



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

**PROGRAMA DE MAESTRIA Y DOCTORADO
EN INGENIERÍA**

FACULTAD DE INGENIERÍA

**Análisis y diseño de una técnica de
calendarización para redes Wimax
móviles basadas en el estándar IEEE
802.16e**

T E S I S
QUE PARA OBTENER EL GRADO DE:
MAESTRO EN INGENIERIA
INGENIERIA ELÉCTRICA-TELECOMUNICACIONES
P R E S E N T A:

JAIME GONZALEZ MENDEZ

DIRECTOR DE TESIS:

DR. VICTOR RANGEL LICEA

Cd. Universitaria, México D.F. Agosto 2011





Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Jurado Asignado:

Presidente:	Dr. Gutiérrez Castrejón Ramón	Instituto de Ingeniería
Secretario:	Dr. Gómez Castellanos Javier	Facultad de Ingeniería
Vocal:	Dr. Rangel Licea Víctor	Facultad de Ingeniería
Primer Suplente:	Dr. Pascoe Chalke Michael	UAM, Iztapalapa
Segundo Suplente:	Dr. Benítez Pérez Héctor	IIMAS

Ciudad Universitaria, México D.F.

Tutor
Victor Rangel Licea

*A mis padres y mis hermanos,
como reconocimiento y gratitud a su apoyo incondicional.*

Ing. Jaime González Méndez

Agradecimientos

A la vida y a Dios, por darme la oportunidad y el tiempo de poder alcanzar las metas que me he propuesto.

A mis padres, por ser ejemplo de vida y de superación.

A mis hermanos, por su hacerme ver la vida desde diferentes ángulos.

A Alberto, Arturo, César, Oscar y Ruben, por no ser amigos, sino la familia que la vida me permitió elegir.

A mis amigos, por sus calidez, amistad y sobre todo por su confianza.

Al Dr. Víctor Rangel Licea, por su asesoría y paciencia durante la realización de este trabajo.

Al M. en C. Luis Manuel Ortiz Buenrostro, por su valioso apoyo en cuestiones técnicas durante la realización de este trabajo.

A mis sinodales, Dr. Ramón Gutiérrez Castrejón, Dr. Javier Gómez Castellanos, Dr. Michael Pascoe Chalke y Dr. Héctor Benítez Pérez, por el tiempo dedicado a la revisión de este trabajo.

A la CEP-UNAM, por la beca otorgada para la realización de mis estudios de maestría.

A la DGAPA-UNAM, por el apoyo recibido por parte de los proyectos de investigación PAPIIT No. IN108910: *Diseño de Algoritmos de reservación de capa cruzada en redes móviles y mesh de banda ancha*.

Al CONACYT, por el apoyo recibido del proyecto CONACYT 105279: *Diseño de técnicas de reservación de capacidad de redes BWA móviles*.

A la Universidad Nacional Autónoma de México, mi Alma Mater, por formarme como Maestro en Ingeniería, impulsar mi desarrollo académico y humano y darme la oportunidad de ser orgullosamente universitario.

Resumen

Se propone un algoritmo de calendarización de Otorgamientos por Estación Suscriptora (GPSS) para redes Wimax basadas en el estándar IEEE 802.16. Este diseño demuestra ser capaz de complementar y subsanar algunas limitaciones del algoritmo de calendarización propuesto por el software OPNET Modeler. La solución presentada considera la *latencia máxima* en la calendarización, como lo indica el estándar IEEE 802.16, y logra disminuir el retardo de encolamiento de las conexiones con tipo de calendarización rtPS diseñando un algoritmo calendarizador para la Estación Suscriptora (SS), introduciendo el concepto de *satisfacción del retardo* y creando una nueva forma de calcular el tiempo entre *polls* para el tipo de calendarización mencionado.

El algoritmo calendarizador GPSS propuesto en este trabajo permite que una SS, en una red Wimax, tenga la capacidad de reasignar los recursos proporcionados por la Estación Base (BS) como mejor le convenga en un momento determinado. Lo anterior implica que si una conexión debe enviar un paquete de inmediato porque éste está llegando a su tiempo límite (*deadline*), la SS distribuirá los recursos que le fueron concedidos de tal forma que dicho paquete sea enviado como se necesita con el fin de evitar el descarte. Este escenario conlleva a ofrecer una mejor Calidad de Servicio (QoS) en las conexiones sensibles al retardo, sin afectar significativamente a las demás conexiones.

Abstract

A Grant Per Subscriber Station (GPSS) scheduling algorithm for Wimax networks based on the IEEE 802.16e standard is proposed. This design is capable of overcoming some restraints of the scheduling algorithm proposed by OPNET Modeler. This solution considers *Maximum Latency* in scheduling, as IEEE 802.16 indicates, and achieves queuing delay reduction in rtPS connections by designing an scheduling algorithm for Subscriber Stations (SS). The concept of *delay satisfaction* and a new way of polls inter-arrival time computation is introduced in this algorithm.

The algorithm proposed allows an SS, in a Wimax network, to be able to reallocate the resources granted by the Base Station (BS) as it suits best. This means that if a connection must immediately send a packet because its deadline is approaching, the SS will manage the grant so that the packet would be sent as soon as possible in order to avoid dropping it. This scenario leads to provide a better Quality of Service (QoS) in non delay-tolerant connections, without damaging significantly the other active connections.

Índice general

1. Introducción	15
1.1. Antecedentes	15
1.2. Definición del Problema	17
1.3. Objetivos	18
1.3.1. General	18
1.3.2. Específicos	18
1.4. Contribuciones	18
1.5. Estructura de la Tesis	19
2. Conceptos Básicos de Redes y Estado del Arte	21
2.1. Conceptos Básicos de Redes	21
2.1.1. Redes e Internet	21
2.1.2. Clasificación de Redes	22
2.1.2.1. Topología	22
2.1.2.2. Cobertura geográfica	22
2.1.3. Modelo TCP/IP	23
2.1.3.1. Capa de Aplicación TCP/IP	24
2.1.3.2. Capa de Transporte TCP/IP	24
2.1.3.3. Capa de Internet	25
2.1.3.4. Capa de Acceso a la Red	25
2.1.3.5. Encapsulando Información	25
2.1.4. Modelo de Referencia OSI	26
2.2. Redes Inalámbricas de Banda Ancha	27
2.2.1. Arquitectura de las Redes BWA	28

2.2.2. Componentes de un sistema BWA	28
2.3. Estado del Arte	30
3. Descripción del Estándar IEEE 802.16e	33
3.1. Antecedentes	33
3.2. Topologías IEEE 802.16	34
3.3. Capa Física	35
3.3.1. Modulación Digital [21]	36
3.3.1.1. Binary Phase Shift Keying (BPSK)	36
3.3.1.2. Quadrature Phase Shift Keying (QPSK)	36
3.3.1.3. Quadrature Amplitude Modulation (QAM): 16-QAM y 64-QAM	37
3.3.2. Transmisión OFDM	38
3.3.3. IFFT y OFDM	38
3.3.4. Consideraciones de OFDM en el dominio del Tiempo [22]	39
3.3.5. Consideraciones de OFDM en el dominio de la Frecuencia [22]	41
3.3.6. Parámetros del símbolo OFDM	41
3.3.6.1. Duración de símbolo OFDM	42
3.3.6.2. Valores de la tasa de datos	42
3.3.7. Slot físico	43
3.3.8. Estructura de Frame y Slot	43
3.3.9. Codificación y modulación adaptiva	44
3.4. Capa MAC	45
3.4.1. MAC PDU: Construcción y transmisión	46
3.4.2. Calidad de Servicio (QoS)	50
3.4.2.1. Piggyback Request	50
3.4.2.2. Bandwidth Stealing	50
3.4.2.3. Polling	51
3.4.2.4. Unsolicited Grant Service (UGS)	51
3.4.2.5. Real Time Polling Service (rtPS)	52
3.4.2.6. Non Real Time Polling Service (nrtPS)	52
3.4.2.7. Best Effort (BE)	52
3.4.3. Solicitudes y asignación de ancho de banda	52

3.4.3.1.	Solicitudes de ancho de banda	53
3.4.3.2.	Modo de Asignación por Conexión (GPC)	53
3.4.3.3.	Modo de Asignación por Estación Suscriptor (GPSS)	53
3.4.3.4.	Consultas (Polling)	54
4.	Análisis de la Asignación de Recursos en una Red Wimax	55
4.1.	OPNET Modeler	55
4.2.	Elementos de una red Wimax móvil en OPNET	58
4.2.1.	Estación Suscriptor (SS)	58
4.2.1.1.	Procesos relevantes de la SS	58
4.2.2.	Estación Base (BS)	60
4.2.2.1.	Procesos relevantes de la BS	61
4.2.3.	Características de los modelos de proceso	62
4.3.	Solicitud de recursos en Wimax	64
4.3.1.	Un paquete llega de las capas superiores	65
4.3.1.1.	Directivas de entrada de <i>hl_pk</i>	65
4.3.2.	Solicitud de ancho de banda: Caso DL	67
4.3.2.1.	Directiva <i>BW_REQ_PROCESS</i>	68
4.3.3.	Solicitud de ancho de banda: Caso UL	68
4.3.3.1.	Llegada de un <i>poll</i> a la SS	69
4.3.3.2.	Directiva <i>ul_map_decode</i>	69
4.3.3.3.	Directiva de entrada de <i>tx_schedule</i>	69
4.3.3.4.	Solicitud Remota llega a la BS	70
4.3.3.5.	Directiva de entrada de <i>control_pk</i>	70
4.3.3.6.	Directiva <i>PACKET_PROCESS</i>	70
4.4.	Asignación de recursos en Wimax	71
4.4.1.	Creación del mapa	71
4.4.1.1.	Directivas <i>MAP_GENERATE</i>	72
4.4.1.2.	Inserción de otorgamientos periódicos: Polls rtPS y nrtPS, ertPS y UGS	72
4.4.1.3.	Inserción de solicitudes de rtPS, nrtPS y BE	74
4.4.2.	Función <i>wimax_bs_control_sched_bw_req_dequeue ()</i>	75

4.4.2.1.	Sistema Calendarizador de la BS	75
4.4.2.2.	Descripción de la función	77
4.4.3.	Particularidades y Limitaciones del Sistema Calendarizador de OPNET	78
4.4.3.1.	Cálculo del <i>peso</i> en MDRR	78
4.4.3.2.	rtPS y nrtPS con igual prioridad	79
4.4.3.3.	Sistema Calendarizador GPSS Inexistente en la SS	79
5.	Diseño y Simulación de una Técnica de Calendarización	81
5.1.	Diseño de un algoritmo Calendarizador GPSS en la SS	81
5.1.1.	Introducción del concepto <i>Satisfacción del retardo</i>	82
5.1.2.	Descripción del Algoritmo	82
5.1.3.	<i>Tiempo entre polls</i> en conexiones rtPS	83
5.1.3.1.	<i>Tiempo entre polls</i> en OPNET	83
5.1.3.2.	Nuevo cálculo para el <i>tiempo entre polls</i>	85
5.2.	Escenario de Simulación	86
5.2.1.	Escenario A: Aplicaciones diversas en cada SS	86
5.2.2.	Escenario B: VoIP en BE, nrtPS y rtPS	88
5.3.	Resultados	88
5.3.1.	Caracterización de la Red	90
5.3.1.1.	Carga y Tráfico Enviado	90
5.3.1.2.	Throughput y Tráfico Recibido	92
5.3.1.3.	Tráfico Descartado	92
5.3.1.4.	Espacio Libre en el Frame	93
5.3.2.	Escenario A	93
5.3.3.	Escenario B	97
5.3.3.1.	Carga	97
5.3.3.2.	Tráfico enviado y descartado	98
5.3.3.3.	Retardo de encolamiento	99
5.3.4.	Comparación Final	99
5.3.4.1.	Retardo y Tráfico Descartado	99
5.3.4.2.	Carga y Tráfico Enviado	100

6. Conclusiones	103
6.1. Discusiones Finales	103
6.2. Contribuciones	103
6.3. Trabajo a Futuro	104
A. Código del Nuevo Algoritmo calendarizador GPSS	107
A.1. Función <i>wimax_ss_control_gpss_ie_decompose()</i>	107
A.2. Función <i>wimax_ss_control_one_svc_flow_parse ()</i> – Fragmento	110
B. Diagramas de Árbol	113
B.1. Solicitud de ancho de banda para conexiones DL	113
B.2. Solicitud de ancho de banda para conexiones UL	116
B.2.1. Un paquete de capas superiores llega a la capa MAC	116
B.2.2. Un <i>poll</i> llega desde la BS	118
B.2.3. La solicitud de ancho de banda llega a la BS	124
C. Lista de Acrónimos	129

Capítulo 1

Introducción

1.1. Antecedentes

Entre todos los elementos esenciales para la existencia humana, la necesidad de relacionarse con otros se ubica sólo por debajo de las necesidades que preservan la vida. La comunicación es casi tan importante para nosotros como la necesidad de aire, comida y hogar.

Día con día, la necesidad de mantenerse comunicado se hace más importante. Con el crecimiento de la población que accede a las telecomunicaciones, la demanda de aplicaciones de banda ancha con mayor tecnología y eficiencia aumenta de forma acelerada, tanto en la redes cableadas como en la tecnología inalámbrica.

En el ambiente inalámbrico y móvil, muchas tecnologías han sido propuestas con el fin de afrontar este gran desafío; CDMA2000-EDVO [1] (Code Division Multiple Access - Evolution-Data/Voice)¹, IEEE 802.11 [2] WLAN (Wireless Local Area Network) e IEEE 802.16 [3] Wimax (Worldwide Interoperability for Microwave Access) son ejemplo de ellas. Las redes inalámbricas se han convertido en el nuevo atractivo, gracias a su relativamente rápida implementación, inversión inicial pequeña, alta escalabilidad, bajos costos de mantenimiento y actualización, además de una inversión que se ajusta al crecimiento del mercado.

Aunado a la necesidad de sistemas con banda ancha, aparece un nuevo requerimiento: La integración de diferentes servicios. A través de una sola red debe ser posible transmitir de forma correcta diferentes servicios como video, voz, datos e Internet. Entre las tecnologías de próxima generación propuestas, Wimax fue diseñada para satisfacer este nuevo reto en redes inalámbricas.

Tecnologías como WLAN fueron diseñadas para reemplazar a Ethernet en situaciones donde es imposible instalar un cableado de red. El estándar IEEE 802.11, de donde proviene WLAN, no incluye el concepto de Calidad de Servicio (QoS) y su falta de soporte en movilidad limita su aplicación en escenarios donde esa característica es requerida. Por otro lado, las redes de telefonía celular se enfocan en proveer servicio de voz y movilidad. El inconveniente más importante que estas redes presentan es su limitada tasa de datos, alrededor de 10[Mbps] o menor.

¹Consulte la lista completa de acrónimos que se proporciona al final de este escrito.

En 1999, un grupo de trabajo del Instituto de Ingenieros Eléctricos Electrónicos (IEEE, Institute of Electrical and Electronics Engineers) publicó el primer estándar 802.16 para el despliegue global de redes inalámbricas con banda ancha de área metropolitana. A la fecha el estándar ha sufrido modificaciones y ahora podría ser considerado una familia de estándares.

Aunque en la IEEE la familia de estándares 802.16 es denominada oficialmente como WirelessMAN, ésta es mejor conocida comercialmente como Wimax, impulsada por una alianza de empresas llamada Wimax Forum, la cual tiene por objetivo promover y certificar la compatibilidad e interoperabilidad de productos inalámbricos de banda ancha basados en el estándar 802.16.

La implementación más popular del estándar IEEE 802.16 es el Mobile WirelessMAN, originalmente definido en la enmienda 802.16e-2005. Esta versión está actualmente en desarrollo alrededor del mundo en más de 140 países con la participación de más de 450 operadores de este tipo de sistemas. Entre otras cosas, esta enmienda ofrece soporte a la movilidad en redes inalámbricas de banda ancha, es por eso que su implementación es mejor conocida como "Mobile Wimax".

La tecnología Wimax combina efectivamente las ventajas de las dos tecnologías descritas previamente, WLAN y redes inalámbricas celulares, ofreciendo soporte a la movilidad, tasa de datos alta, integración de diferentes servicios y además garantiza Calidad de Servicio. Entre las características de Wimax, la integración de servicios y soporte en la calidad de servicio para dichos servicios son los rasgos que más diferencian a Wimax de otros tecnologías en redes.

Debido a sus características más notables, el estándar IEEE 802.16-Wimax es considerado un sistema de Acceso a Banda Ancha Inalámbrica (BWA, Broadband Wireless Access), categoría a la que también pertenece el estándar ETSI BRAN (European Telecommunications Standards Institute Broadband Radio Access Networks). El estándar IEEE 802.16, es mucho más robusto que el ETSI BRAN al ser un estándar que se deriva del protocolo americano de televisión por cable para la transmisión de datos (DOCSIS, Data Over Cable Service Interface Specification) [4].

El Foro WIMAX, donde participan más de 400 empresas, promueve la adopción del protocolo IEEE 802.16 por los operadores comerciales de sistemas de acceso inalámbrico de banda ancha y el uso de estos sistemas sobre los ya existentes. El Foro Wimax trabaja de cerca con los proveedores de servicio y con los reguladores para asegurar que los sistemas certificados por este foro reúnan los requerimientos impuestos por el gobierno y los consumidores de los diferentes países.

WiMAX busca proveer sistemas de banda ancha inalámbricos fijos y móviles, sin la necesidad de línea de vista directa a la estación base, con un radio típico que puede alcanzar hasta 50 kilómetros, con una capacidad de más de 40[Mbps] por canal para aplicaciones de acceso fijo y móvil. Este ancho de banda es suficiente para soportar al mismo tiempo cientos de negocios con una velocidad de conexión similar a la de un enlace E1 (2,048[Mbps]) y a miles de hogares con una velocidad de conexión semejante a la de un DSL (Digital Subscriber Line).

