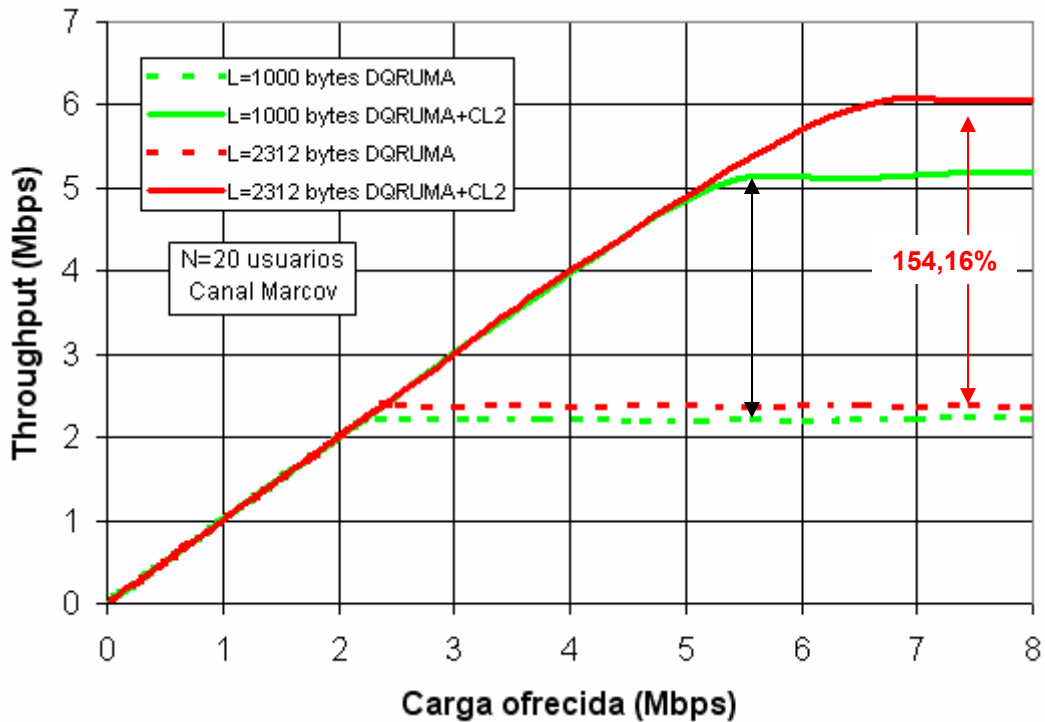


Fig. 5.11 Throughput alcanzable con tráfico de control real y canal Marcov

Por una parte podemos observar que con un mayor tamaño de paquete obtenemos un mayor throughput.

Por otra parte el canal Good-Bad permite obtener cotas mayores de throughput que Marcov, es decir que funciona mejor cuanto más hostil sea el canal.

En términos de ganancia de CL2 sobre DQRUMA básico, la diferencia entre ambos canales es más pequeña que la vista con CL1. Cuantificando esta ganancia obtenemos, por ejemplo, en el canal Marcov y para L=2312 bytes, una ganancia del 154,16% de CL2 sobre el máximo alcanzable con DQRUMA básico. En el canal Good-Bad conseguimos un 47,52% de ganancia.

En la figura 5.12 mostraremos los diferentes porcentajes de utilización de cada tipo de tasa en función del modelo de canal y de la técnica *scheduling* utilizada (DQRUMA básico, CL1 o CL2) tal y como hemos hecho en CL1, pero en este caso como vemos, añadimos el efecto de la técnica CL2.

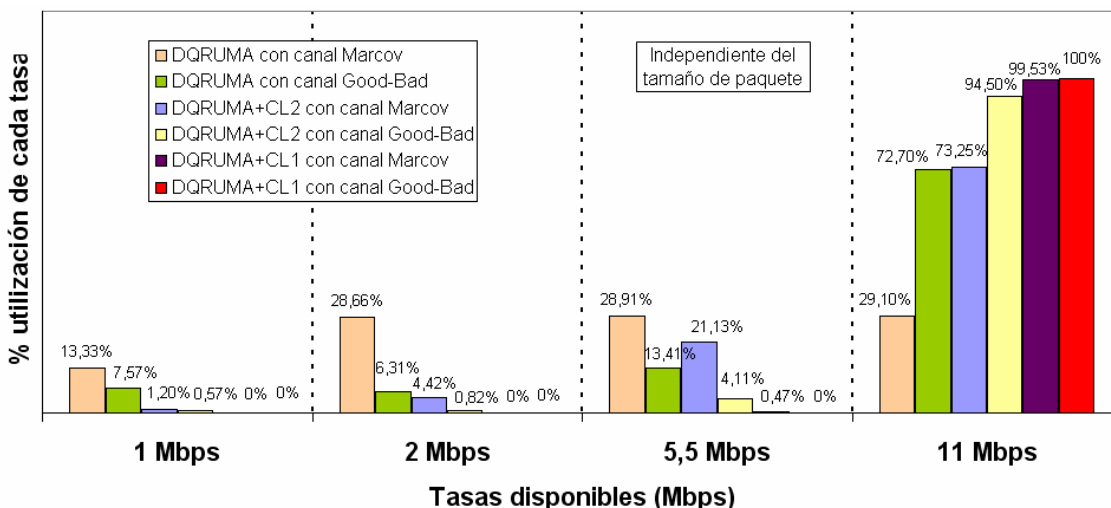


Fig. 5.12 Porcentaje de utilización de cada tipo de tasa para cada modelo de canal y técnica *scheduling*, con throughput máximo y tráfico real (Figura 5.6 + CL2)

Podemos observar como la técnica CL2 mantiene un compromiso entre la calidad del canal del usuario y su posición en la tabla de acceso, es decir, que se encuentra en un estado intermedio entre DQRUMA básico y CL1. De esta manera, los usuarios que se encuentren a más distancia del AP y por lo tanto sólo dispongan de velocidades inferiores a la máxima (1, 2 y 5.5 Mbps), tendrán más posibilidades de transmitir que con CL1.

Vemos que los valores de los porcentajes se corresponden en proporción con el throughput de las figuras 5.10 y 5.11. Esto es debido a la relación directa entre el porcentaje de utilización de cada trama con el throughput.

5.3.2. Análisis del retardo medio

CL2 prioriza mediante una fórmula que permite que haya más posibilidades de transmitir a una tasa superior que con DQRUMA básico, pero no obliga a transmitir siempre a tasa máxima como hace CL1, así que nos encontramos con un retardo superior en estos paquetes y consecuentemente con un retardo medio del sistema superior que el obtenido con CL1. En la figura 5.13 se compara el retardo medio de DQRUMA básico con CL1 y con CL2, también utiliza el canal Good-Bad y lo valora para los dos tamaños de paquete vistos antes.

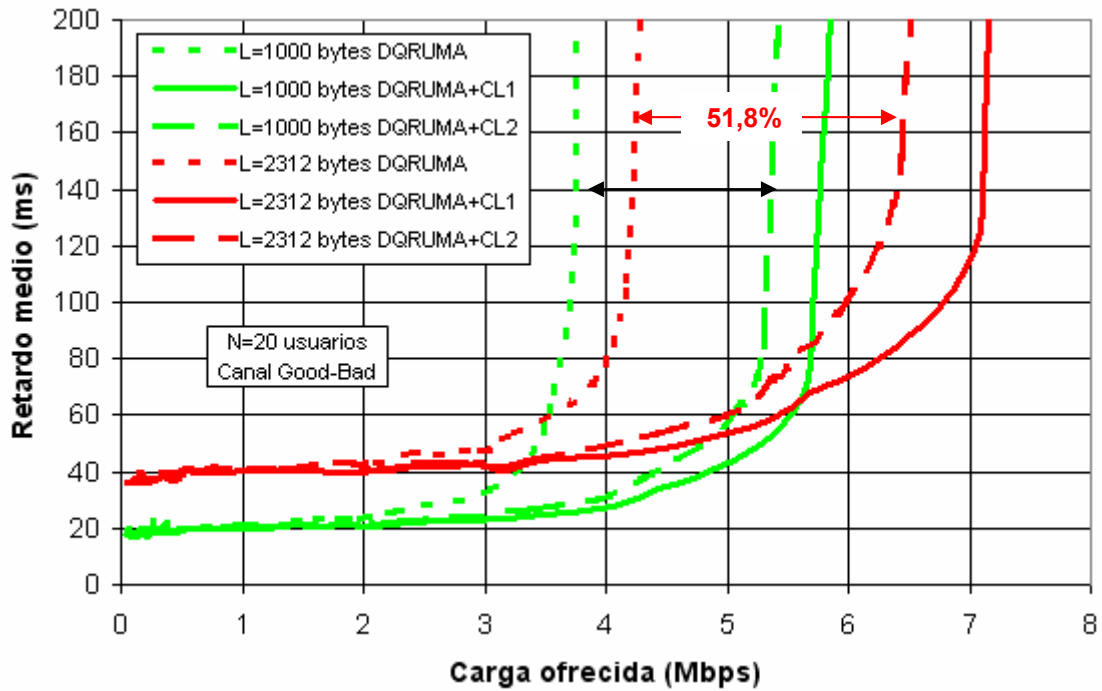


Fig. 5.13 Retardo medio con tráfico de control real y canal Good-Bad

Podemos observar como la técnica CL2 mejora mucho la carga soportada respecto DQRUMA básico (ganancia de 51,8%), pero además se ve como CL1 mejora a CL2.