1.2. Definición del Problema

La versión e del estándar IEEE 802.16, publicado en 2005, propone una tecnología para el acceso a sistemas inalámbricos de banda ancha para usuarios móviles. Wimax Móvil (o IEEE 802.16e) permite el uso de diferentes órdenes de modulación, lo que se refleja en mayores tasas de transmisión. Generalmente, la modulación está directamente asociada a la calidad del canal por donde se transmite la información y es a partir de este parámetro que una de las modulaciones disponibles es seleccionada y utilizada para transmitir la información, de ahí el término "Modulación Adaptiva".

Las modulaciones soportadas comúnmente por la versión móvil del estándar 802.16 son: BPSK(1/2) , QPSK(1/2), QPSK(3/4), 16-QAM(1/2), 16-QAM(3/4) 64-QAM(2/3) y 64-QAM(3/4). Usualmente una modulación de orden bajo, BPSK por ejemplo, ofrece una ancho de banda bajo pero una cobertura bastante amplia. Por otra parte, las modulaciones de orden elevado ofrecen una tasa de transmisión alta pero una cobertura limitada. Un problema importante en el uso de la modulación adaptiva es que una modulación de orden alto, como 64-QAM, provoca alta interferencia entre celdas e impide el re-utilización adecuado de frecuencia.

Otra característica importante de WIMAX, es el concepto de Calidad de Servicio (QoS); la asignación de recursos en la capa de Control de Acceso al Medio (MAC, Media Access Control) es decisiva para asegurar QoS, sin embargo, el estándar Wimax no define algoritmo de encolamiento alguno. No obstante, algunos algoritmos al respecto han sido propuestos, [9] y [10] son ejemplos de ello.

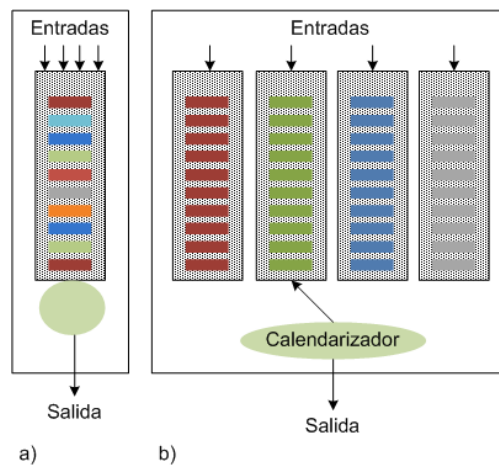


Figura 1.1: a)Esquema FCFS b)Esquema usando Calendarización

Una red Wimax es capaz de ofrecer QoS debido a que cuenta con calendarizadores. Un calendarizador permite compartir de una forma conveniente el ancho de banda de una red. Tradicionalmente, el esquema FCFS (First Come First Served) era usado para la calendarización de paquetes. Los paquetes provenientes de todas las fuentes eran enfilados en una cola FIFO (First In First Out) para luego ser servidos uno por uno hacia el enlace de salida, como se muestra en figura 1.1a. Hoy en día, con el fin de clasificar los paquetes, se especifican colas diferentes para paquetes diferentes. De este modo, se logra la calendarización. La principal

tarea de un algoritmo de calendarización es escoger el próximo paquete a ser desencolado de las múltiples colas disponibles y remitirlo al enlace de salida, como se observa en la figura 1.1b.

El presente trabajo de investigación pretende analizar una técnica de calendarización para redes Wimax, particularmente, la utilizada por el software OPNET Modeler con el fin de conocer una forma de asignación de recursos. Dicho análisis conducirá a diseñar un algoritmo calendarizador que mejore la QoS en una red Wimax, dando prioridad a aquellas conexiones que sean vulnerables al retardo. Opnet Modeler permitirá también simular el nuevo calendarizador en un red Wimax móvil con usuarios moviéndose en un sola celda.

1.3. Objetivos

1.3.1. General

La presente tesis tiene como objetivo general:

Analizar y diseñar una técnica de calendarización para redes Wimax móviles, con usuarios circulando en una sola celda, a partir del diseño propuesto por el software OPNET Modeler.

1.3.2. Específicos

El objetivo general incluye los siguientes objetivos particulares:

- Analizar los procedimientos de solicitud y asignación de recursos en una red Wimax.
- Diseñar un algoritmo de calendarización para redes Wimax que dé mayor prioridad a las aplicaciones en tiempo real sobre las que no lo son.
- Simular el diseño propuesto en el software OPNET Modeler para estudiar su posible implementación.
- Presentar resultados que evidencien una mejora en la red Wimax causada por el diseño propuesto en términos de: Tasa de pérdidas de paquetes, rendimiento promedio, retardo total y retardo de encolamiento.

1.4. Contribuciones

Se sabe que Wimax ha emergido como uno de los candidatos más fuertes para proveer servicio inalámbrico y móvil de banda ancha a usuarios y que la Calidad de Servicio es una de sus virtudes más importantes.

Recientes investigaciones aseguran que la asignación de recursos en la capa de enlace de datos, específicamente la subcapa de Control de Acceso al Medio, es crucial para asegurar y

garantizar Calidad de Servicio. Es ahí donde el trabajo aquí presentado ofrece contribuciones. Con el diseño propuesto en este documento, la QoS se garantiza de manera más eficiente, lo que se refleja en una mejor transmisión de paquetes pertenecientes a aplicaciones sensibles al retardo.

Los resultados obtenidos del trabajo de investigación aquí descrito, evidencian optimización en el retardo de encolamiento en la capa MAC, que sin duda representarán un mejor desempeño de la Calidad de Servicio de una red Wimax. Finalmente, la garantía de una QoS adecuada es de suma importancia para empresas que ofrecen aplicaciones en tiempo real donde el retardo es un factor importante, ejemplo de ellas son: Video Chat, Servicio de Mensajería Multimedia (MMS, Multimedia Messaging Service), Televisión en alta definición, Televisión Móvil y Difusión de Video Digital (DVB, Digital Video Broadcasting) y Voz sobre IP (VoIP, Voice over IP).

Finalmente, este trabajo contribuirá en el avance de la investigación acerca de la implementación de nuevos algoritmos de encolamiento que proporcionen mejores desempeños en términos de tasa de pérdida de paquetes, rendimiento promedio y equidad. Esta línea de investigación es ampliamente estudiada actualmente debido al impacto que tienen las redes Wimax, gracias a sus múltiples beneficios.

1.5. Estructura de la Tesis

La presente tesis está conformada por 6 capítulos que se describen brevemente a continuación.

En el Capítulo 1 se presenta la introducción al trabajo de investigación realizado. Se da una perspectiva general del concepto de redes inalámbricas de banda ancha, los estándares y las tecnologías relacionadas. Asimismo, se plantea el problema a solucionar y los objetivos a alcanzar en este trabajo de investigación. Finalmente, se explican las contribuciones que esta tesis proporciona.

El Capítulo 2 muestra un panorama general de los conceptos básicos relacionados con el tema tratado en la tesis, los cuales serán utilizados a lo largo del presente documento. De igual manera, se presentan los últimos trabajos de investigación realizados que guardan relación con el diseño de algoritmos de calendarización en redes inalámbricas.

Durante el Capítulo 3 se hace una descripción detallada de las secciones del estándar IEEE802.16e que resultan de gran importancia para el desarrollo del presente trabajo. La capa Física es descrita con detalle con la intención de explicar con precisión las características de cada una de las modulaciones disponibles que el estándar indica. Para la subcapa MAC, también se hace una descripción detallada debido a que es ahí donde se realiza el proceso de encolamiento y asignación de recursos, conceptos de suma importancia para este trabajo.

En el Capítulo 4 se realiza la descripción y el análisis de la solicitud y asignación de ancho de banda en un red Wimax. Se describe también al algoritmo calendarizador propuesto por OPNET Modeler y se discuten sus particularidades y limitaciones. Este capítulo es crucial

para poder entender la función de un calendarizador y estar en condiciones de diseñar uno nuevo.

El Capítulo 5 presenta el nuevo diseño de calendarizador para redes Wimax, subsanando las debilidades del algoritmo propuesto por OPNET Modeler. También, se abordan aspectos relacionado con las simulaciones realizadas para probar el nuevo algoritmo con la ayuda del software ya mencionado. Se describen las condiciones bajo las cuales dichas simulaciones fue realizadas y los resultados obtenidos a partir de éstas. Los resultados de las simulaciones son confrontados con el fin de evidenciar los beneficios obtenidos con el nuevo algoritmo calendarizador.

Finalmente, en el Capítulo 6 se dan a conocer las conclusiones a las que se llegaron una vez que el algoritmo de calendarización propuesto fue implementado y simulado. Con el fin de dar continuidad al trabajo de investigación aquí descrito, al final de este capítulo se presentan posibles trabajos futuros a realizar a partir de los resultados proporcionados por esta tesis.

Conceptos Básicos de Redes y Estado del Arte

Este capítulo tiene el propósito de proporcionar al lector un panorama general relacionado con las redes BWA. La primera parte de este capítulo presenta una explicación ligeramente detallada de algunos conceptos importantes que guardan relación con la redes y sobre todo con las redes inalámbricas de banda ancha, área de aplicación de esta tesis. La segunda y última parte expone el estado del arte, es decir las últimas investigaciones y/o publicaciones realizadas con respecto al encolamiento y diseño calendarizadores en la estación base y/o en la estación suscriptora, temas específicos de este trabajo.

2.1. Conceptos Básicos de Redes

Los métodos usados para compartir nuestras ideas e información cambian constantemente. Desde la prensa escrita hasta la televisión, cada nuevo desarrollo ha mejorado nuestra comunicación. La creación e interconexión de redes de datos ha tenido un efecto sumamente importante. Esta plataforma nos proporciona acceso una gama importante de métodos de comunicación que permiten a la gente poder interactuar con otros, de forma prácticamente instantánea.

2.1.1. Redes e Internet

En términos generales, una red de transmisión de datos es una sistema formado por determinados medios físicos y lógicos diseñado para satisfacer las necesidades de comunicación. Se trata entonces, de un soporte que permite la interconexión de diversos equipos informáticos con el objetivos de suministrarles la posibilidad de intercambiar información [5].

Actualmente, es común que cuando un estudiante asiste a la escuela éste se conecte a un red interna la cual, a su vez, está conectada a Internet. Para comunicarse, muchas redes de empresas también se conectan a Internet. Internet es una gran red mundial formada por una multitud de redes conectadas unas con otras de forma que hace posible el intercambio de información entre ellas. Las redes con acceso a Internet se conectan a éste a través de un

Proveedor de Servicios de Internet (ISP, Internet Service Provider), Internet es en realidad un colección de ISPs que se interconectan con otros.

2.1.2. Clasificación de Redes

Existen múltiples redes, cada una de ellas con características específicas que la hacen diferente del resto. Es posible clasificar a las redes atendiendo a diferentes criterios. Las clasificaciones que se presentan a continuación obedecen a dos de los criterios más importantes.

2.1.2.1. Topología

Esta clasificación considera la arquitectura de la red, es decir la forma en la que los elementos de la red se encuentran conectados o dispuestos físicamente.

- **Bus:** Esta es una topología de tipo lineal en la cual todos los dispositivos están conectados por medio de un solo un cable. El inconveniente de esta topología es que si algún enlace falla, todos los elementos de la red quedan aislados.
- **Anillo:** En este tipo de topología todos los nodos están conectados a una única vía que tiene los dos extremos unidos, esto quiere decir que cada dispositivo se encuentra directamente conectado a otros dos vecinos, de tal manera que la señal que se transmite es repetida en una sola dirección creando un loop o anillo. Al igual que con la topología de bus, en un anillo, cuando un enlace falla, la comunicación se pierde en cada uno de los elementos.
- **Estrella:** En una estrella, los dispositivos se conectan a un nodo central que es utilizado para distribución, conmutación y control. En este caso si el nodo central falla, la red queda aislada por completo. Sin embargo, si la falla ocurre en algún otro dispositivo, sólo ese dispositivo quedará incomunicado.
- **Malla:** Una topología de malla proporciona a cada dispositivo, una conexión directa con cada uno de los dispositivos restantes. Debido al gran consumo de cable en este tipo de redes, su costo es muy elevado.

La figura 2.1 muestra gráficamente ejemplos de las topologías de redes más comunes.

2.1.2.2. Cobertura geográfica

La cobertura es un factor importante en el diseño e implementación de una red. Es claro que no es lo mismo montar una red para un salón de clases que para interconectar las oficinas de dos sucursales que una misma empresa tiene instaladas en diferentes países. La clasificación de la redes de acuerdo a la cobertura se presenta a continuación.

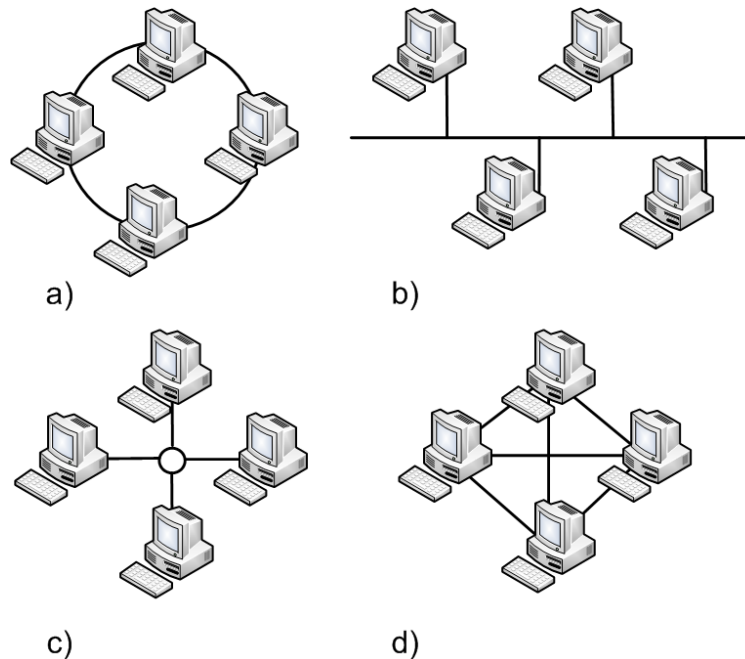


Figura 2.1: Clasificación de redes según la topología

- Red de Área Local (LAN, Local Area Network):** Una LAN permite la conexión de múltiples usuarios, en una área geográfica relativamente pequeña, para intercambiar archivos y mensajes, así como acceder a recursos compartidos como una impresora o archivos en un servidor. Generalmente se consideran dentro de esta categoría las redes cuyas dimensiones no sobrepasan un mismo edificio.
- Redes de Área Metropolitana (MAN, Metropolitan Area Network):** Se les denomina así a las redes que se extienden entre edificios y que tiene cobertura en una misma ciudad. Una red Wimax es un ejemplo de red MAN.
- Redes de Área Amplia (WAN, Wide Area Network):** Una WAN abarca varias ciudades, regiones e incluso países. Las WAN interconectan LANs que se encuentran geográficamente dispersas. Los enlaces WAN son ofrecidos generalmente por empresas de telecomunicaciones que utilizan enlaces de microondas, fibra óptica o enlaces satelitales. Se dice que una red WAN es la interconexión de varias redes MAN.

2.1.3. Modelo TCP/IP

Los servicios de comunicaciones proporcionados por una red de transmisión de datos siguen protocolos bien establecidos y estandarizados. Si a una red en particular se le desea añadir una funcionalidad, se deberá comprobar si ya posee el protocolo adecuado o es necesario agregarlo. Un protocolo de red define normas a seguir en el momento de transmitir la información, estas normas pueden ser la velocidad de transmisión, el tipo de transmisión, el formato de mensaje, etc.

Capa TCP/IP	Ejemplo de Protocolos
Aplicación	HTTP, POP3, SMTP
Transporte	TCP, UDP
Internet	IP
Acceso a la Red	Ethernet, Frame Relay

Tabla 2.1: Modelo TCP/IP y ejemplos de protocolos

Actualmente, resulta prácticamente imposible encontrar una computadora que no soporte el conjunto de protocolos llamado TCP/IP (Transfer Control Protocol/Internet Protocol), los sistemas operativos de Microsoft, Linux e UNIX soportan el protocolo TCP/IP.

Al modelo TCP/IP, como también se le denomina, define una larga colección de protocolos que permiten la comunicación entre computadoras. TCP/IP describe los detalles de cada uno de los protocolos que lo conforman en documentos llamados RFCs (Request for Comments). Implementando los protocolos definidos y requeridos en los RFCs de TCP/IP, una computadora es capaz de comunicarse con otra que también tenga implementado TCP/IP.

Como lo hacen otras arquitecturas de interconexión de redes, TCP/IP clasifica los diversos protocolos en diferentes categorías llamadas capas, La tabla 2.1 muestra las capas del modelo de red TCP/IP.

El modelo TCP/IP representado en la columna 1 de la tabla 2.1 enlista las cuatro capas de TCP/IP, la columna 2 muestra algunos de los protocolos TCP/IP más populares.

2.1.3.1. Capa de Aplicación TCP/IP

La capa de aplicación TCP/IP provee servicios al software que se ejecuta en una computadora. En realidad, la capa de aplicación no define la aplicación. Sin embargo, define los servicios que las aplicaciones necesitan, por ejemplo la habilidad para transferir archivos en el caso de HTTP (Hypertext Transfer Protocol). En resumen, la capa de aplicación provee una interfaz entre el software ejecutándose en una computadora y la red misma.

2.1.3.2. Capa de Transporte TCP/IP

En un dispositivo cualquiera, es posible usar, al mismo tiempo, múltiples servicios como e-mail, web y mensajería instantánea. La información de estas aplicaciones es empaquetada, transportada y entregada a la aplicación apropiada en el dispositivo del destino. La capa de transporte acepta datos de la capa de aplicación y los prepara para las capas inferiores.