También vemos como la mejora del retardo es directamente proporcional al tamaño del paquete de transmisión.

Por último observamos como CL2 vuelve a demostrar un compromiso entre tasa máxima y posición, encontrándose entre DQRUMA básico y CL1.

5.3.3. Análisis de la desviación típica del retardo medio

En el análisis de este parámetro evaluaremos CL2 junto a CL1 y DQRUMA básico, para poder valorar las ventajas que ofrece CL2 en términos de desviación típica del retardo. Lo valoraremos con tres tamaños de paquete diferentes (100, 570 y 2312 bytes). Podemos ver esto en la figura 5.14.

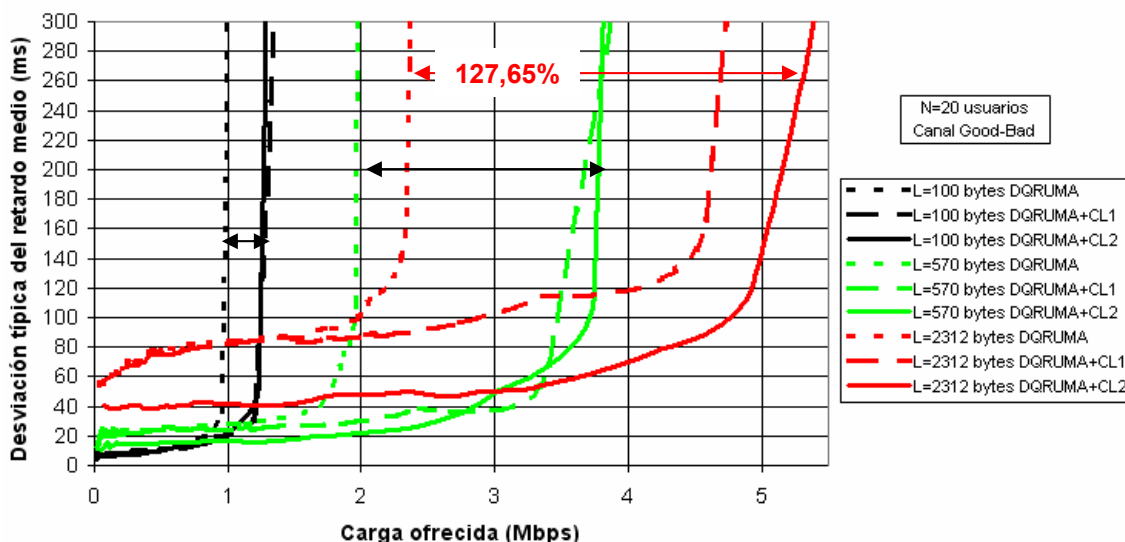


Fig. 5.14 Desviación típica del retardo medio con tráfico de control real y canal Good-Bad

Podemos ver como DQRUMA básico, sufre un desvío superior para tamaños mayores de paquete. En cambio cuanto mayor es el paquete, CL2 mejora incluso más que CL1 con aumentos de carga de hasta el **127,65%**

5.3.4. Conclusiones acerca del rendimiento de la técnica CL2

Después de analizar todos estos parámetros de la segunda técnica Cross-Layer, podemos decir que también se han cumplido los objetivos marcados. Ya que el sistema obtiene, gracias a esta técnica, una mayor eficiencia y un aprovechamiento mayor de los recursos de éste. A continuación resumimos los resultados analizados:

- En términos de throughput incluso cuando el sistema utiliza el tráfico de control real, consigue obtener valores mucho más altos que con el DQRUMA básico (incrementos de hasta un 154.16% para el canal Marcov).
- Se consiguen ganancias de throughput para ambos canales analizados y para cualquier tipo de tamaño de paquete de transmisión. En concreto para el canal Marcov se aumenta el throughput máximo un 154,16%. Aunque cabe decir que CL1 mejora las ganancias y cotas máximas de CL2.
- CL2 también mejora el retardo medio de los mensajes respecto DQRUMA básico para cualquier tipo de paquete, disminuyendo el tiempo de transmisión de cada paquete, y por lo tanto a la vez el retardo total del sistema. Aumenta un 51,8% la carga soportada. Pero debemos decir que es mejorado por CL1.

- La desviación típica con CL2 es menor que con CL1 y mucho menor que DQRUMA básico. Se consiguen ganancias prometedoras del 127,65%.

Después de ver estas cuatro mejoras concluimos que la técnica CL2 mejora de forma significativa la eficiencia de los sistemas WLAN y por ese motivo proponemos utilizar esta técnica en futuros diseños de redes inalámbricas.

5.4. Rendimiento en presencia de tráfico de voz

En este apartado vamos a evaluar el efecto de las técnicas Cross-Layer sobre un entorno con varios usuarios que mantienen comunicaciones de voz en vez de datos. Tal como hemos explicado a lo largo del trabajo, estas comunicaciones tienen una alta restricción en términos de retardo medio (el 1% de los paquetes generados por un usuario no puede retardarse más de 300, 200 o 150 ms). Por ese motivo, para simular este entorno se han considerado estas tres cotas máximas de retardo permitido. También hemos decidido que el tamaño de paquete de transmisión de voz será el mínimo ($L=100$ bytes) ya que es el que produce un retardo inferior (mirar figuras 5.7 y 5.8). Además hemos simulado el sistema para los dos modelos de canal (Good-Bad y Marcov). Y todo esto para ver el incremento del throughput y de la capacidad del sistema (número de usuarios que el sistema soporta sin que ninguno de ellos sobrepase el límite del 1%). A continuación veremos todos estos parámetros de simulación, por una parte en el caso de la utilización de CL1 y por otra la de CL3. El motivo por el cual hemos escogido la técnica CL3 para analizar tráfico de voz es evidente porque ha sido creada con este fin, pero el utilizar como técnica secundaria a analizar CL1 y no CL2 es porque el primero ofrece un retardo inferior (parámetro muy importante en voz), a parte de la mejora en throughput.

5.4.1. Rendimiento de la técnica CL1 en presencia de tráfico de voz

Teniendo en cuenta los parámetros vistos anteriormente, aquí mostraremos los resultados para la técnica CL1 respecto a DQRUMA básico. Estos resultados se muestran en la tabla 5.1.

Tabla. 5.1 Número máximo de usuarios de voz para la técnica CL1

	Técnica de scheduling	Canal Good-Bad		Canal Marcov	
		Capacidad (usuarios)	Throughput (Kbps)	Capacidad (usuarios)	Throughput (Kbps)
300 ms	DQRUMA	124	696,830	116	659,392
	DQRUMA+CL1	136	775,911	124	698,468
	Incremento de Capacidad y Throughput	9,67%	11,35 %	6,89 %	5,92 %
200 ms	DQRUMA	114	643,852	107	604,656
	DQRUMA+CL1	130	734,820	118	666,074
	Incremento de Capacidad y Throughput	14,03 %	14,12 %	10,28 %	10,15 %
150 ms	DQRUMA	108	613,217	103	583,125
	DQRUMA+CL1	122	687,797	115	658,287
	Incremento de Capacidad y Throughput	12,96 %	12,16 %	11,65 %	11,88 %

En la tabla podemos observar como la técnica CL1 aumenta la eficiencia del sistema en términos de capacidad y throughput para los tres tipos de limitación de retardo. También podemos ver que para valores menos restrictivos los resultados máximos alcanzables son más altos. Por último, para el canal Good-Bad además de conseguir resultados más altos las ganancias son superiores.

Concretando más en valores máximos adquiridos, utilizando CL1, con una restricción de 300 ms y en el canal Good-Bad, se puede llegar a mantener hasta **136** usuarios de voz simultáneos de forma estable. En términos de ganancia en porcentaje, en el caso que más se mejora mediante CL1 es para canal Good-Bad y con la restricción temporal de 200 ms obteniendo un **14,03%** al pasar de 114 a 130 usuarios. Recordamos que aunque haya ganancias más altas para retardos más restrictivos, los valores máximos adquiridos en capacidad del sistema se consiguen cuanto mayor es el tiempo límite de retardo, tal como hemos visto en el caso de los 136 usuarios para restricción de 300 ms.

Vistos estos resultados, podemos plantear la utilización de la técnica CL3, ya que además de tener en cuenta la tasa a la que se transmite el paquete de voz, también valora el número de paquetes que tienen los usuarios en su buffer y por lo tanto el tiempo que el usuario va a utilizar una tasa máxima para transmitir más o menos paquetes.

5.4.2. Rendimiento de la técnica CL3 en presencia de tráfico de voz

Manteniendo la misma idea que para el análisis de CL1, a continuación mostramos los resultados de ambas técnicas (CL1 y CL3) para ver qué incremento de capacidad y throughput aplican a DQRUMA básico. Podemos observar estos resultados en la tabla 5.2.

Tabla. 5.2 Número máximo de usuarios de voz para la técnica CL1 y CL2

		Canal Good-Bad		Canal Marcov	
Técnica de schedulling		Capacidad (usuarios)	Throughput (Kbps)	Capacidad (usuarios)	Throughput (Kbps)
300 ms	DQRUMA	124	696,830	116	659,392
	DQRUMA+CL1	136	775,911	124	698,468
	DQRUMA+CL3	139	797,568	126	715,542
	Incremento de Capacidad y Throughput (CL1)	9,67%	11,35 %	6,89 %	5,92 %
	Incremento de Capacidad y Throughput (CL3)	12,10 %	14,45 %	8,62 %	8,51 %
200 ms	DQRUMA	114	643,852	107	604,656
	DQRUMA+CL1	130	734,820	118	666,074
	DQRUMA+CL3	133	749,774	120	677,631
	Incremento de Capacidad y Throughput	14,03 %	14,12 %	10,28 %	10,15 %
	Incremento de Capacidad y Throughput (CL3)	16,66 %	16,45 %	12,15 %	12,06 %
150 ms	DQRUMA	108	613,217	103	583,125
	DQRUMA+CL1	122	687,797	115	658,287
	DQRUMA+CL3	126	715,185	117	662,834
	Incremento de Capacidad y Throughput	12,96 %	12,16 %	11,65 %	11,88 %
	Incremento de Capacidad y Throughput (CL3)	16,66 %	16,63%	13,60 %	13,67 %

En esta tabla podemos observar como los valores de CL3 siguen la misma idea explicada en la tabla 5.1, pero en este caso son superiores que los de CL1.

Concretando más en valores máximos adquiridos, utilizando CL3, con una restricción de 300 ms y en el canal Good-Bad, se puede llegar a mantener hasta **139** usuarios de voz simultáneos de forma estable. En términos de ganancia porcentual, el mayor valor se consigue mediante CL3, el canal Good-Bad y con la restricción temporal de 200 ms y de 150 ms. Con ambas restricciones se obtiene un **16,66%** al pasar de 114 a 133 y de 108 a 126 respectivamente. Recordamos que aunque haya ganancias más altas para retardos más restrictivos, los valores máximos adquiridos en capacidad del sistema se consiguen cuanto mayor es el tiempo límite de retardo, tal como hemos visto en el caso de los 139 usuarios para restricción de 300 ms.

5.4.3. Conclusiones acerca del rendimiento de las técnicas CL1 y CL3 en presencia de tráfico de voz

Después de analizar los resultados obtenidos con ambas técnicas podemos decir que se han conseguido los objetivos fijados. Hemos logrado una mejora de la eficiencia en términos de capacidad del sistema y throughput máximo alcanzable. Resumimos las mejoras ofrecidas por estas dos técnicas a continuación:

- Ambas técnicas consiguen aumentar la eficiencia del sistema en los dos parámetros antes comentados.
- La técnica CL3, como era de esperar, ofrece una ganancia mejor para una misma calidad en requerimientos mínimos. Por ejemplo CL3 consigue una capacidad en número de usuarios de 139 en contra de 136 de CL1 y de 124 de DQRUMA básico. Además obtiene una ganancia máxima en capacidad del 16,66%, en cambio CL1 del 14,03%, pero esta no es la máxima diferencia entre ellas. Si valoramos todas las comparaciones entre ambas, existe una diferencia que puede variar del 2 al 4 %.

Vistas estas mejoras concluimos que la técnica CL3 es una buena opción (mejor que CL1) a tener en cuenta, en términos de tráfico de voz, para el diseño de futuras redes inalámbricas.

5.5. Rendimiento con tráfico heterogéneo

A continuación mostraremos los últimos resultados que nos quedaban por analizar, los del tráfico heterogéneo, es decir, de tráfico de datos y voz compartiendo el mismo canal radio simultáneamente. La idea principal en este caso consiste en una simulación de un número variable de usuarios de datos y de voz que deben mantener la calidad de servicio necesaria. No utilizamos ningún tipo de prioridad entre ellos, de esta forma podremos valorar con total imparcialidad la respuesta que nos ofrece cada tipo de tráfico.

5.5.1. Análisis del retardo medio de mensaje y ráfaga

El modelo de canal utilizado es Good-Bad. Antes de ver resultados de CL mostraremos como varía el retardo medio con DQRUMA básico para varios tamaños de paquete diferentes. Mantendremos el número de usuarios de voz en 10 y los de datos también. De esta forma podremos valorar qué tamaño de paquete ofrece mejores prestaciones para igualdad de número de usuarios. Estos resultados se pueden ver en la figura 5.15.

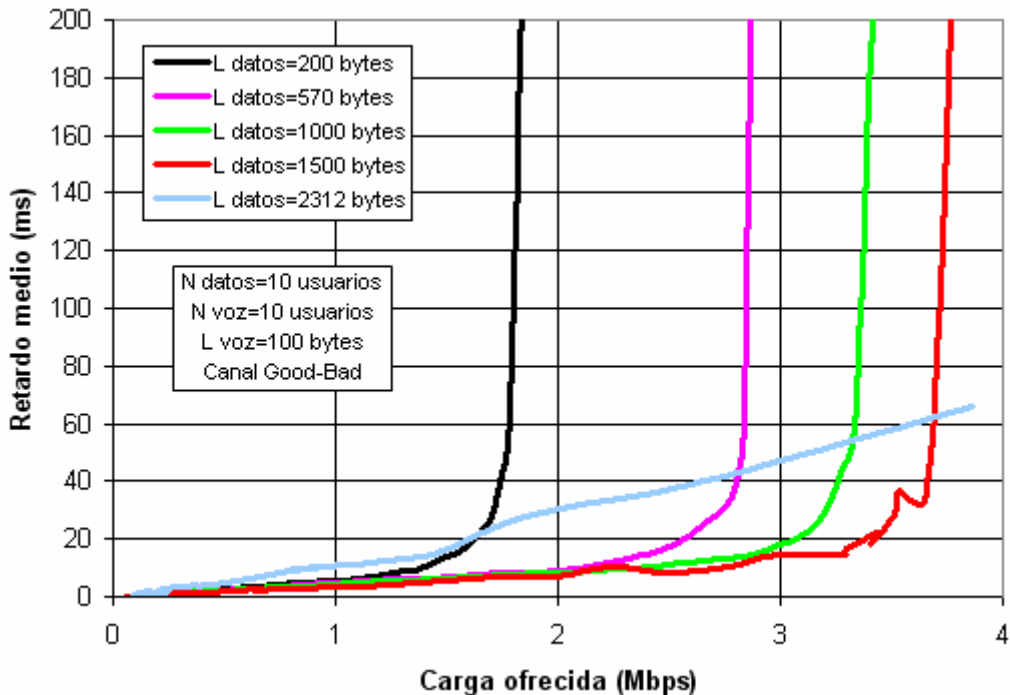


Fig. 5.15 Retardo medio de los mensajes de datos y ráfagas de voz con DQRUMA básico

Podemos observar como para un mayor paquete, el retardo soporta más carga, pero si analizamos cada tamaño de paquete de forma independiente podemos ver que para el tamaño $L=2312$ bytes la gráfica se corta, esto es debido a que para cargas superiores a esa no se cumplirían los requerimientos mínimos de calidad. Por otra parte vemos que para $L=1500$ bytes en la zona de funcionamiento es muy inestable. De esta forma escogemos $L=1000$ como tamaño de paquete óptimo para las comunicaciones mixtas de voz y datos, ya que es el que soporta una carga superior manteniéndose estable. Por lo tanto la evaluación de técnicas CL se hará con el tamaño de paquete de $L=1000$ bytes.