La capa de transporte se encarga principalmente del rastreo de la comunicación individual entre aplicaciones en los dispositivos fuente y de destino, la segmentación de información y el manejo, el re-ensamble de cada segmento en un flujo de datos de aplicación, así como, la identificación de las diferentes aplicaciones.

La capa de transporte TCP/IP puede contener uno de los siguientes protocolos: TCP (Transmission Control Protocol) y UDP (User Datagram Protocol). El protocolo TCP se utiliza en transferencias de datos *orientados a conexión* y el protocolo UDP en transferencia de datos *no orientados a conexión*. El término *conexión* se refiere al establecimiento previo de un enlace lógico por donde fluyan los datos.

2.1.3.3. Capa de Internet

La capa de Internet provee servicios para intercambiar los segmentos de datos, proporcionados por la capa de transporte, a través de una red.

La capa de Internet del modelo TCP/IP, IP (Internet Protocol), funciona como el servicio postal. IP define direcciones para que cada computadora pueda tener una única dirección IP. Después, IP define el proceso de enrutamiento a utilizar para la entrega de la información, como consecuencia, los dispositivos denominados *routers* pueden escoger hacia donde enviar los paquetes de información y éstos últimos pueden ser entregados en el destino correcto.

2.1.3.4. Capa de Acceso a la Red

La capa de acceso a la red define los protocolos y el hardware que se requieren para enviar datos a través de una red física. Esta capa define como conectar un dispositivo a una red.

Por ejemplo, Ethernet es un protocolo de la capa de acceso a la red. Ethernet define el cableado, direccionamiento y protocolos requeridos para crear una LAN Ethernet. Asimismo, los conectores, cables, niveles de voltaje y protocolos usados para transmitir datos a través de enlaces WAN, son definidos en una variedad de protocolos que también son considerados como parte de la capa de acceso a la red.

2.1.3.5. Encapsulando Información

Encapsular describe el proceso de colocar encabezados y colas (trailer) alrededor de los datos. Un dispositivo que necesita enviar datos, los encapsula con encabezados del formato correcto, así el dispositivo receptor sabrá como interpretar los datos recibidos. A estos bloques de información se les denomina unidades de datos de protocolo (PDU, Protocol Data Unit).

El proceso completo de encapsulamiento de datos con TCP/IP es un proceso que consta de 5 pasos descritos a continuación:

1. Crear los datos de aplicación y los encabezados. Esto significa que la aplicación tiene datos para enviar.
2. Empaquetar lo datos para ser transportados. Es decir, la capa de transporte crea el encabezado de transporte y lo coloca delante de los datos, generando la PDU denominada *segmento*.

3. Incluir a los datos las direcciones de destino y fuente de la capa de Internet. Esta capa crea el encabezado correspondiente, el cuál incluye las direcciones mencionadas y lo coloca delante de los datos provenientes de la capa de transporte, dando origen a la PDU denominada *datagrama*
4. Incluir a los datos las direcciones de fuente y destino de la capa de acceso a la red. Esta capa crea el encabezado correspondiente y lo coloca delante de lo datos que recibe de la capa superior, dando origen a la PDU denominada *trama (frame)*.
5. Transmitir bits. En este punto, se codifica una señal sobre un medio de transmisión para la correcta transmisión de bits.

La figura 2.2 puede ayudar a entender un poco mejor el proceso de encapsulamiento en TCP/IP. Cuando cada capa encapsula los datos que recibió de la capa inmediata superior, los detalles de los datos no son importantes para la capa actual, es decir lo que se recibe es sólo considerado como datos. La figura 2.3 muestra la perspectiva de las capas de transporte, Internet, y acceso a la red con respecto al encapsulamiento de datos. También es importante hacer notar que la información toma diferentes nombres dependiendo de la capa del modelo TCP/IP en la que se encuentre. Por ejemplo, el término segmento incluye el encabezado TCP o UDP pero no el encabezado de IP o el de la capa de acceso a la red.

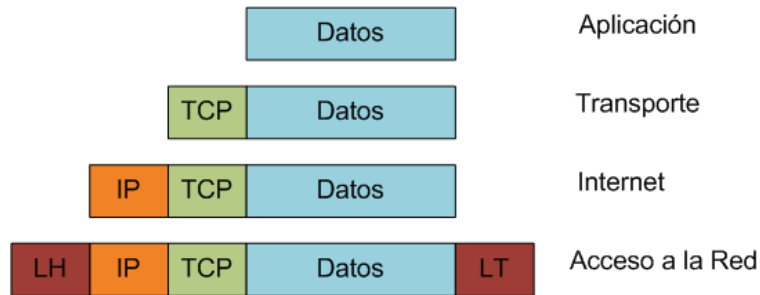


Figura 2.2: Encapsulamiento de Datos: TCP/IP



Figura 2.3: Perspectiva de encapsulamiento en cada Capa

2.1.4. Modelo de Referencia OSI

El modelo OSI (Open System Interconnection) es utilizado principalmente como punto de referencia para discutir especificaciones de otros protocolos.

Nombre de la Capa	Ejemplo de Protocolos
Aplicación (Capa 7)	Telnet, HTTP, FTP, NFS, SMTP
Presentación (Capa 6)	JPEG, ASCII, TIFF, GIF
Sesión (Capa 5)	PRC, SQL, AppleTalk
Transporte (Capa 4)	TCP, UDP, SPX
Red (Capa 3)	IP, IPv6, IPX, AppleTalk DDP
Enlace de Datos (Capa 2)	IEEE 802.3/802.2, HDLC, Frame Relay, PPP, FDDI
Física (Capa 2)	EIA/TIA-232, V.35, RJ-45, Ethernet

Tabla 2.2: Modelo de Referencia OSI y ejemplos de protocolos

El modelo OSI consta de siete capas. Cada capa define un conjunto específico de funciones de red. Hoy en día, el modelo OSI puede ser usado como estándar para hacer comparaciones con otro modelos de red. La figura 2.4 muestra la comparación entre los modelos OSI y TCP/IP

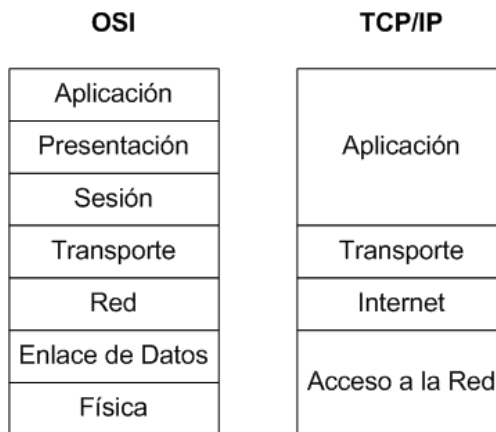


Figura 2.4: Comparación OSI - TCP/IP

Las capas superiores del modelo OSI (Aplicación, Presentación y Sesión - Capas 7, 6, y 5) definen funciones enfocadas a la aplicación. La cuatro capas inferiores (Trasporte, Red, Enlace de Datos y Física - Capas 3, 2 y 1) definen funciones enfocadas en la entrega de datos de extremo a extremo. La tabla 2.2 presenta ejemplos de protocolos concernientes a cada una de las capas del modelo OSI.

2.2. Redes Inalámbricas de Banda Ancha

Se denomina Banda Ancha Inalámbrica a la capacidad de transmitir gran cantidad de información de un punto a otro a través del espectro radioeléctrico, es decir de forma inalámbrica. El BWA es un sistema de transmisión que permite el envío de video, voz, datos e Internet a una alta velocidad. El término banda ancha se refiere a velocidades

	PTP(Punto-Punto)	PMP (Punto-Multipunto)
Número de Puntos	Dos Puntos	Múltiples puntos
Espectro y Banda	Dedicados	Compartidos
Antenas	Haz angosto	Haz angosto
Tráfico transmitido sobre	Un canal fijo libre	Un canal compartido

Tabla 2.3: Principales Características de los enlaces PTP y PMP

de transmisión mayores a $1,5[Mbps]$, incluso existen redes BWA que soporten velocidades significativamente más altas.

2.2.1. Arquitectura de las Redes BWA

Generalmente, las redes inalámbricas de banda ancha consisten de enlaces de radio Punto-Punto (PTP) para transportar tráfico y de enlaces Punto-Multipunto (PMP) para entregar información al destino final (Figura 2.5).

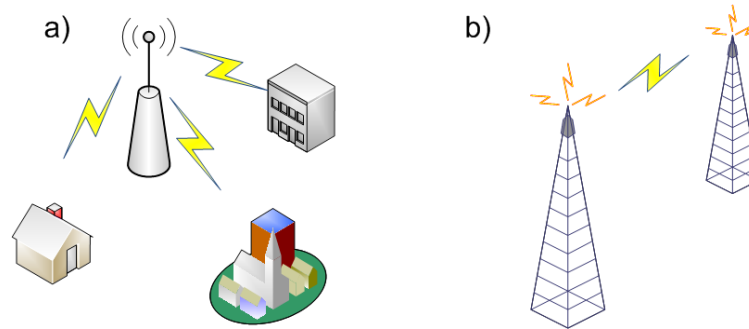


Figura 2.5: Enlaces a)Punto-Multipunto y b)Punto-Punto

De acuerdo a las recomendaciones de la Unión Internacional de Telecomunicaciones (ITU, International Telecommunications Union) ITU-R F-1499, que pertenece a la especificación de los Sistemas de Acceso Inalámbrico de Banda Ancha, es posible mantener transmisiones desde $1,5[Mbps]$ hasta $45[Mbps]$. Las bandas electromagnéticas utilizadas por los sistemas BWA son variadas y regularmente se localizan entre los $2,5$ y $66[GHz]$. La tabla 2.3 muestra las principales características que poseen los enlaces de radios PMP y PTP.

2.2.2. Componentes de un sistema BWA

Generalmente, un sistema BWA se compone de al menos una *Estación Base* (BS, Base Station), también llamada *Radio Base* y una o más *Estaciones suscriptoras* (SS, Subscriber Stations). En la figura 2.6 se puede observar como una misma radio base puede dar servicio a hogares y empresas al mismo tiempo. La BS está generalmente localizada sobre un torre y las SSs usualmente incluyen antenas montadas sobre un techo y unidades de radio conectadas a interfaces de redes en el interior de un edificio o casa.

La estación base o estación central puede conectarse a Internet a través de fibra óptica. La BS es un dispositivo que interactúa directamente con la estación suscriptor. En la SS, el módem actúa como puerto de datos, el cual transporta los datos, de nuevo, por un medio alámbrico.

Tomando como referencia al Modelo OSI, las capas Física y Enlace de datos de una red BWA la distinguen de otras redes. Estas capas especifican los métodos de comunicación y requerimientos entre la SS y la BS.

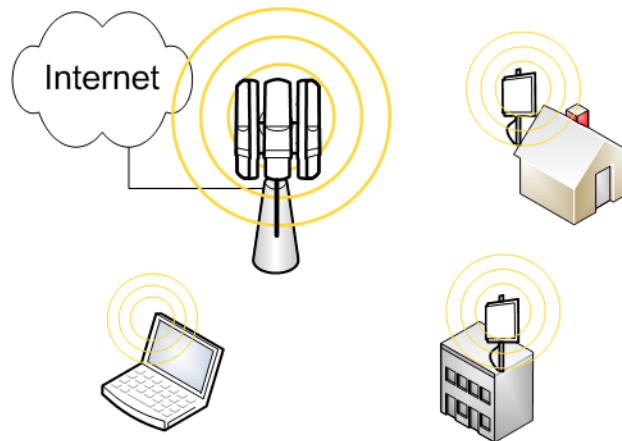


Figura 2.6: Componentes de una red BWA

Un sistema BWA sigue el estándar DOCSIS de forma extensiva, con modificaciones para la transmisión de forma inalámbrica. Se usan diferentes formatos de señalización para el envío de datos, FDD (Frequency Division Duplex) y TDD (Time Division Duplex) son los más comunes.

La BS asigna un canal a las SS de acuerdo a la políticas de control de acceso al medio. Todos los sistemas BWA constan de tres subsistemas, descritos a continuación:

- **Sistemas de Acceso:** Este sistema es el responsable de la estructuración y la administración de tráfico en las dos direcciones posibles, ascendente y descendente. En una red BWA, las SS comparten los canales de subida UL(UpLink) y de bajada DL (DownLink).
- **Sistemas de Radiofrecuencia:** Se le denomina así a la parte inalámbrica de alta capacidad que permite la transmisión y recepción de señales con cada SS. En la BS, los módulos de transmisión y recepción están conectados, por un lado, con el equipo de acceso de datos de la BS y, por otro lado, con una antena cuyas características dependen de los requisitos del sistema. En la SS, un módulo transmisor/receptor recibe la señal de bajada de la BS y la envía al módem de la SS. Ese mismo módulo, recibe el tráfico de la SS y lo transmite a la BS.
- **Interfaz de red en la BS:** Representa la conexión entre el proveedor de servicios y la red de Internet, acceso de alta velocidad a servicios multimedia y a la Red Telefónica Pública Conmutada (PSTN, Public Switched Telephone Network) para proporcionar VoIP.

2.3. Estado del Arte

Como ya se ha mencionado, la asignación de recursos en la capa MAC es decisiva para asegurar QoS en una red Wimax o Wimax Móvil. No obstante, el estándar IEEE 802.16 no define algoritmo de encolamiento alguno. Aunque muchos algoritmos de calendarización están disponibles para redes inalámbricas, éstos no pueden ser directamente aplicados a redes inalámbricas debido a las diferencias fundamentales entre estos tipos de redes. Por ejemplo, la capacidad del canal en redes inalámbricas varía con el tiempo debido a los desvanecimientos por multitrayectorias y al efecto Doppler. En ocasiones, aunque se asigne una porción grande de ancho de banda a cierta conexión, el retardo máximo o el rendimiento pueden no satisfacerse y el ancho de banda asignado será desperdiciado cuando el canal inalámbrico presente desvanecimientos profundos [6].

Una perspectiva general de las técnicas de calendarización puede ser encontrada en [7], donde se resume una lista de características deseables en un calendarizador y varias clases de ellos son comparados con base en dichas características. Del mismo modo, en [8] se discuten los asuntos claves y factores de diseño a tomar en cuenta para el diseño de calendarizadores y se presenta un estudio extensivo acerca de la investigación en el campo de los calendarizadores.

Como el estándar IEEE 802.16 no propone ningún algoritmo calendarizador, varios algoritmos al respecto han sido propuestos, [9] y [10] son ejemplo de ellos. En [9] se propone un encolamiento con un enfoque de capa cruzada, donde la administración de los paquetes se hace de acuerdo a una función de prioridad definida para cada conexión, esta función de prioridad incorpora la información sobre el canal físico para lograr eficiencia en el ancho de banda. En [10] se propone un encolamiento equitativo y efectivo, el cual tiene buen rendimiento en términos de equidad y de tasa de paquetes bloqueados con el beneficio de contar con algoritmo de baja complejidad. WFQ (Weighted Fair Queueing) para redes Wimax es propuesto en [11]. Aunque WFQ puede garantizar la tasa mínima reservada de cada conexión, éste no controla los requerimientos de retardo y su complejidad es alta.

En [12], se propone una arquitectura jerárquica de dos niveles. En el nivel alto, se utiliza SP (Strict Priorization) para clasificar el tráfico en cuatro colas dependiendo su tipo. En el nivel bajo, cada cola es calendarizada de acuerdo a un algoritmo en particular, por ejemplo, EDF (Early Deadline First) para rtPS y WFQ para nrtPS. Un enfoque similar es propuesto en [13], donde se reemplaza el algoritmo SP por DFPQ (Deficit Fair Priority Queue).

En [14], se deja en claro que el uso de la modulación adaptiva y la codificación tiene un impacto directo en la QoS. En dicho trabajo fue posible analizar cuantitativamente el impacto de diferentes tráficos y calidades de canal, usando Modulación y Codificación Adaptivas (AMC, Adaptive Modulation and Coding), en los parámetros de QoS. En [15] y [16], se presenta un análisis y diseño de capa cruzada para enlaces inalámbricos con un usuario y múltiples usuarios respectivamente. Dicho diseño optimiza la tasa de paquetes erróneos de AMC en la capa física, lo que provoca una disminución en la tasa de paquetes perdidos y una maximización del rendimiento promedio del encolamiento en la capa de enlace de datos. A diferencia del caso anterior, en [17], se analiza un diseño de capa cruzada en sistemas MIMO (Multiple Input Multiple Output). En este trabajo se investigan las características de encolamiento de dos categorías de sistemas MIMO. Los resultados presentan mejoras en la

tasa de paquetes perdidos y retardo de encolamiento.

Los modelos analíticos son también parte importante en la investigación de algoritmos de calendarización. En [11], se deriva un límite promedio de retardo usando un modelo de encolamiento M/G/FQ. En [10], se establece un análisis de encolamiento bastante profundo para medir el rendimiento en términos de tasa de paquetes descartados y *throughput*.

Por otra parte, en [18] se propone un algoritmo calendarizador denominado NewMDRR (New Modified Deficit Round Robin) para tráfico multi-clase, como el de Wimax, que es una mejora del MDRR (Modified Deficit Round Robin) usado por OPNET Modeler. Este trabajo ofrece también mejoras en el procedimiento de asignación de recursos reduciendo el *desperdicio* de 40 % a 20 %, lo que implica la posibilidad de dar servicio a 10 usuario más. En [19], se presenta otra modificación al mismo algoritmo tomando en consideración la SNR (Signal Noise Ratio). Sin embargo, este trabajo no presenta gráficas que sustenten la mejora en rendimiento que se anuncia.