A continuación estudiaremos las prestaciones de las técnicas CL para el canal Good-Bad, para el tamaño de paquete $L=1000$ bytes, con un número de usuarios de datos fijo de 10 usuarios pero variando el número de usuarios de voz. Hemos decidido utilizar la técnica CL1 para éste tráfico mixto, ya que al no utilizar ningún tipo de prioridad entre tráficos, nos encontrábamos con que datos enmascaraba a voz. Esto es debido a que el número de paquetes de una ráfaga de voz no pasa de 4, en cambio el de datos tiene una media de 10 paquetes por mensaje. De esta forma si priorizábamos mediante CL3 nos encontrábamos que para una misma tasa máxima entre un usuario de datos y otro de voz, se escogería el que tuviese un mayor número de paquetes, es decir mayoritariamente datos. De esta forma se impediría la transmisión de voz, provocando que el retardo aumentase rápidamente, incumpliendo los requerimientos mínimos de calidad de servicio de voz. En cambio con CL1 se consigue total imparcialidad entre voz y datos, ya que los dos tipos de tráfico tienen las mismas posibilidades de conseguir una tasa alta. Y si dos usuarios

con tráfico diferente tienen la misma tasa, transmitirá el que se encuentre en una posición menor, favoreciendo al retardo nuevamente. En la figura 5.16 mostramos los resultados de DQRUMA básico y CL1.

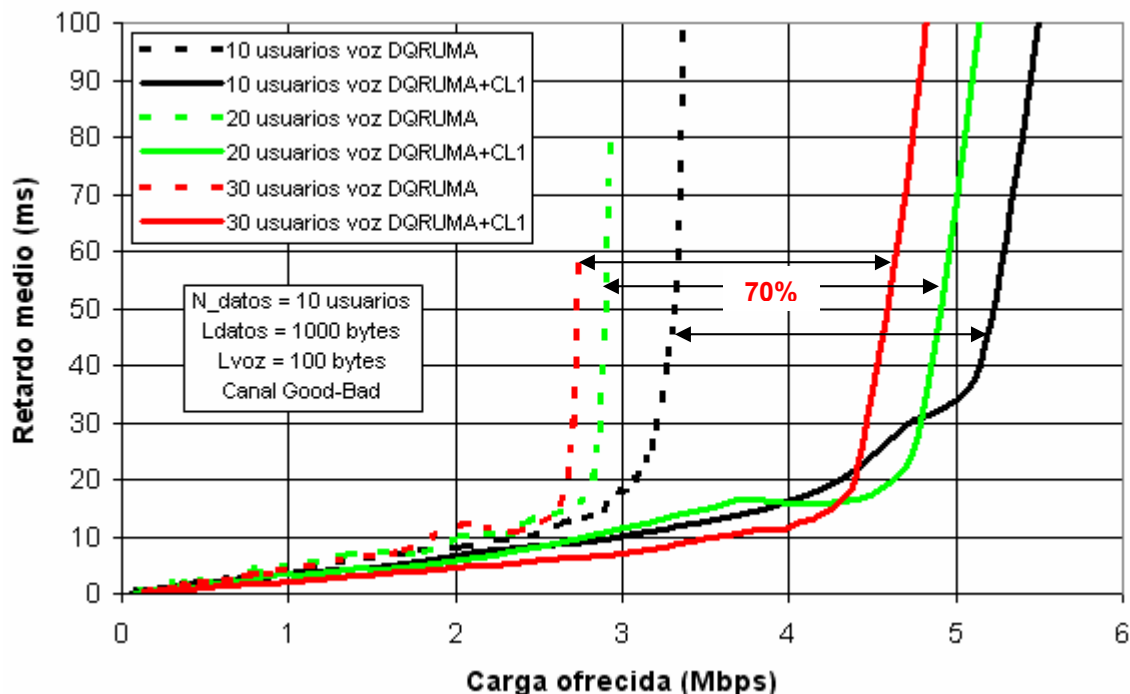


Fig. 5.16 Retardo medio de los mensajes de datos y ráfagas de voz con CL1

Por una parte vemos como para un mayor número de usuarios de voz en el sistema, el retardo empeora. Por otra parte observamos que la técnica CL1 obtiene mejora para los diferentes valores de número de usuarios de voz. En concreto mantiene una ganancia prácticamente constante de 2 Mbps (70% de ganancia) de carga sobre DQRUMA básico aunque varíe el número de usuarios de voz.

5.5.2. Conclusiones acerca del rendimiento con tráfico heterogéneo

Después de analizar el parámetro de retardo medio de mensaje y ráfaga para CL1 y diferentes tamaños de paquete, podemos decir que se han cumplido los objetivos marcados. Ya que el sistema obtiene, gracias a esta técnica, una mayor eficiencia y un mejor aprovechamiento de los recursos de éste. A continuación resumimos los resultados analizados:

- Para DQRUMA básico hemos llegado a la conclusión después de evaluar diferentes tamaños de paquete que el óptimo es $L=1000$ bytes.
- Para el tamaño de paquete óptimo se ha analizado DQRUMA básico y CL1 obteniendo resultados mucho mejores con la técnica CL. El sistema mediante ésta técnica es capaz de soportar del orden de 2 Mbps más de

carga en términos de retardo medio que DQRUMA básico que equivale a una ganancia del 70% en términos de retardo medio.

Por todo ello consideramos que la técnica CL1 es una buena opción a tener en cuenta para el tráfico heterogéneo en el diseño de futuras redes inalámbricas.

5.6. Conclusiones generales tras el análisis de resultados

Por todo esto finalizamos diciendo que estas técnicas benefician todos los parámetros analizados en los diferentes escenarios. Podemos concretar en la eficiencia de cada una en función del tráfico utilizado. Para el caso de tráfico de datos, de entre CL1 y CL2, actualmente elegimos CL1 que optimiza más el sistema, pero no descartamos que modificando la fórmula utilizada por CL2 se obtengan resultados prometedores. En el caso de tráfico de voz, entre CL1 y CL3, escogemos CL3 porque nos ha dado mejores resultados. Y por último para el tráfico heterogéneo, entre CL1 y CL3, si no utilizamos prioridad entre tráficos (que como hemos comentado antes es nuestro objetivo) escogemos CL1, ya que ofrece mejora manteniendo los requerimientos mínimos de calidad. Aunque cabe decir que si en el tráfico heterogéneo utilizásemos prioridades, la técnica CL3 podría ser una opción a estudiar.

CAPÍTULO 6. CONCLUSIONES Y LÍNEAS FUTURAS DE INVESTIGACIÓN

6.1. Conclusiones finales

Teniendo en cuenta el aumento actual de la utilización de las comunicaciones inalámbricas y en concreto las redes locales inalámbricas con el estándar 802.11b, hemos realizado un trabajo de fin de carrera en el que se estudian técnicas de gestión de recursos radio para optimizar estos sistemas que están en creciente demanda en el mercado.

Para situar el marco en el cual se han realizado los análisis y del cual se han sacado las conclusiones finales diremos que el estudio se ha realizado teniendo en cuenta el estándar 802.11b y se ha utilizado DQRUMA como protocolo de alta eficiencia MAC. También se han utilizado dos modelos diferentes de canal (Good-Bad y Marcov) y diferentes tamaños de paquetes de transmisión (0-2312 bytes). Por último, comentar que para evaluar la eficiencia de las técnicas CL1 y CL2 sobre el tráfico de datos, se han valorado los parámetros de: throughput, retardo medio de los mensajes, desviación típica del retardo medio y porcentaje de utilización de las tasas disponibles (1, 2, 5'5 y 11 Mbps). Para valorar las técnicas CL1 y CL3 sobre tráfico de voz, se han utilizado los parámetros de capacidad del sistema en número de usuarios y throughput. Y para analizar la técnica CL1 en tráfico heterogéneo se ha usado el parámetro de retardo medio de los mensajes y ráfagas.

A continuación mostramos las conclusiones principales consideradas tras acabar el trabajo:

- Tanto CL1, CL2 como CL3 ofrecen beneficios en las diferentes situaciones estudiadas.
- Concretando en la técnica CL1 (que prioriza por tasa), hemos obtenido diferentes gráficas para tráfico de datos y heterogéneo (voz y datos).
 - En el caso de datos se evaluaba el throughput máximo obtenido, el retardo medio mínimo, la desviación típica del retardo medio y el porcentaje de utilización de las diferentes tasas disponibles para transmitir (1, 2, 5'5 y 11 Mbps). Todos estos parámetros han mejorado al utilizarse la técnica CL1. Como ejemplo de todas estas mejoras, se obtiene una ganancia del throughput máximo alcanzable de 267,94% sobre DQRUMA básico.
 - En el caso de tráfico heterogéneo hemos analizado el retardo medio de mensajes y ráfagas, obteniendo mejoras del 70% con CL1 respecto DQRUMA básico.

- Con la técnica CL2 (que prioriza por la relación entre la tasa y la posición del usuario en la tabla de acceso), hemos obtenido también mejoras considerables en throughput, retardo medio, porcentaje de utilización de las tasas y desviación típica del retardo medio, respecto DQRUMA básico. CL1 supera a CL2 en todos los aspectos estudiados excepto en desviación típica del retardo medio que aumenta un 127,65% sobre DQRUMA básico.
- La técnica CL3 la hemos utilizado junto a CL1 para evaluar su eficiencia sobre el tráfico de voz. Con ambas técnicas hemos obtenido mejoras sobre DQRUMA básico, cuando las hemos simulado con tres tipos de restricciones en término de requerimientos mínimos de calidad de servicio (300, 200 y 150 ms). Aunque la técnica CL3 en todos los casos valorados para tráfico de voz, ha mejorado la eficiencia de CL1 en términos de capacidad del sistema en número de usuarios y en throughput máximo alcanzable. Con CL3 hemos llegado a obtener incrementos en número de usuarios sobre DQRUMA básico del 16,66%.

Después de realizar el análisis de las tres técnicas Cross-Layer, podemos concluir que éstas ofrecen una mejora en tráfico de datos, de voz y heterogéneo, gracias al intercambio de información que realiza entre la capa MAC y la capa física de las capas OSI. Y que por lo tanto se han logrado todos los objetivos marcados.

Además de todo esto, cabe decir que no sólo es de óptimo funcionamiento y aprovecha bien los recursos del sistema, sino que también es de fácil implementación.

6.2. Consideraciones medioambientales

En este trabajo se ha propuesto mejorar las comunicaciones inalámbricas teniendo en cuenta el medioambiente. Es decir que además de optimizar las comunicaciones, la aplicación de las técnicas CL reducen el nivel de contaminación electromagnética y consiguen ahorro energético.

6.3. Líneas futuras de investigación

Una vez analizadas las posibilidades de las técnicas utilizadas en este trabajo, proponemos diversos caminos por donde podrían dirigirse las futuras investigaciones sobre este tema.

- La técnica CL2 se podría estudiar intentando encontrar una fórmula de prioridad diferente a la que utiliza actualmente, y así mejorar en términos de throughput y retardo medio.

- Investigar el rendimiento de las técnicas CL sobre escenarios de trabajo con distintas condiciones a las que hemos utilizado nosotros, y analizar por ejemplo:
 - La consideración de la probabilidad de error en el bit (BER)
 - Utilización de codificación de canal.
 - Modificación del modelo de movilidad de los nodos.
 - Probar otros modelos de canal diferentes.
 - Considerar capas físicas de estándares diferentes 802.11x, como por ejemplo las basadas en OFDM del estándar 802.11g.
 - Tener en cuenta el mecanismo de control de potencia.
- Recibir información sobre parámetros físicos diferentes a los utilizados y así encontrar nuevas técnicas Cross-Layer.
- Analizar diferentes criterios de prioridad para sistemas con distintos tráficos de información compartiendo el medio, como por ejemplo prioridad en tráfico heterogéneo de datos y voz para evaluar los efectos de la técnica CL3.

7. REFERENCIAS

- [1] KAROL, M.J., Liu, Z., Eng, K.Y., "An efficient demand-assignment multiple access protocol for wireless packet (ATM) networks", *Wireless Networks* 1 (1995) 267-279.
- [2] YASAMITA, T., MATSUMOTO, T., HARUKI, T., OKADA, H., "DQRUMA/PAR Method for Multimedia Wireless Access Networks", Kansas university, 3-3-35 Yamate-chou, Suita, Osaka 564-8680 Japan.
- [3] YAMAMOTO, M., MACHIDA, S., IKEDA, H., "Access Control Scheme for Multimedia ATM Wireless Local Area Networks", *IEICE TRANS. COMMUN.* Vol. E80-B, nº 1, January 1997.
- [4] SHAKKOTTAI, S., RAPPAPORT, T.S., KARLSSON, P.C., "Cross-Layer Design for Wireless Networks" *IEEE Communications Magazine*, October 2003.
- [5] CARNEIRO, G., RUELA, J., RICARDO, M., "Cross-Layer Design in 4G Wireless Terminals", *IEEE Wireless Communications*, April 2004.
- [6] RAISINGHANI, V.T., IYER, S., "Cross-layer design optimizations in wireless protocol stacks", Received 23 October 2003; accepted 24 October 2003.
- [7] ALONSO, L., FERRUS, R., AGUSTI, R., "WLAN Throughput Improvement via Distributed Queuing MAC", *IEEE Communication letters*, Vol. 9, nº4, April 2005.

ANEXOS

ANEXO 1. ENTORNO DE TRABAJO

1.1. Introducción

En este capítulo presentaremos y describiremos con detalle los elementos principales del entorno de trabajo sobre el que funcionan las técnicas CL y los parámetros relevantes que debemos tener en cuenta para valorar la eficiencia de estas técnicas.

1.2. Entorno de trabajo

1.2.1. Sistema WLAN 802.11b

Teniendo en cuenta que el escenario de trabajo es un sistema WLAN en modo infraestructura basado en el estándar 802.11b. Consideramos un número variable de nodos que se comunican con un punto de acceso o base y que comparten el mismo canal de acceso radio. El protocolo MAC utilizado es DQRUMA (ver capítulo 2), y como allí mismo hemos comentado el estudio se ha realizado para tráfico de datos en UL y el de control DL, tal como muestra la dirección y sentido de las flechas en la figura 1.1.

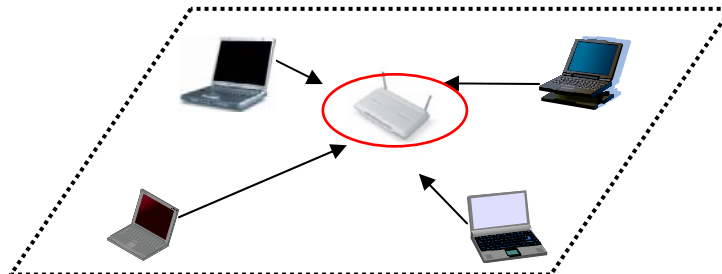


Fig. 3.1 Sistema WLAN simulado por ordenador

Los valores de los parámetros principales de la capa física los mostramos en la tabla 1.1.

Tabla 1.1 Principales parámetros de la capa física 802.11b

Bandas frecuenciales	2,4 – 2,4835 GHz
Técnica de espectro ensanchado	DSSS
Potencia de transmisión	100 mW
Tasas de transmisión disponibles	1, 2, 5'5, 11 Mbps

La potencia a la que transmiten los nodos es constante, es decir que no utilizan control de potencia. Debido a esto, si se producen pérdidas en el canal radio provocadas por interferencias o lejanía del usuario respecto a la base, la relación señal a ruido (SNR) del nodo disminuirá. Si pretendemos mantener una probabilidad de error en el bit (P_b) constante, la velocidad de transmisión deberá disminuir adaptándose al cambio de SNR. Por este motivo el estándar 802.11 establece cuatro velocidades de transmisión (R_b) disponibles (1, 2, 5.5, o 11 Mbps), las cuales se utilizarán en función del estado del canal. En el caso de tráfico de datos será un valor de los cuatro mostrados, pero en el caso del tráfico de control se transmitirá a la mínima velocidad (1 Mbps) para asegurar que se recibe correctamente.