Una técnica de calendarización puede ser implementada también en una SS, de modo que ésta pueda reasignar los recursos proporcionados por la BS de la forma más conveniente. No obstante, son pocos los trabajos que proponen un calendarizador para SS en una red Wimax; de hecho, OPNET Modeler no tiene implementado ningún algoritmo calendarizador para SS en su modelo de red Wimax. En [20], se presenta un sistema calendarizador basado en el modo GPSS (Grant Per Subscriber Station) del estándar IEEE 802.16. Dicho sistema abarca un calendarizador en la BS y otro en la SS. Esta solución proporciona servicios de tiempo real y servicios con requerimientos elevados de ancho de banda con retardos mínimos.

Descripción del Estándar IEEE 802.16e

Como se mencionó en capítulos anteriores, el Wimax Forum surgió con el objetivo de lograr la interoperabilidad entre equipos de distintos fabricantes. Este foro es una organización sin fines de lucro creada para promover y verificar la compatibilidad y la calidad de dispositivos de banda ancha inalámbricos basados en la familia de estándares 802.16.

En este capítulo se presenta un panorama general del estándar IEEE 802.16e, describiendo las características más relevantes de la capa Física y la capa MAC, que son necesarias en el análisis realizado en este trabajo.

3.1. Antecedentes

El 7 de diciembre de 2005, se aprobó el estándar IEEE 802.16e-2005, el cual ofrece una característica que no poseía su predecesor, el estándar IEEE 802.16-2004: la movilidad.

Las versiones fija y móvil del estándar 802.16 no son compatibles. La razón principal es que la versión móvil utiliza Scalable Orthogonal Frequency Division Multiple Access (SOFDMA) en los canales de subida y bajada, UL y DL respectivamente. SOFDMA incrementa el número de subportadoras OFDM escalando de 128 hasta 2048 subportadoras (con un esquema de 128, 256, 512, 1024 y 2048), dependiendo de la frecuencia de la señal recibida, los requerimientos del usuario y el ancho de banda del canal que se esté utilizando.

El estándar 802.16e contempla cuestiones de escalabilidad en tecnología de acceso inalámbrico y arquitectura de red; debido a esto, dicho estándar ofrece gran flexibilidad en el despliegue de redes y servicios ofrecidos. Algunas de las características sobresalientes son QoS, movilidad, seguridad, uso de técnicas MIMO y Adaptive Antena System (AAS) que le permiten lograr altas tasas de transmisión y gran cobertura. Ejemplo de ellos es que es posible soportar velocidades de hasta 63[Mbps] para el canal downlink y hasta 28[Mbps] en el uplink, esto en canales de 10[MHz].

El estándar 802.16 utiliza el modelo de referencia OSI, de hecho las especificaciones contenidas en el estándar se centran principalmente en las capas más bajas del dicho modelo. Particularmente, la capa de enlace de datos es dividida, en el estándar, en dos subcapas denominadas Logical Link Control(LLC) y MAC. La figura 3.1 muestra la estructura de capas consideradas en el estándar 802.16e y que son tomadas a partir de ahora como referencia.

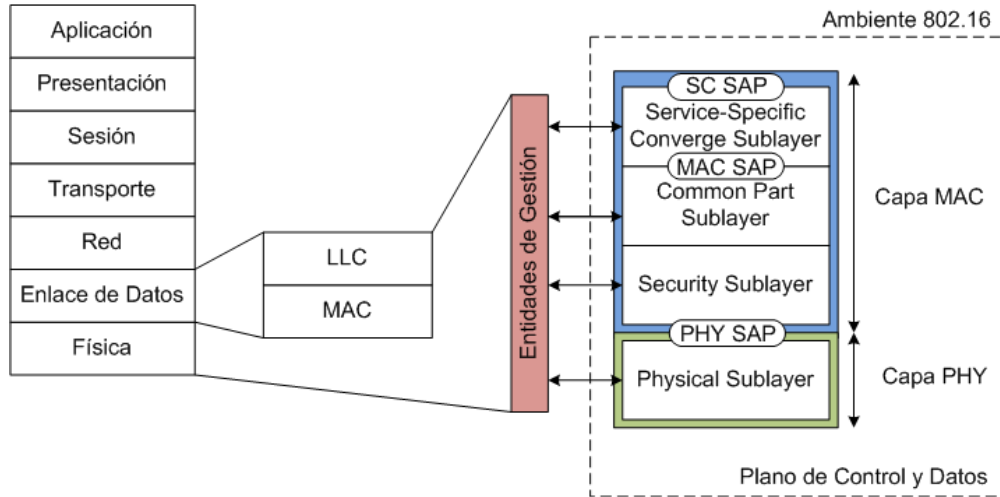


Figura 3.1: Modelo de referencia de IEEE 802.16e

3.2. Topologías IEEE 802.16

El estándar 802.16 define dos posibles topologías de red, cada una con especificaciones particulares.

- Topología PMP (Figura 3.2).
- Topología Mesh (Figura 3.3).



Figura 3.2: Topología PMP

La principal diferencia entre los modos mencionados es que en el modo PMP, el tráfico puede enviarse entre una BS y sus respectivas SSs; mientras que en el modo Mesh, el tráfico puede ser enrutado a través de otras SSs hasta la BS y puede existir comunicación entre SSs. PMP es una topología centralizada donde la BS es el centro del sistema, en la topología Mesh esto no ocurre. Los elementos en una red Mesh son llamados nodos, por ejemplo, una SS Mesh es un nodo.

En una topología Mesh, cada estación es capaz de crear su propia comunicación con cualquier otra estación en la red y, por lo tanto, la comunicación no está restringida a ser sólo con la BS. Por consiguiente, una mayor ventaja del modo Mesh es que el alcance de una BS puede ser mucho más grande dependiendo del número de saltos hasta la SS más distante.

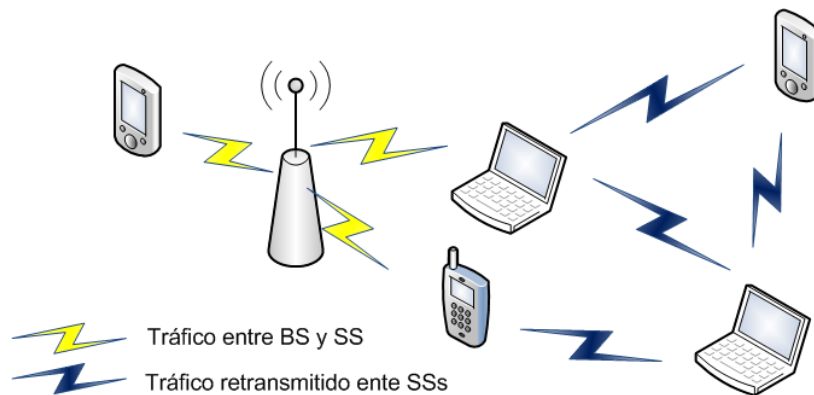


Figura 3.3: Topología Mesh

En un principio, el despliegue de redes Wimax estuvo planeado para seguir una topología principalmente PMP. Sin embargo, se ha reportado que algunos fabricantes están planeando incluir características Mesh en sus productos.

La topología utilizada durante el desarrollo de este trabajo es el modo PMP. Se utiliza entonces una BS y una o más SS que permiten el tráfico solamente con su respectiva BS.

3.3. Capa Física

Como ya se mencionó, Wimax es un sistema BWA, por lo tanto los datos son transmitidos a altas velocidades en la interfaz aérea a través de ondas electromagnéticas usando una determinada frecuencia.

La capa Física o PHY establece la conexión física entre dos dispositivos, frecuentemente en dos direcciones, uplink y downlink. En el estándar 802.16e la capa PHY es responsable de la transmisión de secuencia de bits. Define también el tipo de señal utilizada, el tipo de modulación y demodulación, la potencia de transmisión y otras características físicas.

La familia 802.16 considera la banda de frecuencia 2 a 66[GHz]. Esta banda se divide en dos partes:

- El primer rango abarca de 2 a 11[GHz] y se destina para transmisiones sin línea de vista (NLOS, Non Line of Sight).
- El segundo rango abarca de 11 a 66[GHz] y se destina para transmisiones con línea de vista (LOS, Line of Sight).

Designación	Banda de Frecuencia	Sección en el Estándar	Duplexión
WirelessMAN-SC	10 – 66[GHz] (LOS)	8.1	TDD y FDD
WirelessMAN-SCa	Menor a 11[GHz] (NLOS)	8.2	TDD y FDD
WirelessMAN-OFDM	Menor a 11[GHz] (NLOS)	8.3	TDD y FDD
WirelessMAN-OFDMA	Menor a 11[GHz] (NLOS)	8.4	TDD y FDD
WirelessHUMAN	Menor a 11[GHz] (NLOS)	8.5	TDD

Tabla 3.1: Las cinco interfaces físicas definidas en 802.16

Cinco interfaces físicas se definen en los estándares 802.16. Estas interfaces se muestran en la tabla 3.1. Para frecuencias de 10 a 66[GHz], se propone la interfaz WirelessMAN-SC PHY. Para frecuencias debajo de 11[Ghz], se proponen tres interfaces físicas.

- WirelessMAN-OFDM, mejor conocida como OFDM y usando transmisión Orthogonal Frequency Division Multiplexing (OFDM).
- WirelessMAN-OFDMA, mejor conocida como OFDMA y usando transmisión OFDM y Orthogonal Frequency Division Multiple Access (OFDMA).
- WirelessMAN-SCa, mejor conocida como SCa y utilizando modulaciones con una portadora.

3.3.1. Modulación Digital [21]

Como casi todas las comunicaciones recientes, Wimax/802.16 utiliza esquemas de modulación digital. Muchas técnicas de modulación están disponibles actualmente, cuatro modulaciones son soportadas en el estándar 802.16: BPSK, QPSK, 16-QAM y 64-QAM.

3.3.1.1. Binary Phase Shift Keying (BPSK)

La BPSK es una modulación digital binaria; esto proporciona alta inmunidad al ruido e interferencia. Una modulación digital con cambio de fase, como es el caso de BPSK, usa dos variaciones de fase para codificar los bits, es decir cada símbolo de modulación es equivalente a una fase. La fase de la señal modulada BPSK es π ó $-\pi$ de acuerdo al valor del bit de datos. Una forma común de ilustrar una modulación digital es la constelación. La figura 3.4 muestra la constelación de BPSK, los valores que la fase de la señal puede tomar son 0 ó $-\pi$.

3.3.1.2. Quadrature Phase Shift Keying (QPSK)

Cuando se necesita una mayor eficiencia espectral, mayor número de símbolos de modulación puede usarse. Por ejemplo, QPSK considera dos bits por símbolo. Los anterior hace

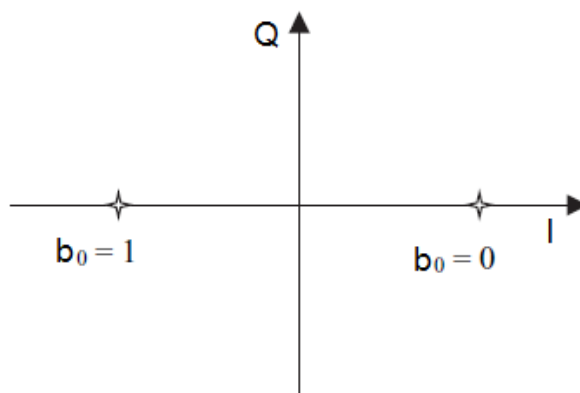


Figura 3.4: Constelación BPSK

evidente que la decisión en el receptor sea más complicada que una decisión para BPSK. Por lo tanto, la modulación QPSK es menos resistente al ruido que BPSK. Muchas variantes de QPSK pueden usarse pero siempre se tendrá una constelación de cuatro puntos, como la que se muestra en la figura 3.5.

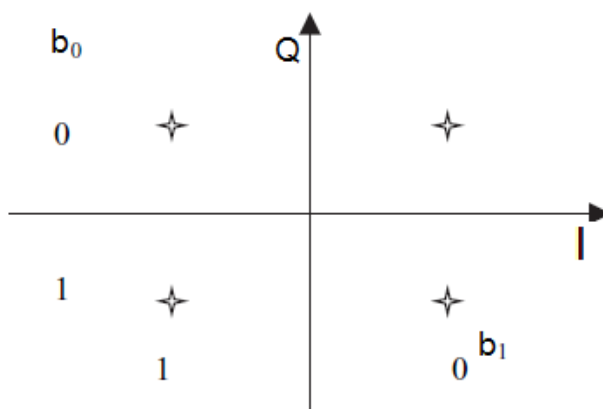


Figura 3.5: Constelación QPSK

3.3.1.3. Quadrature Amplitude Modulation (QAM): 16-QAM y 64-QAM

QAM cambia la amplitud de dos señales portadoras dependiendo de la secuencia digital a ser transmitida; las dos portadoras están fuera de la fase $+\pi/2$, este desfase es denominado cuadratura. 16-QAM (4 bits por símbolo) y 64-QAM (6 bits por símbolo) se incluyen en el estándar 802.16. 64-QAM se considera como la modulación con mayor eficiencia espectral de 802.16.

3.3.2. Transmisión OFDM

La técnica Orthogonal Frequency Division Multiplexing fue propuesta en 1966 por los Laboratorios Bell; sin embargo, hasta 1985 fue sugerida como técnica en la comunicaciones móviles. En 1999, el estándar IEEE 802.11g consideró OFDM para su capa física.

OFDM es una técnica de transmisión sumamente potente, está basada en el principio de transmisión simultánea de muchas frecuencias ortogonales de banda angosta. El número de subportadoras es frecuentemente denotado como N . Estas señales son ortogonales entre ellas, lo cual, en teoría, elimina la interferencia entre canales. Cada canal de frecuencia se modula con una modulación digital diferente. Las bandas de frecuencia asociadas con cada uno de estos canales menores al ancho de banda total que sería ocupado por una sola modulación, como es el caso de SC. Con OFDM, la duración en tiempo de un símbolo de datos es N veces más largo, proporcionando una mejor resistencia a la multitrayectoria. La figura 3.6 muestra lo anterior de forma gráfica.

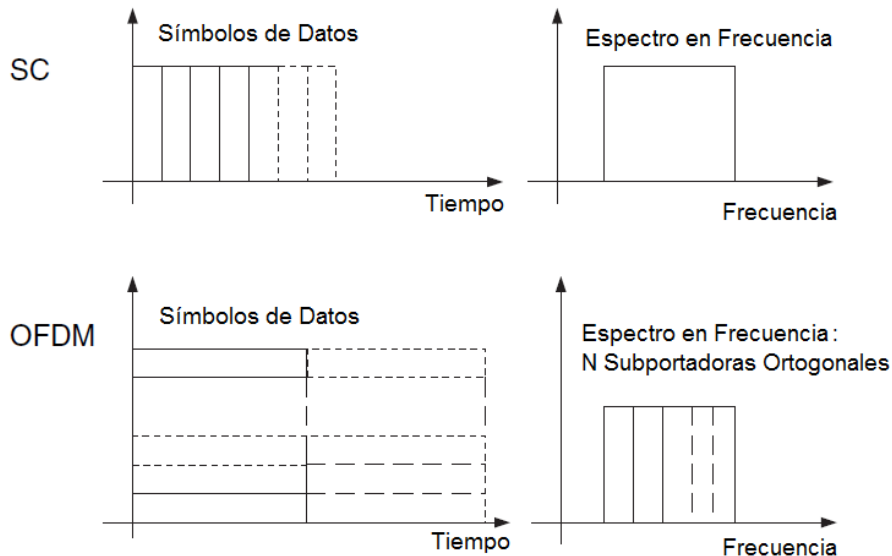


Figura 3.6: Representación en Tiempo y Frecuencia de SC y OFDM. En OFDM, N símbolos de datos son transmitidos simultáneamente en N subportadoras ortogonales

Tener un ancho de banda menor para cada canal equivale a periodos de tiempo más grandes y entonces mejor resistencia a los problemas por propagación multitrayectoria, con respecto a SC. Lo anterior y el hecho de que la portadoras sean ortogonales permiten una eficiencia espectral superior.

3.3.3. IFFT y OFDM

La FFT (Fast Fourier Transform) es el operador Transformada Rápida de Fourier. Este operador consta de una matriz que permite calcular la transformada de Fourier de una señal. La

FFT trabaja con cualquier número de puntos; sin embargo la operación se simplifica cuando se utiliza un número N que sea potencia de 2. La IFFT (Inverse Fast Fourier Transform) es el operador inverso de la FFT. La teoría de OFDM establece que la IFFT de magnitud N , aplicada en N símbolos, construye una señal OFDM, donde cada símbolo es transmitido en una de las N frecuencias ortogonales. El término símbolo se refiere a símbolos de datos de tipo BPSK, QPSK, 16-QAM y 64-QAM. La figura 3.7 muestra una ilustración simplificada del principio de generación de una señal OFDM.

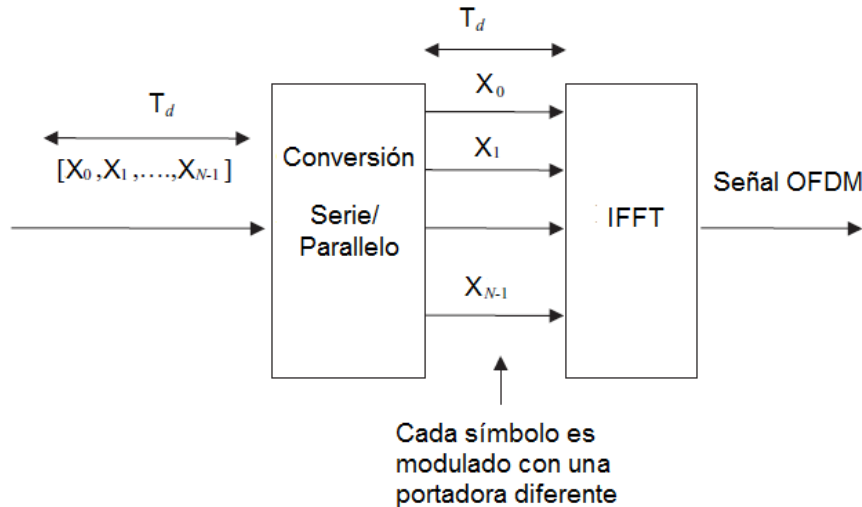


Figura 3.7: Generación de una señal OFDM

Si se considera T_d como la duración de un símbolo de datos de modulación transmitido, entonces $T_d = 1/\Delta f$, donde Δf es el ancho de banda de las frecuencias ortogonales. Como los símbolos de modulación son transmitidos al mismo tiempo, entonces se tiene que:

$T_d =$ duración de una símbolo OFDM = duración de un símbolo de datos de modulación transmitido.