Los nodos o terminales del sistema consisten en dispositivos que generan información en forma de mensaje o ráfaga dependiendo de si el modelo de tráfico utilizado es datos o voz respectivamente. Cabe decir que tanto los mensajes como las ráfagas se acaban descomponiendo en paquetes. En el punto 1.2.7 se habla con más profundidad sobre los modelos de tráfico. Los mensajes o ráfagas generados se almacenan en un *buffer* tal como explicamos en el protocolo DQRUMA. Consideramos una capacidad para 200 mensajes, donde cada mensaje tiene una longitud variable de bits. Esta longitud de bits se acabará transmitiendo en paquetes del tamaño estipulado por la base. Si el tamaño de bits del mensaje a transmitir es superior al tamaño de paquete marcado, cosa habitual, la capa MAC se ocupará de fraccionarlo de la forma adecuada y utilizará más de una trama DQRUMA para su transmisión. La figura 1.2 muestra un diagrama donde se representa el modelo de nodos usado.

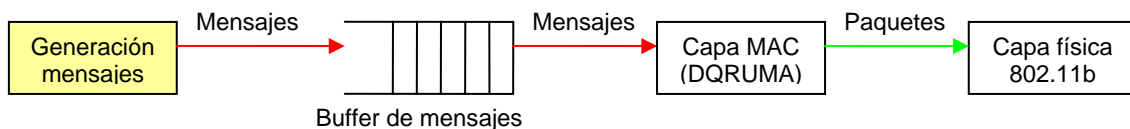


Fig. 1.2 Diagrama funcional de un nodo

1.2.2. Adaptación de DQRUMA original al estándar 802.11b

A parte del paso de un sistema FDD al TDD explicado en 2.3, hemos hecho un nuevo cambio respecto a la propuesta original de DQRUMA que consiste en pasar de un sistema WATM, donde el tiempo dedicado a tráfico de datos (DATA) era fijo, es decir, tanto su tamaño como su velocidad de transmisión eran fijos, a nuestro estándar con el que utilizamos DATA con tiempo variable. Tenemos un tamaño de paquete variable de 0-2312 bytes, donde cada tamaño de paquete se puede transmitir a 4 velocidades diferentes estipuladas por el estándar (1, 2, 5.5 o 11Mbps), así que se puede valorar la gran variedad de tiempos que hay. Debemos decir que el PGBK es de tamaño fijo (1 bit) pero depende de la tasa de DATA, así que puede tener 4 valores de tiempo diferentes. En cambio el tráfico de control es de tiempo fijo. Esto es debido a que el tamaño de control es fijo y a que se debe asegurar su correcta recepción para que el sistema funcione correctamente y de forma eficiente, por eso se

transmite a la mínima velocidad (1Mbps) constante. Concretamos más sobre los valores de tamaño y tiempo que hemos utilizado, en el apartado 1.2.8.

1.2.3. Hipótesis de transmisión

Los siguientes puntos son hipótesis y aproximaciones que utilizamos con el fin de simplificar el escenario de trabajo pero que no adulteran los resultados obtenidos:

- **Selección de tasa binaria y probabilidad de error en el bit (P_b):** Consideramos que la selección de tasa es ideal, convirtiendo la P_b en nula. Es decir, todos los bits enviados a través del canal radio a la tasa seleccionada se reciben correctamente.
- **Control de potencia:** El estándar 802.11b especifica que no se implemente mecanismo de control de potencia, por lo tanto la potencia de transmisión será constante y sufrirá los cambios del canal, obligándonos a variar la SNR en función de ésta.
- **Codificación de canal:** Se considera que todos los bits transmitidos son útiles, es decir que no se utiliza codificación de canal.

1.2.4. Modelo del terreno del escenario de trabajo

Existen muchos escenarios posibles para simular el estudio, pero nosotros por simplicidad en las simulaciones hemos escogido un terreno rectangular, liso y donde todos los usuarios se pueden mover libremente. Las medidas del terreno utilizadas son las de la figura 1.3.

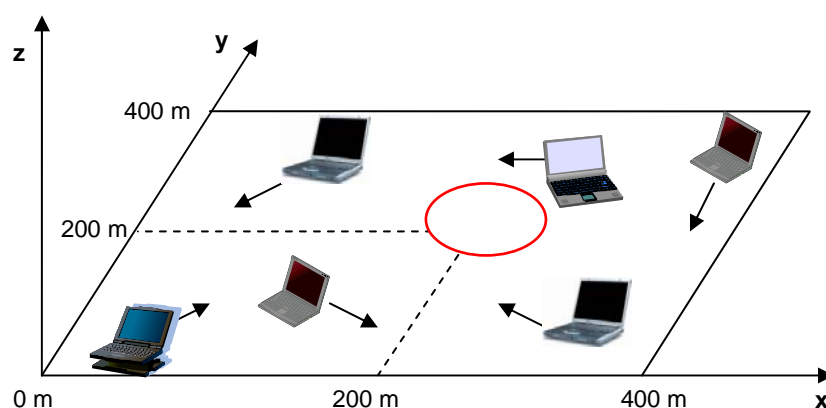


Fig. 1.3 Escenario de trabajo simulado por ordenador

1.2.5. Modelo de movilidad de los nodos

También hay disponible infinidad de modelos empíricos y estadísticos para decidir el tipo de movimiento de los usuarios y variación del canal radio. El que hemos escogido se explica a continuación.

El canal radio varía en el tiempo con una rapidez que depende principalmente de la velocidad relativa entre transmisor y receptor, y de la frecuencia de trabajo. Para contabilizar esta variación de canal se utiliza el “tiempo de coherencia”. Este parámetro indica durante cuánto tiempo la autocorrelación del canal se va a mantener por encima del 50%. Es decir, el tiempo en que las condiciones de canal se mantendrán invariantes y por lo tanto, cada cuanto los usuarios deben actualizarse. La elección de este tiempo la hemos realizado teniendo en cuenta los siguientes parámetros:

- Hemos escogido una velocidad de movimiento para todos los nodos constante de 1m/s, que es una velocidad coherente para entornos de interiores y de movilidad media/baja, típica de entornos WLAN.
- La frecuencia de trabajo es de 2,4 GHz.

Teniendo en cuenta estos dos parámetros, el tiempo de coherencia resultante es de 30,27 ms, un valor coherente en sistemas WLAN.

1.2.6. Modelos de canal

Para realizar las simulaciones de las técnicas CL se necesita un canal sobre el que funcionar. En este trabajo proponemos dos modelos diferentes con los cuales se ha realizado las simulaciones. Primero un modelo de canal denominado *Good-Bad*. Segundo un modelo que tiene memoria, con el fin de simular la correlación existente en los canales móviles reales, llamado *Markov*. Seguidamente describimos las características principales de ambos modelos.

1.2.6.1. Modelo de canal Good-Bad

Este modelo de canal simula la variación de un canal radio mediante una cadena de Markov discreta de dos estados. Es decir que el canal puede estar en favorable (*good*) o desfavorable (*bad*). La posibilidad de estar en un estado u otro lo marcan las probabilidades de la tabla 1.2, y el valor de señal a ruido del canal viene determinado por una variable aleatoria uniforme que puede variar entre los intervalos mostrados también en la tabla.

Tabla 1.2 Parámetros del modelo de canal Good-Bad

Estado del canal	Probabilidad	SNR (v.a uniforme)
Good	0,8	[10-20] dB
Bad	0,2	[0-10] dB

Las tasas de transmisión dependen de la SNR, y la SNR depende también a la vez de la distancia del nodo respecto al AP, por ese motivo marcamos una relación tasa-SNR mostrada en la tabla 1.3 y una relación tasa-distancia que podemos ver en la figura 1.4. Éstas nos permitirán seleccionar de forma correcta la velocidad de transmisión que corresponde a cada umbral de SNR o de distancia.