Δf , la distancia en frecuencia entre dos subportadoras OFDM adyacentes, se presenta en la figura 3.8. Esta figura muestra como las subportadoras OFDM circunvecinas tienen valores iguales a cero en el valor máximo de una subportadora OFDM dada, es por eso que se consideran ortogonales.

3.3.4. Consideraciones de OFDM en el dominio del Tiempo [22]

Después de la aplicación de la IFFT, la teoría de OFDM requiere que un *Prefijo Cíclico* (CP, Cyclic Prefix) sea agregado al inicio del símbolo OFDM. CP permite al receptor absorber de forma más eficiente la dispersión del retardo debido a la multitrayectoria y para mantener las frecuencias ortogonales. El CP que ocupa una duración denominada *Tiempo de Guarda* (GT, Guard Time), a menudo denotada como T_G , es una redundancia temporal que debe

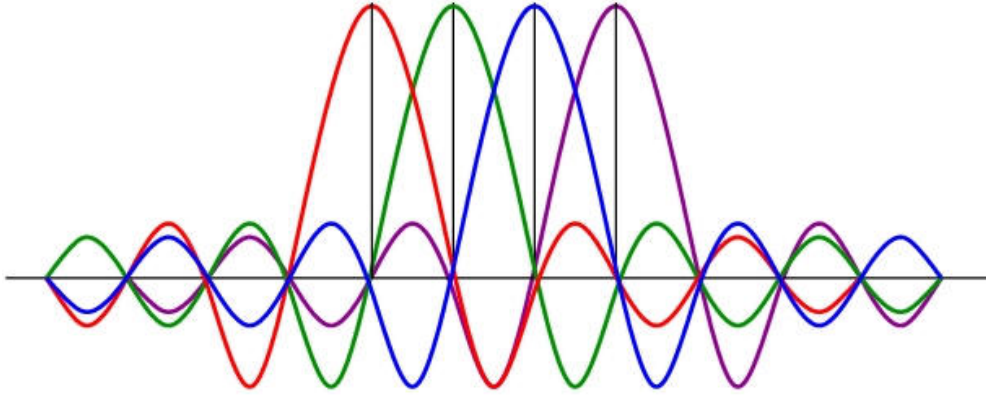


Figura 3.8: Presentación de las subportadoras OFDM en frecuencia

ser tomada en cuenta en el cálculo de tasa de datos (Figura 3.9). La relación T_G/T_d es generalmente denotado como G en el estándar 802.16. La selección de G se realiza de acuerdo a las siguientes consideraciones: si el efecto por multitrayectoria es significativo, se necesita un valor alto de G ; el cual aumenta la redundancia y, por tanto, decrece la tasa de datos útil. En cambio, si el efecto es ligero, un valor de G relativamente bajo es necesario. Para las capas físicas de OFDM y OFDMA, 802.16 define los siguientes valores de G : 1/4, 1/8, 1/16 y 1/32. En la versión móvil, sólo el valor 1/8 es obligatorio. El estándar indica que, para OFDM y OFDMA, una SS busca, en la inicialización, todos los posibles valores de CP hasta que encuentra el CP que la BS utiliza. Luego, la SS usa el mismo CP en el UL. Una vez que una duración de CP ha sido seleccionada por la BS para la operación en DL, ésta no pueda ser cambiada. Un cambio de CP implicaría una resincronización entre la SS y su BS.

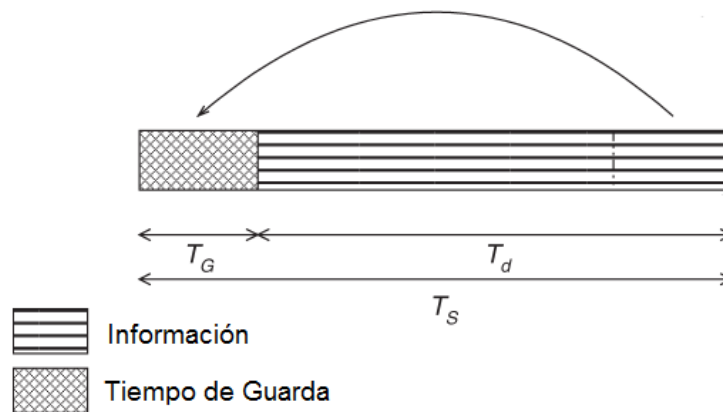


Figura 3.9: Inserción del Prefijo Cíclico en un símbolo OFDM

3.3.5. Consideraciones de OFDM en el dominio de la Frecuencia [22]

No todas las subportadoras de símbolos OFDM transportan datos. Existen en realidad 4 tipos de subportadoras. (Figura 3.10):

- Subportadoras de Datos: Transmisión de datos útiles.
- Subportadoras Pilotos: Principalmente para estimación de canal y sincronización.
- Subportadoras Nulas: Sin transmisión, son frecuencias de banda de guarda.
- Subportadora DC (Direct Current).

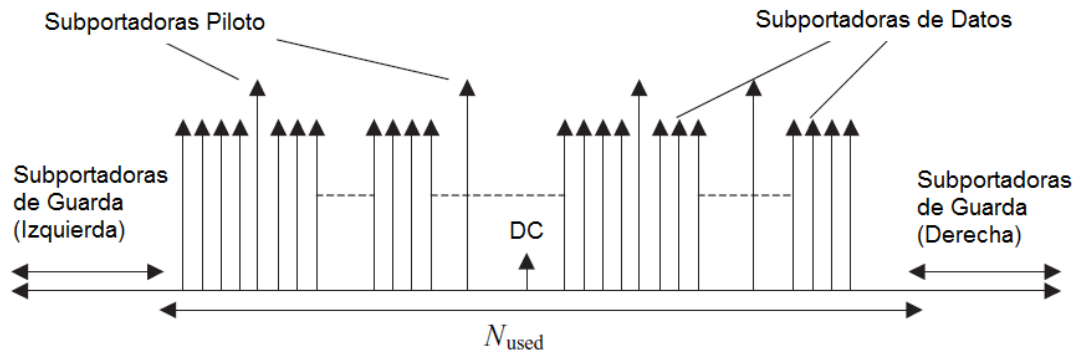


Figura 3.10: Tipos de subportadoras OFDM en 802.16

3.3.6. Parámetros del símbolo OFDM

Los parámetros principales del símbolo OFDM son los siguientes:

- El número total de subportadoras o su equivalente, la magnitud de IFFT, N_{FFT} .
- BW , el ancho de banda de canal.
- n , el factor sampling.

Para OFDM PHY, $N_{FFT} = 256$, de las cuales sólo 192 son para transmisión de datos, N_{USED} . En la versión móvil, el tamaño de la FFT es escalable de 128 hasta 2048. Cuando el ancho de banda disponible aumenta, el tamaño de FFT también lo hace, de tal forma que el espacio entre subportadoras es siempre 10,94[KHz]. Lo anterior conserva, en todos los casos, la duración del símbolo OFDM, el cual es la unidad de recurso básica. El mencionado espacio entre subportadoras fue seleccionado como un buen balance entre satisfacer la dispersión del retardo y los requerimientos de dispersión Doppler para la operación en ambientes fijos y móviles. Este espaciado es capaz de soportar valores de hasta 20[μs] en la dispersión

del retardo y movilidad de hasta 125[Km/h]. Un espacio inter-subportadoras de 10,94[KHz] implica que un FFT de tamaño 128, 512, 1,024 ó 2,048 esta en uso cuando el ancho de banda de canal es 1,25[MHz], 5[MHz], 10[MHz] y 20[MHz] respectivamente.

La *Frecuencia Sampling*, denotada como f_s , está relacionada con el ancho de banda ocupado mediante la siguiente formula:

$$f_s = \text{floor}\left(\frac{nBW}{8000}\right)8000 \quad (3.1)$$

En la ecuación 3.1, $\text{floor}()$ significa truncamiento, en este caso a un valor múltiplo de 8000. El valor numérico n depende del ancho de banda de canal. Los posibles valores son 8/7, 86/75, 144/125, 316/275 y 57/50 para OFDM PHY y 8/7 y 28/25 para OFDMA PHY.

3.3.6.1. Duración de símbolo OFDM

Basado en los parámetros descritos anteriormente, la duración de un símbolo OFDM puede ser calculado como:

$$T_s = T_b + T_g \quad (3.2)$$

donde:

$T_b = 1/\Delta f$, Tiempo útil de símbolo.

$T_g = GT_b$, Tiempo CP.

3.3.6.2. Valores de la tasa de datos

Debido a que la capa física de 802.16 es muy flexible, los valores de la tasa de datos varían dependiendo de los parámetros de operación. Los parámetros que tienen un impacto significativo en la tasa de datos a nivel capa física son el ancho de banda y el esquema de modulación y codificación. Otros parámetros, tales como el número de subportadoras, tiempo de guarda y frecuencia sampling también toman cierta importancia.

La tabla 3.2 muestra un ejemplo de tasa de datos para diferentes esquemas de modulación y codificación (MCS, Modulation y Coding Scheme) y valores de G , cuando se utiliza OFDM con 192 subportadoras de datos. El ancho de banda utilizado es 7MHz y el factor sampling es 8/7. Si se considera de la tabla, una modulación 16-QAM con tasa de codificación 3/4 y G de 1/16, se puede verificar que la tasa de datos es:

$$Data_{Rate} = \frac{bps_{OFDM}}{D_{S_{OFDM}}} = \frac{192(4)(3/4)}{\frac{256}{7 \times 10^6 (8/7)}(1 + 1/16)} = 16,94Mbps \quad (3.3)$$

Donde, bps_{OFDM} es el numero de bits no codificados por símbolo OFDM y $D_{S_{OFDM}}$ es la duración del símbolo OFDM [21].

G	BPSK 1/2	QPSK 1/2	QPSK 3/4	16-QAM 1/2	16-QAM 3/4	64-QAM 1/2	64-QAM 3/4
1/32	2.92	5.82	8.73	11.64	17.45	23.27	26.18
1/16	2.82	5.65	8.47	11.29	16.94	22.59	25.41
1/8	2.67	5.33	8.00	10.67	16.00	22.33	24.00
1/4	2.40	4.80	7.20	9.60	14.40	19.20	21.60

Tabla 3.2: Ejemplo de tasa de datos en Mbps para OFDM PHY

3.3.7. Slot físico

El slot físico (PS, Physical Slot) es una unidad básica de tiempo en el estándar 802.16. El PS corresponde a cuatro símbolos de modulación usados en el canal de transmisión. Para las capa físicas de OFDM y OFDMA, un PS está definido como:

$$PS = \frac{4}{f_s} \quad (3.4)$$

Por consiguiente, la duración PS está relacionada con la tasa de símbolos del sistema.

3.3.8. Estructura de Frame y Slot

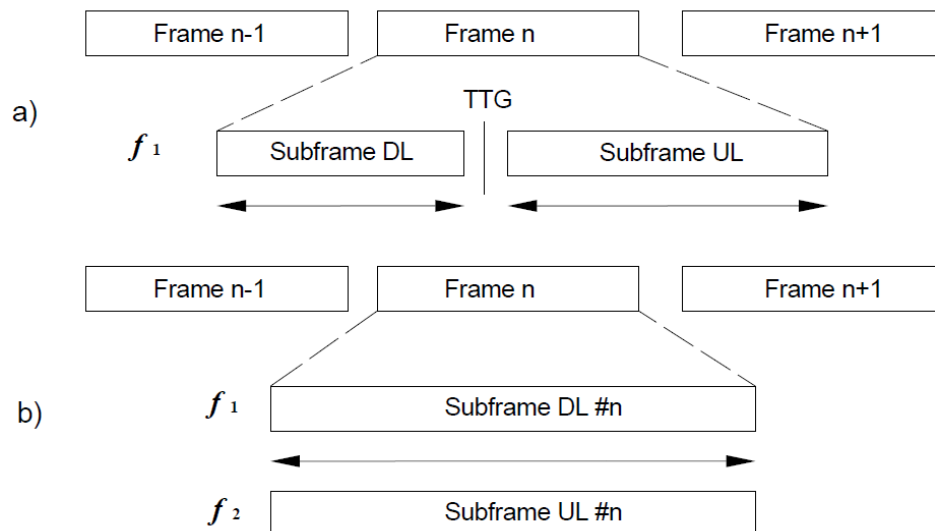


Figura 3.11: Representación de la estructura de un frame en modo TDD a) y FDD b)

El recurso mínimo de tiempo que se puede asignar a una conexión dada, para un sistema basado en el 802.16, se llama *slot* (ranura). La capa PHY es responsable de la asignación de slots y el envío de ellos. La figura 3.11 muestra un frame OFDM cuando opera en modo de

TDD y FDD. En el modo TDD, el frame se divide en dos subframes: un frame DL, después de un intervalo pequeño de guarda (TTG), y finaliza con un frame UL, figura 3.11a. La relación de transformación del frame de downlink a uplink puede variar de 3:1 a 1:1 para soportar diversos perfiles del tráfico. El estándar también soporta FDD, en cuyo caso la estructura del frame es la misma sólo que el downlink y uplink se transmiten simultáneamente, utilizando un canal para downlink y otro canal diferente para uplink, como se muestra en la figura 3.11b.

El frame downlink, figura 3.12, comienza con un preámbulo de downlink que es usado para los procedimientos de capa PHY, tales como sincronización de tiempo y frecuencia y valoración inicial del canal. El preámbulo de downlink es seguido por un encabezado de control de frame (FCH, Frame Control Header), que proporciona información sobre la configuración del frame, tal como la longitud de los mensajes de mapa de uplink y downlink, (DL-MAP y UL-MAP, mensajes que informan la estructura de cada frame), modulación, esquema de codificación, y los subportadoras utilizables. Después de la cabecera FCH, siguen los perfiles de las ráfagas para una o varias SSs transportando información solicitada por las SSs

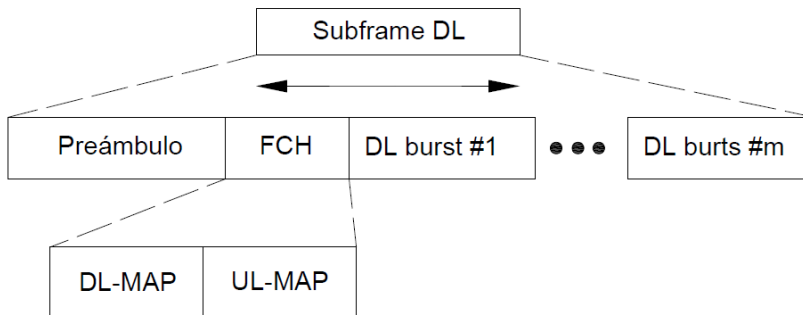


Figura 3.12: Representación de la estructura de un subframe DL

El estándar 802.16 es absolutamente flexible para multiplexar muchos usuarios y paquetes. Un sólo frame de downlink puede contener múltiples ráfagas de datos de varios tipos y tamaños para diferentes estaciones. El tamaño del frame es también variable desde $2[ms]$ hasta $20[ms]$, y cada ráfaga puede contener paquetes o fragmentos de tamaño fijo o variable concatenados. El frame de uplink, figura 3.13, está compuesto por tres regiones significativas. Una región de slots de contención para calibración y mantenimiento, utilizada para el proceso de registro e inicialización en la red. Después, le sigue una región de slots de contención, para que las SSs realicen sus peticiones de ancho de banda. Por último, existen una o varias regiones reservadas donde las SSs envían su información de datos.

3.3.9. Codificación y modulación adaptiva

El protocolo 802.16 utiliza una variedad de esquemas de modulación y codificación y permite que los esquemas cambien de conexión, dependiendo de las condiciones del canal. Para el uplink, la BS puede estimar la calidad del canal dependiendo de la calidad de la señal

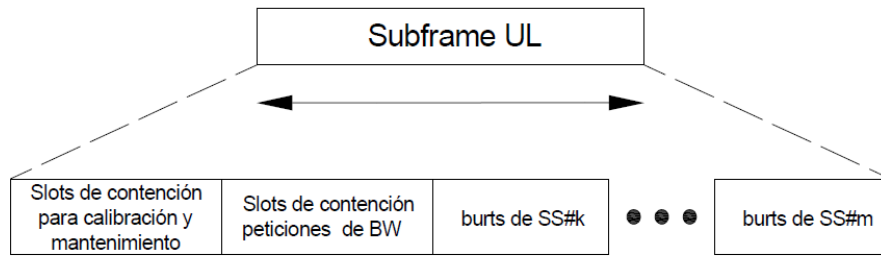


Figura 3.13: Representación de la estructura de un subframe UL

	DOWNLINK	UPLINK
Modulación	BPSK, QPSK, 16-QAM, 64-QAM. BPSK opcional para OFDMA PHY	BPSK, QPSK, 16-QAM. 64-QAM opcional
Codificación	Obligatorio: Códigos convolucionales de 1/2, 2/3, 3/4 y 5/6 Opcional: Turbo códigos convolucionales de 1/2, 2/3, 3/4 y 5/6; códigos de repetición de 1/2, 1/3 y 1/6; LDPC; Código RS para OFDM PHY	Obligatorio: Códigos convolucionales de 1/2, 2/3, 3/4 y 5/6 Opcional: Turbo códigos convolucionales de 1/2, 2/3, 3/4 y 5/6; códigos de repetición de 1/2, 1/3 y 1/6; LDPC

Tabla 3.3: Esquemas de Modulación y Codificación en UL y DL

recibida. El calendarizador de la BS puede analizar la calidad del uplink y downlink de cada SS y asignar un esquema de modulación y codificación, que maximice el rendimiento de procesamiento para la relación señal a ruido disponible. La modulación y codificación adaptiva aumenta considerablemente la capacidad total del sistema, debido a que permite la compensación, en tiempo real, entre el rendimiento de procesamiento y la robustez de cada conexión. La tabla 3.3 enlista varios esquemas de modulación y codificación para el protocolo 802.16. En el downlink, QPSK, 16 QAM, y 64 QAM son obligatorios para WiMAX fijo y móvil; 64 QAM son opcionales en el canal uplink. La codificación FEC (Forward Error Correction) que usa códigos convolucionales es obligatoria. Un total de 52 combinaciones de los esquemas de modulación y codificación se define en el estándar 802.16.

3.4. Capa MAC

En una red, el propósito de la capa PHY es la entrega confiablemente de los bits de información del transmisor al receptor, usando un medio físico como radiofrecuencia, ondas de luz o alambres de cobre. Generalmente, la capa PHY no tiene información de los requisitos de la QoS y no está enterada de la aplicación en uso, como VoIP, HTTP, o FTP (File Transfer Protocol). La capa PHY se puede ver como un tubo, responsable del intercambio de información sobre una sola conexión entre un transmisor y un receptor.

La capa MAC es responsable de controlar y multiplexar varias conexiones sobre el mismo medio físico. Esta capa es una conexión orientada e identifica una conexión lógica unidireccional entre la BS y la SS por un CID (Connection Identifier). Los CIDs para las conexiones de uplink y downlink son diferentes. El CID se puede ver como un direccionamiento temporal y dinámico, de capa 2, asignado por la BS para identificar una conexión unidireccional entre las capas MAC y PHY, y se utiliza para el control del tráfico de los datos. Para asociar ese direccionamiento, la subcapa de convergencia (CS) no debe perder de vista la asociación entre la dirección destino y el respectivo CID.

La capa MAC del estándar 802.16, que reside sobre la capa PHY, como se puede observar en la figura 3.1, se divide en tres subcapas:

- **Subcapa de Convergencia (CS).** Es la interfaz entre la capa MAC y la capa 3 de la red. recibe los paquetes de datos de las capas más altas, estos paquetes también se conocen como MAC Service Data Unit (SDU). La CS es responsable de realizar todas las operaciones que sean dependientes de la naturaleza del protocolo de la capa más alta, tal como la compresión de cabeceras y direccionamiento. La CS se puede ver como capa de adaptación que enmascara el protocolo de capas más altas.
- **Subcapa Común.** Realiza todas las operaciones del paquete que son independientes de capas más altas, tales como: fragmentación y concatenación de SDUs en MAC PDUs (Protocol Data Unit), programar la transmisión de MAC PDUs sobre los recursos disponibles de PHY, control de QoS y la retransmisión de los MAC PDUs que fueron recibidos erróneamente por el receptor cuando se utiliza ARQ (Automatic Repeat Request), un mecanismo de control de errores para la transmisión de datos.
- **Subcapa de seguridad.** Responsable de la cifrado, autorización y del intercambio apropiado de las claves de cifrado entre la BS y el SS.

3.4.1. MAC PDU: Construcción y transmisión

De acuerdo con el tamaño de la carga útil, múltiples SDUs pueden ser enviados en sólo un MAC PDU, o un SDU se puede fragmentar para ser transportado por múltiples MAC PDUs. Cuando un SDU se fragmenta, la posición de cada fragmento del SDU es etiquetada por un número de secuencia. El número de secuencia le permite a la capa MAC, en el receptor, ensamblar los fragmentos del SDU en el orden correcto. Para utilizar eficientemente los recursos de capa PHY, múltiples MAC PDUs son destinados al mismo receptor para ser concatenados y transportados en una sola oportunidad de transmisión o región de datos, según lo mostrado en la figura 3.14.

Las regiones de datos del uplink y downlink de una SS son un conjunto de slots adyacentes reservados para sus oportunidades de transmisión. Para las conexiones que no usan ARQ, cada fragmento del SDU se transmite en secuencia. Para las conexiones que usan ARQ, el SDU primero se fragmenta en bloques ARQ de longitud fija, y se les asigna un número secuencial de bloque (BSN) de ARQ. La longitud de los bloques de ARQ es especificada por la BS para cada CID, usando el parámetro ARQ-BLOCK-SIZE. Si la longitud del SDU no

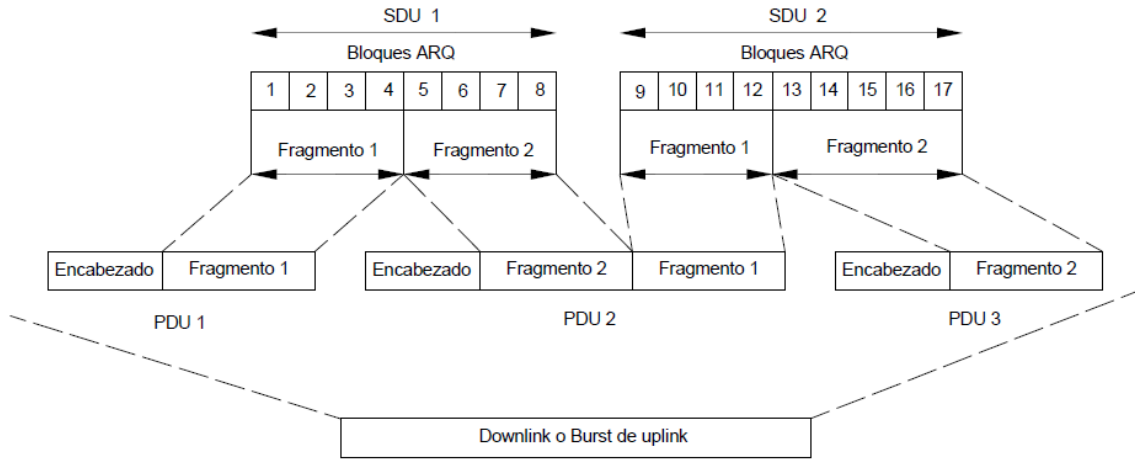


Figura 3.14: Fragmentación y concatenación de SDUs en MAC PDUs

es un múltiplo de ARQ-BLOCK-SIZE, el último bloque ARQ se rellena. Después de que se fragmenta en bloques ARQ, el SDU se ensambla en MAC PDUs como se muestra en la figura 3.14.

Para las conexiones que tienen habilitado el ARQ, el encabezado de fragmentación y empaquetado contiene el BSN del primer bloque de ARQ y del siguiente. La retroalimentación ARQ del receptor viene en forma de ACK (Acknowledgment), indicando la recepción apropiada de los bloques de ARQ. Esta retroalimentación se envía como un MAC PDU independiente o como piggyback en la carga útil de una MAC PDU normal. En el estándar 802.16, la retroalimentación de ARQ puede estar en la forma de ACK selectivo o de ACK acumulativo. Un ACK selectivo, para un BSN determinado, indica que el bloque de ARQ se ha recibido sin errores. Por otra parte, un ACK acumulativo, para un BSN dado, indica que todos los bloques con números o iguales al BSN se han recibido sin error.

El MAC PDU posee tres campos como se puede observar en la figura 3.15. El primer campo es de tamaño fijo y corresponde al encabezado MAC genérico, el segundo campo es opcional, de longitud variable y corresponde a la carga útil que puede tener subencabezados. El tercer campo corresponde al CRC (Cyclic Redundancy Check), es también opcional y protege tanto al encabezado como a la carga útil.

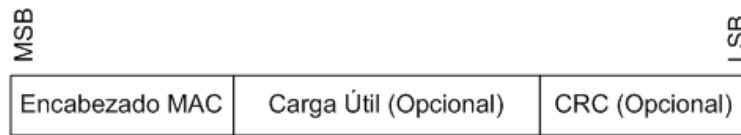


Figura 3.15: Estructura de MAC PDU

El estándar 802.16 define dos tipos de encabezados de MAC PDUs, cada uno con una estructura diferente, como se muestra en la figura 3.16.

La MAC PDU genérica, figura 3.16a, se utiliza para transportar los datos y mensajes de señalización de capa MAC. Los elementos de información en la cabecera de una MAC PDU

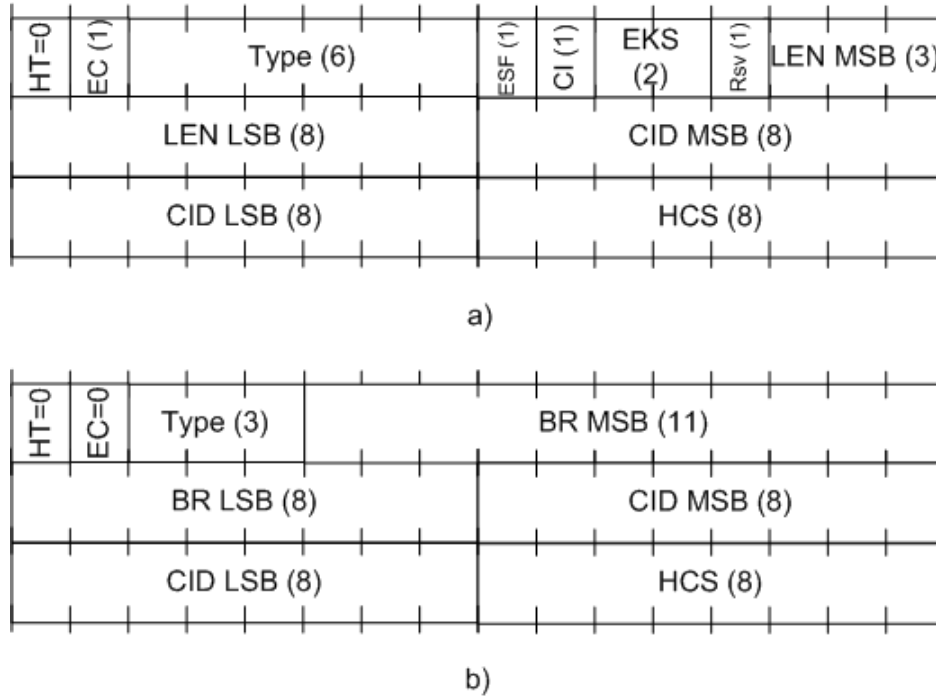


Figura 3.16: Encabezados de PDU a) genérico; b) petición de ancho de banda

genérica se muestran en la tabla 3.4

La MAC PDU de petición de ancho de banda, figura 3.16b, es utilizada por las SSs para indicar a la BS cuánto ancho de banda requieren en el uplink para el envío de datos. Una MAC PDU de petición de ancho de banda consiste solamente en una cabecera de petición de ancho de banda, sin carga útil o el CRC. Los elementos de información de una cabecera de la petición de la anchura de banda se proporcionan en la tabla 3.5.

Además de los encabezados genéricos y de petición de ancho de banda, el estándar 802.16 también define cinco subencabezados que se puedan utilizar en un MAC PDU genérico:

- **Subencabezado Mesh.** Cuando se utiliza una red de tipo Mesh.
- **Subencabezado de fragmentación.** Indica que el SDU está fragmentado en múltiples MAC PDUs.
- **Subencabezado Packing.** Este encabezado de empaquetamiento indica que múltiples SDUs o fragmentos de SDU están empaquetados en una sola MAC PDU y están colocados al principio de cada SDU o fragmentos de SDU.
- **Subencabezado de asignación de retroalimentación rápida.** Este encabezado indica que la PDU contiene retroalimentación del SS sobre la información del estado del canal DL. Proporciona la funcionalidad de retroalimentación de información del estado del canal para sistemas MIMO y non-MIMO.
- **Subencabezado Grant-management.** El encabezado de administración de concesión, es utilizado por la SS y transporta los mensajes relacionados con el manejo del ancho

Campo	Longitud (bits)	Descripción
HT	1	Tipo de encabezado. Se fija en 0 para este encabezado.
EC	1	Control de cifrado, 0: No cifrado 1: Cifrado.
Type	6	Tipo.
ESF	1	Campo de encabezado extendido, 1: Está presente.
CI	1	Indicador de CRC, 1: Si incluye CRC.
EKS	2	Secuencia de la clave de cifrado.
Rsv	1	Reservado.
LEN	11	Logitud en byte del MAC PDU, incluyendo encabezado.
CID	16	Identificador de conexión.
HCS	8	Secuencia de verificación de encabezado.

Tabla 3.4: Campos de la cabecera MAC genérica

Campo	Longitud (bits)	Descripción
HT	1	Tipo de encabezado. Se fija en 1 para este encabezado.
EC	1	Control de cifrado. Se fija en 0 para este encabezado.
Type	3	Tipo.
BR	19	Petición de ancho de banda. Ancho en banda en bytes.
CID	16	Identificador de conexión.
HCS	8	Secuencia de verificación de encabezado.

Tabla 3.5: Campos de la cabecera MAC para peticiones de ancho de banda

de banda, tal como peticiones por consulta y peticiones de ancho de banda adicional. Usar este encabezado es más eficiente que la petición de ancho de banda inicial, es más compacta y no requiere la transmisión de una PDU nueva.

Una vez que se construye un MAC PDU, se entrega al calendarizador, que programa el MAC PDU sobre los recursos disponibles de capa PHY. El calendarizador controla el ID del flujo de servicio y el CID de la MAC PDU, que permite que calibre sus requisitos de QoS. De acuerdo con los requisitos de QoS del MAC PDUs que pertenece a diferentes CIDs y IDs de flujo de servicio, el calendarizador determina la asignación de recursos óptimos de capa PHY para todos los MAC PDUs, sobre los frames.

3.4.2. Calidad de Servicio (QoS)

Una de las funciones dominantes de la capa MAC es asegurarse de que los requisitos de QoS, para MAC PDUs, estén asignados dadas las condiciones de la carga del sistema. Esto implica que el tiempo de espera, la tasa de datos, la tasa de error del paquete y disponibilidad de sistema, deben ser óptimos para cada conexión. Se tienen tres mecanismos para la caracterización de QoS: solicitudes anidadas (Piggyback), robo de Ancho de Banda (Bandwidth Stealing) y consulta (Polling).

3.4.2.1. Piggyback Request

Se trata de solicitudes de asignación de ancho de banda incluidas en los encabezados de los paquetes enviados por las SSs durante una transmisión garantizada. Es decir, cuando la SS ha hecho una solicitud previa en el periodo de contención y la BS le ha proporcionado una cantidad específica de slots de transmisión, es posible que la SS aún tenga paquetes por enviar que no habían sido considerados cuando se realizó la asignación. Para evitar hacer una solicitud por contención nuevamente, la SS incluye en los encabezados de los paquetes que está enviando una solicitud adicional. Esta nueva solicitud tiene carácter preferencial, ya que su recepción en la BS está garantizada, aunque es posible que no se realice la asignación en su totalidad.

3.4.2.2. Bandwidth Stealing

Se refiere al empleo de una porción del ancho de banda asignado a una SS para solicitudes adicionales. Este proceso tiene lugar cuando una SS se encuentra operando bajo un esquema de otorgamiento por SS, esto es, que cuenta con oportunidades garantizadas para la transmisión de paquetes. En este caso, es posible que parte de estas oportunidades, destinadas para el envío de datos, se use para realizar solicitudes adicionales de ancho de banda. El término stealing se refiere a que parte de los slots destinados para enviar su información se usa para hacer solicitudes de ancho de banda.

Calidad de Servicio	Piggyback Request	Bandwidth Stealing	Polling
UGS (Unsolicited Grant Service)	No válida	No válida	Sólo aplica cuando se solicita mediante el bit PM para transmisiones que no sean UGS
rtPS (Real Time Polling Service)	Permitida	Permitida para Otorgamientos por SS (GPSS)	Sólo se permite la consulta unicast.
nrtPS (Non Real Real Polling Service)	Permitida	Permitida para GPSS	Puede restringirse hacer consulta unicast. Otros tipos de consulta están permitidas.
BE (Best Effort)	Permitida	Permitida para GPSS	Todas las formas están permitidas.

Tabla 3.6: Tipos de Servicio para transmisiones en el UL

3.4.2.3. Polling

Es el proceso mediante el cual la BS destina a una SS o a una conexión, oportunidades específicas para el envío de solicitudes de ancho de banda. En este caso, la BS asigna regularmente algunos slots que la SS puede usar para hacer solicitudes de ancho de banda. Dichas oportunidades son independientes de aquellas que se asignan normalmente para las transmisiones de las SSs. Adicionalmente, la SS puede indicar que requiere de este servicio.

Las funciones anteriormente mencionadas permiten que el estándar 802.16 soporte la clasificación de cuatro tipos de QoS para las transmisiones en el uplink, como se describe en la tabla 3.6.

3.4.2.4. Unsolicited Grant Service (UGS)

Soporta servicios de tiempo real que generen paquetes de longitud fija en forma periódica, por ejemplo: T1/E1 y VoIP sin supresión de silencios. La asignación de oportunidades de transmisión periódica y fija en tiempo real, permite reducir la latencia, sin dejar de cumplir con las demandas de la comunicación en tiempo real. La política de solicitud y transmisión impide a la SS utilizar las oportunidades de contención para la asignación de ancho de banda. Adicionalmente, la BS no podrá proporcionar a la conexión que usa este servicio la oportunidad unicast para realizar solicitudes. Por este motivo, la SS sólo podrá transmitir tráfico constante en oportunidades no solicitadas para esa conexión en particular.