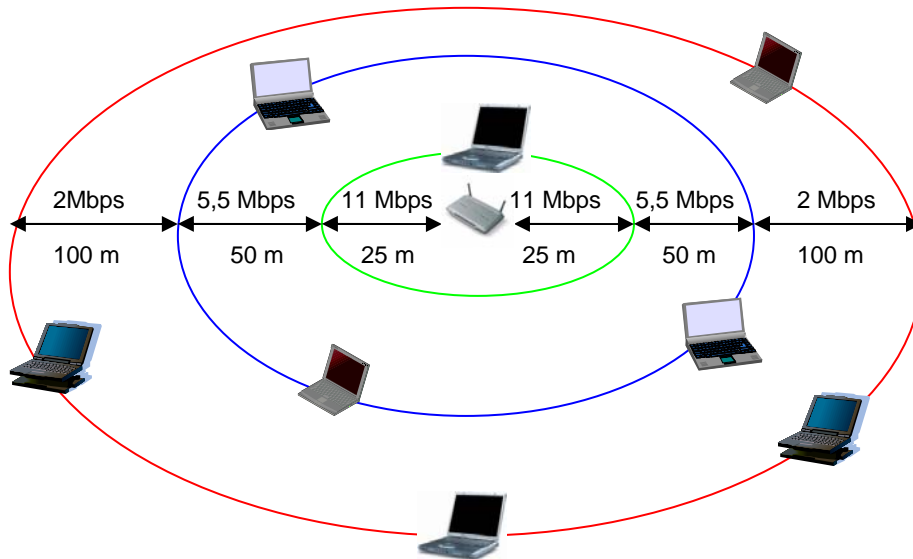


Fig. 1.4 Tasa o velocidad de transmisión en función de la distancia nodo-AP

Tabla 1.3 Umbrales de SNR para la selección de tasas según el modelo Good-Bad

SNR	<4dB	4-7,5 dB	7,5-11 dB	>11 dB
Tasas	1 Mbps	2 Mbps	5,5 Mbps	11 Mbps

Los valores de la figura 1.4 están considerados sin que haya obstáculos entre nodo y estación base, es decir, que la única atenuación supuesta es la de la distancia.

Podemos ver que para una distancia cercana, la SNR que tiene el usuario es alta, por lo tanto se puede permitir transmitir a una alta velocidad, y viceversa para distancias lejanas. Los usuarios que se encuentren al exterior de la línea roja aceptarán la velocidad de transmisión mínima de 1 Mbps.

1.2.6.2. Modelo de canal Markov

Los canales móviles, a pesar de variar en el tiempo, mantienen cierta correlación con los estados de tiempos pasados y futuros. Es decir, que un estado depende del estado actual pero también del anterior. Esta característica evita que un usuario que está transmitiendo a alta velocidad, pueda transmitir a baja velocidad en el siguiente instante inmediato, y viceversa. Ya que en un

caso real, moviéndose a 1 m/s sería imposible estar a la distancia propia de una SNR alta (25 m) y en el instante siguiente a la distancia de una baja (180m). Esta propiedad no se tiene en cuenta en el modelo de canal Good-Bad donde una vez transcurrido el tiempo de coherencia del canal, e independientemente de la tasa de transmisión anterior, una variable aleatoria uniforme determina la nueva relación señal-ruido y por lo tanto la nueva tasa de transmisión disponible.

En la figura 1.5 mostramos los valores de probabilidad de pasar de una tasa de transmisión a la otra en función de la actual.

		Estado futuro			
		1	2	5,5	11
Estado actual	1	0,5	0,4	0,1	0
	2	0,2	0,5	0,2	0,1
	5,5	0,1	0,1	0,5	0,3
	11	0	0,2	0,3	0,5

Fig. 1.5 Matriz de probabilidades de transición del canal Markov

Las filas de la matriz de transiciones representan el estado de transmisión actual o la tasa a la que el nodo puede transmitir. Las columnas representan el estado de transmisión futuro o la tasa a la que el nodo podrá transmitir una vez transcurrido el tiempo de coherencia. Podemos observar como los valores de la matriz confirman la idea teórica de Markov. Ofreciendo una menor probabilidad cuando mayor es la diferencia entre tasas, una mayor probabilidad si la diferencia entre tasas es pequeña y una probabilidad intermedia para las tasas que se mantienen iguales.

Cabe decir que este modelo sólo decide los valores en las transiciones pero nunca el estado inicial. Para obtener el estado inicial en las simulaciones, hemos utilizado en el primer instante el modelo Good-Bad y posteriormente Markov de forma constante. La tabla 1.4 muestra las probabilidades de estado en régimen permanente haciendo uso del modelo de canal Markov.

Tabla 1.4 Probabilidades de estado en régimen permanente para el canal Markov

P(1 Mbps)	0,1734
P (2 Mbps)	0,2931
P (5,5 Mbps)	0,295
P (11 Mbps)	0,2385

1.2.7. Modelos de tráfico

En este punto vamos a detallar los modelos de tráfico que se han utilizado en el sistema. En concreto explicaremos dos, el dedicado a datos y el específico para voz.

1.2.7.1. Datos genéricos (*best-effort*)

Este tipo de generación de tráfico se utiliza para servicios y conexiones que no requieren una alta calidad para funcionar. Las llamamos comunicaciones *best-effort*.

El presente modelo consiste en que los *buffers* de los usuarios recibirán los nuevos mensajes con estadística de Poisson, donde cada mensaje tiene una longitud cuyo valor es una variable exponencial aleatoria. Esta estadística se convierte en una probabilidad que nos permitirá variar el volumen de carga ofrecida al sistema, sin necesidad de variar el número de usuarios. Los parámetros que definen esta clase de tráfico se muestran en la tabla 1.5.

Tabla 1.5 Valores de los parámetros usados para el tráfico de datos genéricos

Tipo de servicio	Datos (<i>best-effort</i>)
Modelo de generación tráfico	Poisson
Tamaño del paquete de datos	$L=[100,200,570,1000,1500,2312]$ bytes
Longitud media de los paquetes	$10 \cdot L$ bytes
Retardo máximo de paquetes admitido	N/A

1.2.7.2. Voz (*ON-OFF*)

Este modelo de generación de tráfico va dedicado a usuarios que crean paquetes de datos correspondientes a un flujo de audio vocal. Consiste en un modelo clásico ON-OFF que explicamos mediante un diagrama de estados en la figura 1.6.

Los usuarios parten de un estado OFF o de silencio, en el que no generan información. Cada cierto tiempo tienen una probabilidad α de pasar al estado ON o activo, en el que generan paquetes de información (son los llamados *talkspurts* o ráfagas vocales). En este estado tienen a su vez una probabilidad de volver al estado OFF. Por tanto, el tiempo medio de estancia en cada estado es una variable aleatoria exponencial de media $1/\alpha$ y $1/\beta$ respectivamente.

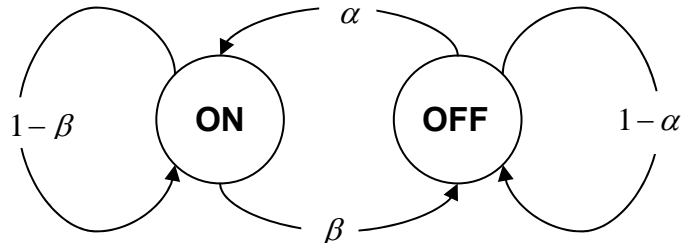


Fig. 1.6 Modelo de tráfico de voz ON-OFF

Los valores de las probabilidades α y β dependen del tiempo medio real de estancia en cada estado y de la duración de las tramas DQRUMA. Estos valores junto con el resto de parámetros de calidad de servicio definidos para este tipo de tráfico se detallan en la tabla 1.6.

Tabla 1.6 Valores de los parámetros usados para el tráfico de voz

Tipo de servicio	Voz
Modelo de generación de tráfico	ON-OFF
Duración media de los periodos ON	1,41 s
Duración media de los periodos OFF	1,74 s
Tasa de generación en estado ON	13 kbps
Tamaño del paquete de voz	100 bytes
Retardo máximo de paquetes admitido	300 ms
Porcentaje máximo de paquetes perdidos	1%

1.2.8. Tiempos de trama de DQRUMA

A continuación mostramos la tabla 1.7 con todos los tiempos de trama que hemos utilizado en la simulación. Como hemos informado en el apartado 1.2.2 de este anexo, los tiempos de tráfico de control son fijos permitiéndonos la determinación de su valor, en cambio los de tráfico de datos son variables así que determinamos su tamaño.

Tabla 1.7. Tiempos de trama de DQRUMA

Nombre	Explicación	Duración o tamaño
RTS	Petición de acceso (UL)	160 μ s
DATA	Transmisión de datos (UL)	0-2312 bytes
PGBK	Piggy backing (UL)	1 bit
CTS	Reconocimiento de acceso (DL)	112 μ s
XMT PERM	Elección del usuario que transmite (DL)	112 μ s
ACK DATA	Reconocimiento de datos	112 μ s
SIFS	Separación entre UL y DL	10 μ s

1.3. Definición de parámetros relevantes

Éstos son parámetros asociados a las redes de comunicación que utilizamos en la simulación para valorar desde diferentes puntos de vista la eficiencia de las técnicas Cross-Layer sobre el sistema.

1.3.1. Velocidad de transmisión o tasa (Throughput)

La velocidad de transmisión o tasa, en inglés throughput, consiste en la medición de la tasa de transferencia de datos o voz en un sistema de comunicaciones. Es decir, la cantidad de información transferida por unidad de tiempo, por lo tanto sus unidades son bits/s o bytes/s.