3.4.2.5. Real Time Polling Service (rtPS)

Soporta servicios de tiempo real que generan paquetes de longitud variable, en forma periódica, por ejemplo: video MPEG. Mediante este servicio se ofrecen oportunidades periódicas unicast que permiten a la SS especificar la longitud deseada para cada una de las asignaciones. Puede mejorarse la eficiencia en la utilización del canal inalámbrico y la periodicidad de las asignaciones de oportunidades de transmisión debe cubrir los requerimientos del tráfico generado por la SS. Cabe mencionar que también es posible que la SS emplee el UGS, sin embargo, no está permitido que emplee la región de contención para el envío de solicitudes de asignación de ancho de banda para esa conexión en particular.

3.4.2.6. Non Real Time Polling Service (nrtPS)

Soporta servicios que no son de tiempo real y que generan paquetes de longitud variable a intervalos regulares de tiempo, tales como FTP. La BS deberá proporcionar periódicamente, durante intervalos menores a un segundo, oportunidades de transmisión para la SS, aún en el caso de congestión en la red. De acuerdo con la política de solicitud y transmisión, en este servicio la SS puede emplear los intervalos de contención para hacer solicitudes adicionales de ancho de banda, así como oportunidades unicast y del tipo UGS.

3.4.2.7. Best Effort (BE)

No reserva ancho de banda, es decir, no asegura tasas de transmisión. Asigna tasas de transmisión disponibles y no da prioridad. Este tipo de servicio es característico del tráfico intermitente de IP, donde la SS puede emplear cualquier tipo de oportunidad para realizar solicitudes de asignación de ancho de banda, como son las regiones de contención y las oportunidades unicast.

Todos los tipos de servicio requerirán de aumentos o disminuciones de forma dinámica del ancho de banda asignado, exceptuando el UGS el cual tiene una tasa fija de transmisión asignada. Por esto, la BS debe permitir a las SSs que informen sus requerimientos de ancho de banda.

3.4.3. Solicitudes y asignación de ancho de banda

Con la finalidad de tener acceso al canal inalámbrico, cada una de las SSs participa en un mecanismo de solicitud de ancho de banda a través del cual se indica a la BS la necesidad de incrementar o disminuir los requerimientos de ancho de banda. Los mecanismos mediante los cuales las SS solicitan a la BS la asignación de ancho de banda de acuerdo con los requerimientos específicos de cada conexión se conoce como Acceso Múltiple por Asignación de Demanda (DAMA, Demand Assigned Multiple Access). Cada una de las SSs envía mensajes para solicitar la reservación de slots en el uplink. Las oportunidades de transmisión requeridas dependen del tipo de tráfico generado así como los parámetros de QoS especificados durante el establecimiento inicial de la conexión. Existen diversos métodos

empleados por las SSs para realizar sus peticiones: Solicitudes, Reservaciones por Conexión, Reservaciones por Estación Suscriptora y Consulta.

3.4.3.1. Solicitudes de ancho de banda

Las solicitudes consisten en mensajes enviados por las SSs hacia la BS para solicitar la reservación de ancho de banda en el uplink para la transmisión de datos. Debido a las características dinámicas del canal inalámbrico, las solicitudes se expresan en términos de bytes sin tomar en cuenta el overhead de la capa física. Estas solicitudes pueden transmitirse durante el periodo de contención de solicitud de ancho de banda, o bien aprovechando algún periodo asignado por la BS para la conexión.

Las solicitudes realizadas durante una región reservada se realizan mediante los encabezados de solicitudes piggyback, que permiten optimizar el uso del canal inalámbrico, ya que reducen la posibilidad de que ocurran colisiones durante el envío múltiple de solicitudes de asignación de ancho de banda por varias estaciones.

Cabe señalar que las solicitudes pueden ser *Incrementales* o *Agregadas*, lo cual se indica en el campo Type del encabezado del mensaje de solicitud de ancho de banda. Dado que los encabezados para las peticiones piggyback no cuentan con el campo Type, se ha definido que siempre serán del tipo Incrementales. La BS cuenta con un registro de las peticiones realizadas por cada SS. En el caso de las solicitudes Incrementales, dicho registro del ancho de banda solicitado sólo se modifica de acuerdo con los mensajes enviados, esto es, cada SS indica que para la cantidad de bytes requeridos con anterioridad es necesario agregar otra cantidad, como se indique la solicitud Incremental.

Por otro lado, en las solicitudes *Agregadas*, la BS renueva su registro del ancho de banda solicitado por la SS, de acuerdo con la última petición enviada, sin importar que del ancho de banda requerido anteriormente sólo se haya reservado una porción. Debido a las características dinámicas del canal, la BS puede requerir que se generen solicitudes *Agregadas* bajo un esquema periódico, lo cual dependerá de los parámetros de QoS para un servicio en particular, así como de las características del canal. Es importante señalar que las solicitudes realizadas durante las regiones de contención siempre deberán ser *Agregadas*.

3.4.3.2. Modo de Asignación por Conexión (GPC)

En las *Asignaciones por Conexión* (GPC, Grant per Connection) el ancho de banda es asignado explícitamente para una conexión y las SSs usan el ancho de la asignación sólo para esa conexión.

3.4.3.3. Modo de Asignación por Estación Suscriptora (GPSS)

Con el modo de *Asignación por Estación Suscriptora* (GPSS, Grant per SS), a las SSs se les otorga ancho de banda adicional dentro de la misma asignación. En este modo, las SS deberán ser más inteligentes en su manejo de calidad de servicio. Típicamente usará el ancho

de banda para la conexión que la solicitó, pero no necesariamente. Además si la calidad de servicio de la SS ha cambiado desde la última petición, la SS tiene la opción de mandar los datos con mayor calidad de servicio.

3.4.3.4. Consultas (Polling)

Mediante las consultas, la BS establece periodos para la transmisión en el uplink destinados específicamente para que las SSs envíen solicitudes de asignación de ancho de banda. Ya que la BS no envía mensajes explícitos para indicar tales oportunidades, dichos periodos se definen mediante los correspondientes elementos de información dentro de los UL-MAPs.

Las asignaciones de estas oportunidades pueden realizarse para una sola SS (Unicast) o para un grupo de SSs o de conexiones (Multicast). En este último caso, las estaciones o conexiones involucradas tendrán que contender para conseguir el envío de sus solicitudes. En base al tipo de consulta, la asignación podrá ser del tipo GPC o GPSS.

A continuación se mencionan las características para Consultas de tipo Unicast y de tipo Multicast.

- **Consulta Unicast:** Cuando se establece una Consulta Unicast para una SS en particular, la BS incluye en el UL-MAP una oportunidad exclusiva para la SS que podrá ser empleada sólo para solicitar ancho de banda.
- **Consulta Multicast:** Cuando la BS no dispone de ancho de banda suficiente para atender a cada una de las SS asociadas a ella, puede emplear consultas a un grupo de SSs. Tales grupos se identifican mediante CIDs exclusivos y las oportunidades que reciben se especifican a través del UL-MAP.

Las SSs involucradas sólo envían solicitudes si lo necesitan, participando en contenciones con las otras SSs de su grupo. Debido a que se encuentran sujetas a la ocurrencia de colisiones, emplean el algoritmo exponencial backoff tanto para determinar el slot en el que enviarán su solicitud para reprogramarla, en el caso de haberse presentado una colisión. Las solicitudes se considerarán exitosas sólo si en algún futuro UL-MAP se incluye una reservación para la CID en cuestión.

Análisis de la Asignación de Recursos en una Red Wimax

Para poder diseñar una técnica de calendarización de una red Wimax móvil, primero es necesario conocer a fondo la forma en que los recursos de dicha red son asignados. A continuación, se presentan la descripción y el análisis detallados de las diferentes secuencias de operaciones realizadas normalmente en un red Wimax móvil durante la asignación de recursos a cada SS.

Las descripciones de los eventos, acciones y operaciones aquí analizados están apegados a la herramienta de programación que se utilizó para la simulación de los diferentes escenarios de este trabajo: Optimized Network Engineering Tool (OPNET) Modeler Versión 16.6. OPNET Modeler es comúnmente utilizado como una de las principales herramientas de programación y simulación de modelos de red, ya que está orientado hacia el modelado de redes de comunicaciones de datos. Asimismo, soporta una gran cantidad de protocolos de comunicaciones, entre ellos el protocolo IEEE 802.16. También contiene un extenso conjunto de características diseñadas para soportar el modelado de redes y proporciona una gran flexibilidad para desarrollar detalladamente modelos de red personalizados.

El presente capítulo proporciona un panorama general de OPNET Modeler y su diseño jerárquico. Asimismo se describe el modelo de simulación genérico de una red Wimax móvil. Posteriormente, se describen de manera general los módulos y procesos relevantes que integran cada uno de los diferentes tipos de nodos que conforman una red, incluyendo los procesos denominados como hijos.

4.1. OPNET Modeler

OPNET Modeler fue creado como herramienta en el diseño y prueba de protocolos de comunicaciones y para simular el desempeño de las redes de datos. Éste proporciona una interfaz gráfica jerárquica para la definición de sistemas de comunicaciones y protocolos. En el nivel jerárquico más alto se encuentra la red, en el siguiente nivel se encuentran los nodos y finalmente los procesos. La red se construye gráficamente conectando los nodos a través de enlaces de comunicación. Los nodos se construyen definiendo modelos de colas y

procesos conectados entre sí. Una cola o proceso típicamente representa una capa en la pila de protocolos, pero es lo suficientemente flexible para ser definido(a) por el usuario.

Los procesos y colas se construyen especificando máquinas de estados finitos (FSM) con transiciones condicionales o automáticas. El comportamiento de los estados finitos está definido por eventos y por un código en proto-C, el cual puede ser modificado y/o especificado por el usuario.

Con el propósito de tener una perspectiva de la forma en la que los modelos de simulación son implementados, a continuación se presenta una visión general del diseño jerárquico utilizado por OPNET. Dichos modelos están basados en una estructura gráfica jerárquica de tres niveles, como se muestra en la figura 4.1.

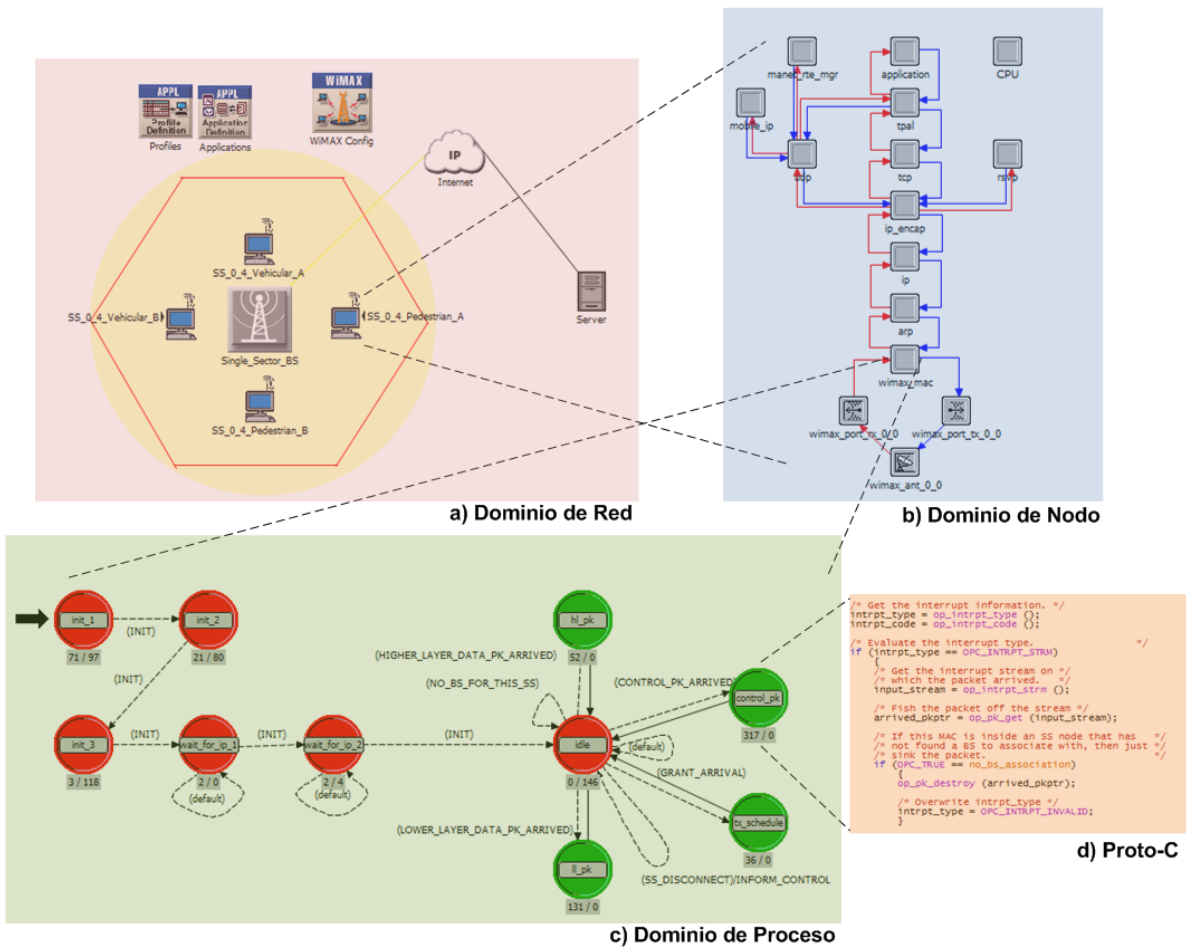


Figura 4.1: Estructura jerárquica del modelo de red en OPNET Modeler

- Dominio de Red.** El nivel más alto en la estructura jerárquica es el dominio de red, como se muestra en la figura 4.1a. El principal objetivo de éste nivel es definir la topología de la red de comunicaciones. Las entidades de la red son llamadas nodos y se interconectan entre sí utilizando enlaces. La estructura interna de los nodos y enlaces no está visible en el nivel de red. El modelo de red define el alcance general

del sistema a ser simulado. También especifica los objetos en el sistema, así como su ubicación física, interconexiones y configuraciones. Un modelo de red puede contener un nodo, una subred o muchos nodos y subredes interconectados, ya que la estructura y complejidad de un modelo de red, típicamente sigue a los sistemas reales a ser simulados.

- **Dominio de Nodo.** El siguiente nivel es el dominio de nodo, como se muestra en la figura 4.1b. Éste nivel define la funcionalidad de cada dispositivo de comunicación que puede ser desplegado e interconectado en la red, como por ejemplo routers, bridges, terminales, switches, etc. La estructura de estos objetos no es visible desde el nivel de red, por tal motivo, el dominio de nodo especifica la estructura interna y muchas de las capacidades de los nodos de comunicación. Un modelo de nodo está compuesto de una serie de bloques interconectados denominados módulos. Cada módulo contiene un conjunto de entradas y salidas, algunos estados de memoria y un método para procesar dichas entradas y salidas. Cada nodo consiste de diferentes tipos de módulos, como generadores de tráfico (aplicaciones), procesos (pilas de protocolos), colas (buffers) y varios transmisores y receptores que permiten al nodo establecer enlaces de comunicación en la red. El modelo del nodo especifica la manera en la que las entradas y salidas de los módulos están interconectados, empleando objetos llamados conexiones.

- **Dominio de Proceso.** El dominio de proceso se muestra en la figura 4.1c. En este nivel se especifica el comportamiento de un procesador o de una cola que residen en el dominio de Nodo. Los modelos de proceso pueden ser usados para implementar una amplia variedad de subsistemas de hardware y software, incluyendo protocolos de comunicaciones, algoritmos, generadores de tráfico, etc. En este nivel la funcionalidad de cada cola y proceso es definida usando una FSM. Los estados y transiciones definen gráficamente la progresión de un proceso o cola en respuesta a eventos. En general, cada FSM puede definir variables de estado privadas y puede hacer llamadas a librerías de código incluidas en OPNET o proporcionadas por el usuario. Una FSM tiene un comportamiento dinámico y puede ser invocada por otra FSM durante una simulación y en respuesta a eventos específicos. La dinámica de la FSM simplifica en gran medida la especificación de los protocolos, los cuales manejan un número escalable de recursos y sesiones. Finalmente, cada estado de un modelo de proceso o cola contiene código proto-C, el cual contiene una extensa librería de funciones diseñadas para la programación de protocolos.

- **Código proto-C.** Se trata de un lenguaje de programación que fue diseñado específicamente para soportar desarrollo de protocolos y algoritmos. Proto-C está basado en una combinación de diagramas de transición de estados (STD), una biblioteca de comandos de alto nivel conocida como Kernel Procedures y las facilidades generales de los lenguajes de programación C y C++. Este código puede ser modificado por el usuario e incluye llamadas para establecer y manejar colas, retardos de proceso, manejo de eventos, generación aleatoria de números basada en una variedad de distribuciones de probabilidad, construcción y encapsulamiento de paquetes y recolección estadística de datos.

4.2. Elementos de una red Wimax móvil en OPNET

OPNET Modeler posee prácticamente todas las herramientas para la simulación de una red Wimax Móvil, dichas herramientas, que incluyen modelos de nodos y procesos, se apegan completamente al estándar IEEE 802.16. A continuación se describen de manera general cada uno de los modelos que integran el modelo de red utilizado en este trabajo.

4.2.1. Estación Suscriptora (SS)

La SS es el modelo de nodo que representa al equipo transceptor del usuario. Está compuesta por varios procesos encargados de realizar las principales funciones del modelo de referencia TCP/IP, como se muestra en la figura 4.2. Estos procesos incluyen la capa de aplicación, las capas TCP y UDP (Transporte), la capa IP (Internet) y la capa WiMAX MAC (Acceso al Medio). Este modelo cuenta, también, con procesos específicos que le permiten soportar movilidad, y otros módulos para el envío y recepción al y desde el canal inalámbrico. De la misma manera, se incluye un módulo que realiza las funciones de una antena WiMAX.