A continuación se muestra la expresión considerada para el cálculo del throughput medido en el AP:

$$\text{Throughput} = \sum_i \rho_i \sigma_i r_{bi} = \frac{\text{bits enviados correctamente}}{T_{\text{simulación}}} \quad (1.1)$$

Donde:

r_{bi} : Tasa de transmisión i -ésima de las disponibles en el sistema.

ρ_i : Eficiencia de las tramas DQRUMA haciendo uso de la tasa r_{bi} (ver más adelante).

σ_i : Porcentaje de utilización de la tasa de transmisión i -ésima.

El throughput es uno de los parámetros más importantes en la medida de la eficiencia de un sistema de comunicaciones, por ese motivo debe ser lo mejor posible. En concreto mide la cantidad de información útil transmitida respecto al tiempo total. Es decir, que todos los tiempos de trama visto en el apartado 2.3.1 que no sean exclusivamente DATA, es tráfico de control y no de datos/voz, que empeora el throughput pero que es necesario.

En un sistema WLAN 802.11b con protocolo de acceso DQRUMA, el throughput máximo alcanzable depende de tres factores:

- **Tasas de transmisión posibles (r_{bi}):** Tasas de transmisión a las que se pueden transmitir los paquetes. Según el estándar 802.11, disponemos de cuatro tasas posibles: 1, 2, 5'5 y 11 Mbps.
- **Eficiencia de las tramas DQRUMA (ρ_i):** Definida como el porcentaje de trama DQRUMA destinado a transmisión de datos o voz. El tamaño y velocidad de tráfico de control es fijo porque de éste depende que el sistema funcione correctamente. Sus características ya las hemos mostrado en la tabla 1.7 de este anexo. Así que lo único que podemos

modificar es el tráfico de datos. Por eso, cuanto mayor sea el tamaño del paquete transmitido y a más velocidad se envíe, más eficiencia tendremos. El tamaño puede variar de 0-2312 bytes, por lo tanto es variable pero está limitado a 2312 bytes. Por otra parte, si conseguimos que la tasa de transferencia en DATA sea la mayor parte del tiempo máxima, obtendremos mejor optimización en el sistema. Éste es el propósito de las técnicas CL que explicamos en el capítulo 3.

Se puede calcular como el cociente entre la duración del slot de datos (T_{datos}) o voz (T_{voz}) y la duración total de la trama DQRUMA (T_{trama}):

$$\rho_{datos} = \frac{T_{datos}}{T_{trama}} \quad \text{ó} \quad \rho_{voz} = \frac{T_{voz}}{T_{trama}} \quad (1.2)$$

- **Utilización de las tasas de transmisión (σ_i):** Definida como el porcentaje de tramas en el que se utiliza la tasa de transmisión r_{bi} .

1.3.2. Retardo medio de los mensajes

Definimos el retardo de un mensaje como el intervalo de tiempo transcurrido desde que un mensaje es generado por un nodo hasta que es recibido correctamente por el AP. El retardo medio de los mensajes se define entonces como el valor medio de los retardos sufridos por todos los mensajes enviados.

A continuación se muestra la expresión utilizada para el cálculo del retardo medio de los mensajes:

$$\text{Retardo medio mensajes} = \frac{\sum_i \text{retardo mensaje}_i}{\text{Total mensajes enviados}} \quad (1.3)$$

Donde:

retardo_mensaje_i: retardo sufrido por el mensaje *i*-ésimo.

Total_mensajes_enviados: *N*º total de mensajes recibidos por el AP.

1.3.3. Desviación típica del retardo de los mensajes

Definida como la 'dispersión respecto al valor medio', la desviación típica del retardo de los mensajes se ha utilizado para cuantificar las diferencias entre los retardos de los mensajes mediante la siguiente expresión:

$$\text{Desviación típica retardo} = \sqrt{E[\text{retardo mensajes}^2] - (E[\text{retardo mensajes}])^2} \quad (1.4)$$

Hay aplicaciones que requieren que la desviación típica del retardo esté acotada, es decir, que además de que los mensajes sufran un retardo medio bajo, las diferencias entre ellos deben ser mínimas.

1.3.4. Porcentaje máximo de paquetes perdidos

La fórmula para determinar el porcentaje de paquetes perdidos es:

$$\% \text{ paquetes perdidos} = \frac{\text{Total paquetes perdidos}}{\text{Total paquetes generados}} \times 100 \quad (1.5)$$

Donde:

Total_paquetes_perdidos: N° total de paquetes de voz de un usuario que sufre un retardo superior a 300, 200 o 150 ms, dependiendo del requerimiento del sistema.

Total_paquetes_generados: N° total de paquetes de voz generados por un usuario.

Este parámetro es importante en un sistema con tráfico de voz, ya que determinará si el oído humano percibe o no una buena comunicación, y conlleva dos restricciones. La primera consiste en que los paquetes generados por los usuarios no deben tardar más de (300, 200 o 150 ms) en ser recibidos por el AP, y el segundo es que el número de paquetes perdidos de un usuario no puede superar el 1% de sus paquetes generados. El incumplimiento del primer parámetro provocaría que la voz se oyese a destiempo y el segundo que se perdiese parte de la conversación.

Cabe decir que en el caso de datos se consideran paquetes perdidos cuando estos se retrasan más de 1s, y en ese caso se vuelven a reenviar, pero no tiene restricción por porcentaje de paquetes perdidos ya que si algún paquete no llega, se puede reenviar sin que el mensaje se degrade. Por ese motivo no se utiliza la fórmula anterior para el caso de datos.

ANEXO 2. PRINCIPALES PROTOCOLOS MAC PARA SISTEMAS INALÁMBRICOS

2.1 ALOHA

Este protocolo fue ideado en los años 70 en la universidad de Hawai como una solución al acceso al medio en redes de computadores mediante enlaces radioeléctricos. Desde entonces ha sido uno de los protocolos que más ha influido en el mundo de las redes de computadoras, y muchas han sido las mejoras que le han sido aplicadas.

La idea básica del protocolo es muy sencilla: simplemente se transmite un paquete de datos en cualquier momento en el que se tengan datos que enviar. Esto hace que la probabilidad que el paquete se destruya por colisión con los datos de otros usuarios sea muy alta. Sin embargo, los usuarios pueden, monitorizando el canal, saber si sus datos han colisionado con otros paquetes, o bien han llegado correctamente. Si el paquete de datos ha sido destruido, se espera un tiempo aleatorio y se vuelve a intentar la transmisión. Este tiempo de retransmisión debe ser aleatorio, puesto que si no lo fuera colisionaría con el paquete retransmitido por el otro usuario.

2.2 Carrier Sense Multiple Access Collision Avoidance (CSMA/CA)

Consiste en una modificación de ALOHA y en este caso se escuchan las colisiones. Todas las estaciones son consideradas igual, por ello compiten por el uso del canal, cada vez que una de ellas desea transmitir debe escuchar el canal, si alguien está transmitiendo espera a que termine, caso contrario transmite y se queda escuchando posibles colisiones, en este último espera un intervalo de tiempo y reintenta nuevamente.

2.3 Inhibit Sense Multiple Access (ISMA)

Este protocolo utiliza un canal de control en vez de escuchar el canal. Es una variación del CSMA/CA y está diseñado para adaptar las ideas de éste a entornos de transmisión por radio. ISMA no evita las colisiones en la transmisión ya que éstas se producen cuando dos o más usuarios inician sus transmisiones al mismo tiempo haciendo uso de un recurso indicado como libre.

2.4 Distributed Queuing Collision Avoidance (DQCA)

DQCA es un protocolo de acceso al medio que mediante el uso de colas distribuidas elimina casi por completo las colisiones en el sistema WLAN. Se trata de una propuesta válida para funcionar sobre cualquier capa física 802.11. DQCA está basado en un protocolo MAC para entornos CDMA. Se trata de un protocolo distribuido de alta eficiencia que se mantiene estable para cualquier

condición de tráfico. Uno de los aspectos más relevantes del protocolo DQCA es que pasa de forma transparente y progresiva de un sistema del tipo Slotted-ALOHA de acceso libre para cargas de tráfico bajas (minimizando así el retardo), a un sistema de reserva cuando el tráfico es alto (evitando las colisiones y mejorando el caudal) de modo que se retienen las ventajas de ambos mecanismos. Para información complementaria sobre este protocolo mirar [7]