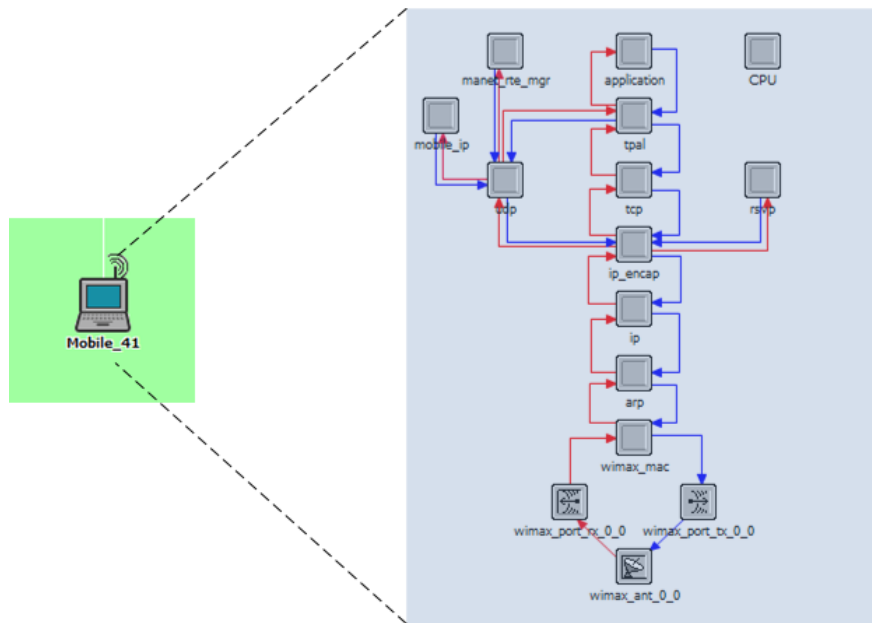


Figura 4.2: La Estación Suscriptora en el dominio de red y de nodo

4.2.1.1. Procesos relevantes de la SS

El modelo de nodo de la SS está conformado por módulos de diferentes tipos, entre los que se encuentran procesos, módulos de transmisión/recepción y un módulo de antena. A continuación se describen los procesos de la SS involucrados en la asignación de recursos, es decir, el algoritmo de calendarización.

4.2.1.1.1. Proceso *wimax_mac*

Este proceso representa la capa MAC del estándar IEEE 802.16 e incluye la implementación de las principales características y mecanismos que define dicho estándar. A este proceso se le denomina proceso *raíz* o proceso *padre* y es responsable del control del plano de datos de la capa WIMAX MAC (figura 4.3). Entre sus funciones se encuentran las siguientes:

- Reenvío de Datos
- Clasificación del tráfico de capas superiores en los diferentes flujos de servicios
- Encapsulamiento/Desencapsulamiento de paquetes de capa superiores en frames de capa MAC
- Mecanismo de solicitud/concesión de ancho de banda
- Entrega de Paquetes

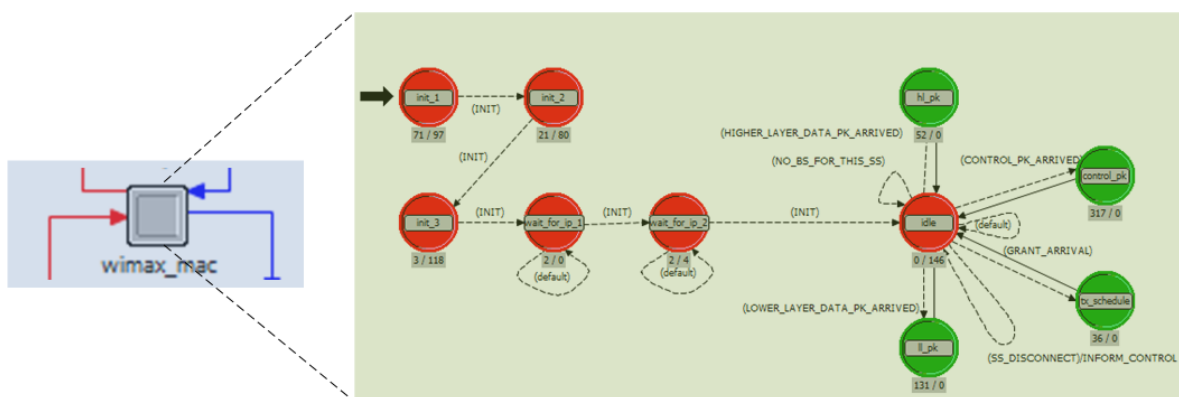


Figura 4.3: Proceso padre *wimax_mac*

Debido a que *wimax_mac* es un proceso padre, es un proceso común para los modelos de nodo de SS y BS. La diferencia en funcionalidad, a nivel de capa WIMAX MAC, la hacen los procesos *hijos* de cada uno de estos modelos.

4.2.1.1.2. Proceso *wimax_ss_control*

Este proceso es el proceso hijo de *wimax_mac* cuando el nodo es una SS (figura 4.4). Por tal motivo, *wimax_ss_control* implementa funciones de Wimax en estaciones suscriptoras, entre las cuales se encuentran:

- Inicialización del Control de Admisión
- Activación de los Flujos de Servicios
- Decodificador de Mapa
- Inicialización de Handover

- Iniciar el modo de búsqueda y/o la búsqueda de estaciones base vecinas
- Ranging (inicial y periódico)
- Selección de Estación Base

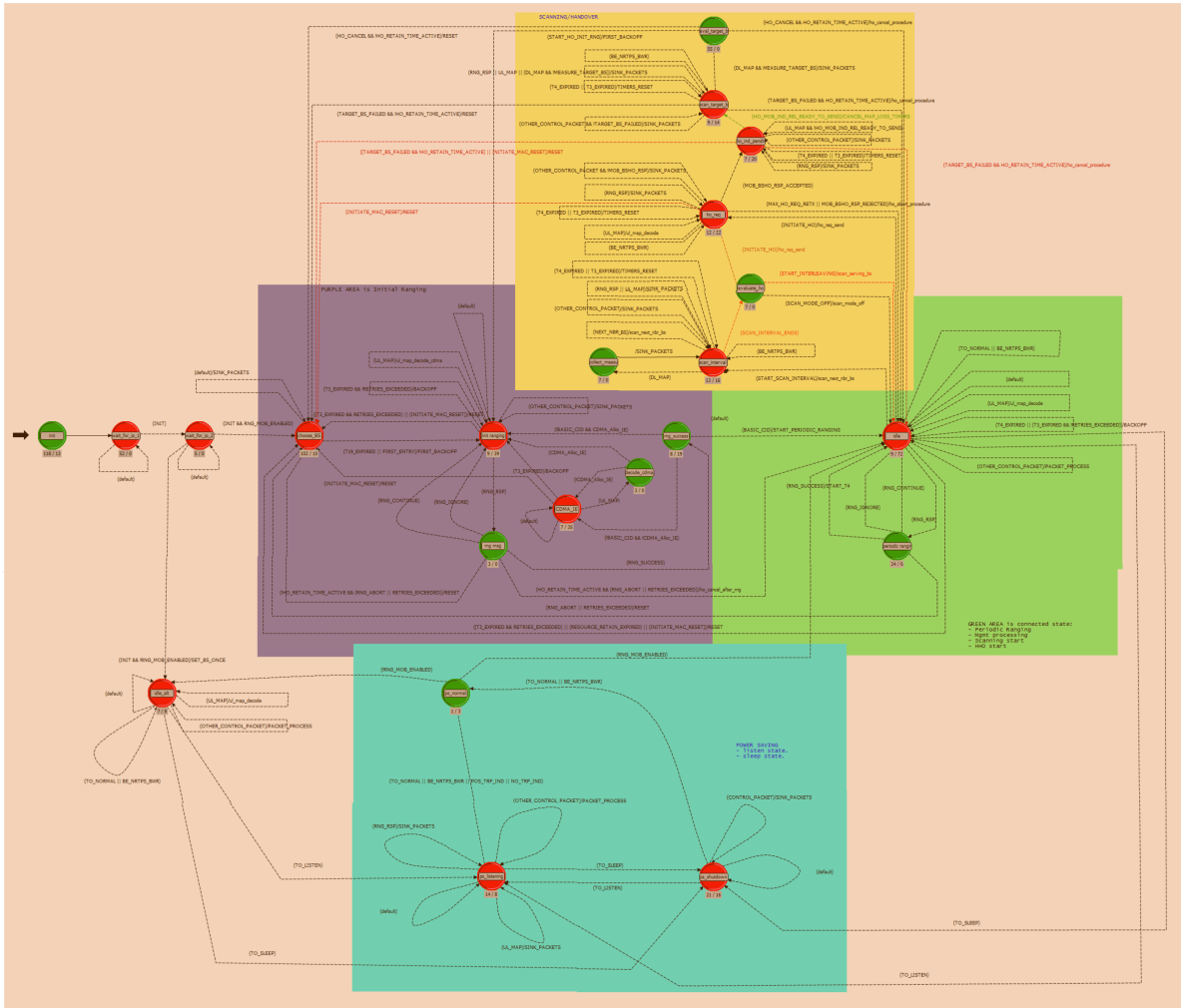


Figura 4.4: Proceso *hijo* wimax_ss_control

4.2.2. Estación Base (BS)

La BS es el nodo que representa al equipo transceptor del proveedor de servicios de telecomunicaciones, ya sean datos, servicios de voz o video conferencias, su respectivo modelo de nodo se presenta en la figura 4.5. Como ya se mencionó, la BS cuenta con un proceso padre *wimax_mac* idéntico al de la SS.

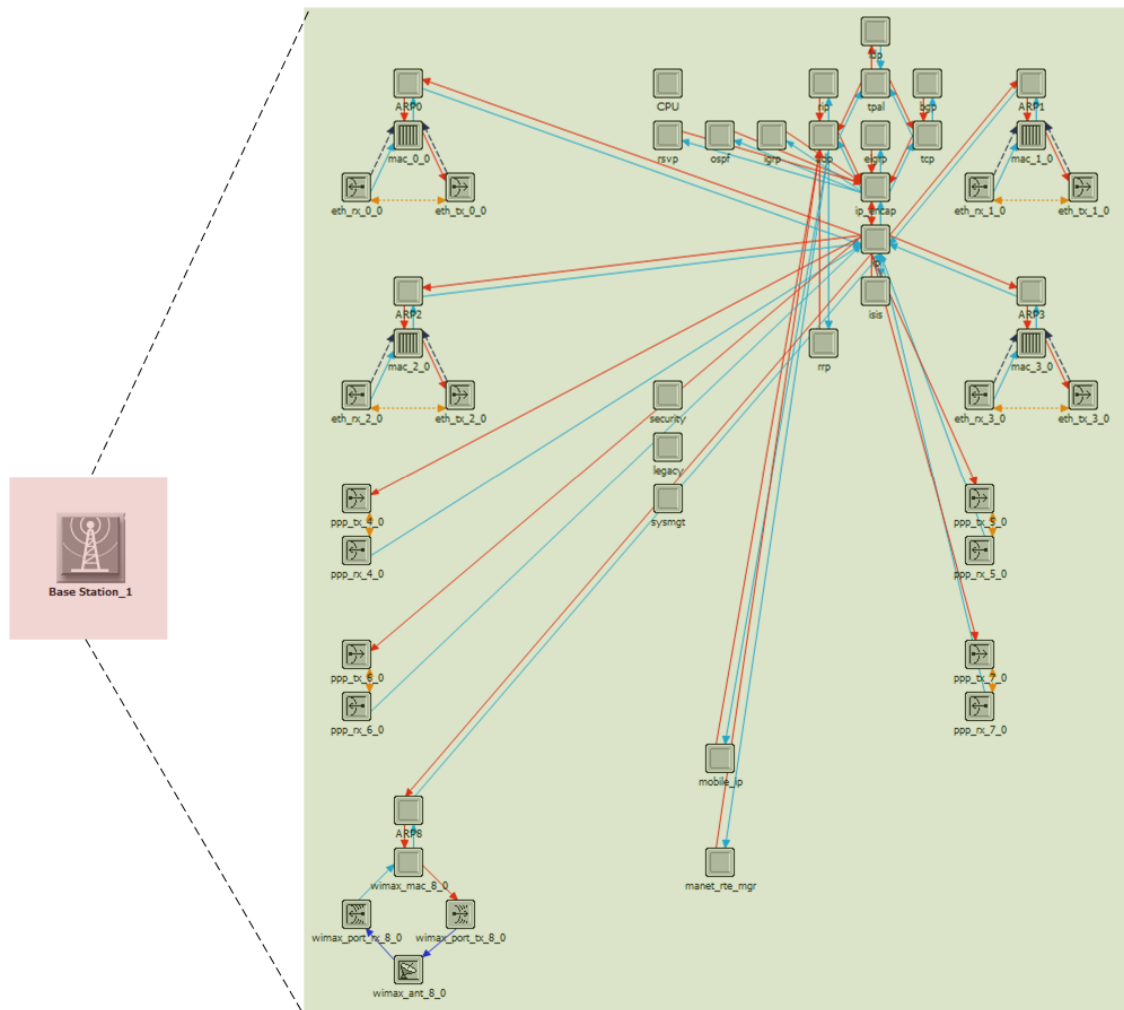


Figura 4.5: La Estación Base en el dominio de red y de nodo

4.2.2.1. Procesos relevantes de la BS

Como es posible observar en la figura 4.5, la BS cuenta con diversos procesos. El proceso más importante en este trabajo es el denominado *wimax_mac_8_0* que finalmente es el mismo proceso *wimax_mac* mencionado anteriormente. Este proceso se muestra explícitamente en la figura 4.3 y, como en el caso de la SS, es el proceso *padre* de la BS.

4.2.2.1.1. Proceso *wimax_bs_control*

Este proceso es el proceso hijo de *wimax_mac* cuando el modelo de nodo es el de la estación base. Entre las funcionalidades de Wimax que se implementan en este proceso, se encuentran:

- Control de Admisión
- Calendarizador para la asignación de Ancho de banda
- Generación de Mapa

- Activación de servicios de flujos admitidos
- Envío de avisos a vecinos
- Ordenar a la SS entrar al estado de búsqueda
- Respuesta a mensajes de ranging
- Liberar los recursos de la SS después de un handover

4.2.3. Características de los modelos de proceso

Los modelos de procesos se desarrollan para representar sistemas planeados o ya existentes en la vida real. En algunos casos, los modelos de procesos son desarrollados para representar procesos hipotéticos que podrían ocurrir en el mundo real. Sin importar lo anterior, las acciones requeridas para la implementación y el tiempo usado para la realización de estas acciones pueden ser bastante complicadas y pueden variar de aplicación a aplicación. El uso de una convención de representación o lenguaje permite que el desarrollo del proceso a implementar sea extremadamente versátil. OPNET Modeler soporta un lenguaje para desarrollar modelos de procesos. Este lenguaje, denominado Proto-C, es soportado por el editor de procesos el cual está integrado en la aplicación de OPNET modeler. Proto-C provee un método eficiente y poderoso para describir el comportamiento de un proceso; ofrece también las siguientes características principales:

- **Basado en diagramas de transición de estados (STDs):** También conocido como máquina de estado finito, este es el método de descripción imperante en sistemas de eventos discretos que mantienen información de estados (por ejemplo, información acumulada en el pasado). Es usado en la especificación y desarrollo de un gran variedad de software y hardware debido a su simplicidad y a su representación gráfica mayormente intuitiva.
- **Representación gráfica/textual combinada:** Los modelos de proceso de Proto-C describen transiciones de estados en un formato gráfico. Sin embargo, como las representaciones gráficas generalmente no describen eficientemente acciones detalladas, Proto-C soporta también especificaciones textuales de ciertos aspectos de un proceso. La especificación textual ocurre dentro de un estado y en las transiciones del diagrama.
- **Capacidad para representar información general de estado:** La información de estado de un proceso es un conjunto de datos que ha sido generada por el proceso mismo o facilitada a él. Esta información es generalmente retenida por el proceso. Los diagramas de transiciones de estado tradicionales representan el estado completo de un sistema, identificando cuál de un limitado conjunto de estados es ocupado en cierto tiempo dado, esto resulta impráctico con estados complejos e insuficiente para sistemas con estados expandiéndose dinámicamente. Proto-C usa la representación de ocupación de estado sólo para representar modos de alto nivel de un sistema y provee un mecanismo para describir información general de estado. Los modelos pueden

declarar tanta cantidad de información de estados como variables cuyo tipo de datos sean provistos por OPNET Modeler y los lenguajes C y C++.

- **Desempeño eficiente:** Los diagramas de Proto-C y la lógica contenida dentro de ellos son convertidos a lenguaje C o C++ con el mínimo gasto computacional, permitiendo una ejecución extremadamente eficiente. En particular, el acceso a información de estados y las declaraciones de flujo de control (por ejemplo, código con condicionales o bucles *for*) son implementadas de forma directa para evitar cualquier pérdida de desempeño.
- **Interrupciones:** En OPNET Modeler, los procesos son manejados por eventos. Cuando un evento es entregado a un proceso, a lo se denomina *Interrupción*, se dice que sufrió una interrupción lo que significa que dicho proceso fue invocado para permitirle tomar acciones como respuesta a la interrupción. Las razones por la que un proceso es interrumpido varía de proceso a proceso; además, los tipos de interrupciones que un proceso puede recibir pueden ser diversas. De hecho, una de la primeras acciones que un proceso toma una vez interrumpido es determinar que tipo de interrupción ha ocurrido.
- **Bloque de funciones para cálculos recurrentes:** El enfoque más común cuando se tiene cálculos recurrentes es hacer uso del mecanismo denominado *invocación de funciones* que los lenguajes C y C++ proveen de forma predeterminada. Las funciones son especificaciones de cálculos que pueden aceptar o no parámetros, conocidos como argumentos. Los argumentos pueden ser de los tipos soportados por C, C++ y OPNET Modeler. Las funciones también pueden devolver valores que resulten de su cálculo o simplemente hacer una series de acciones. En Proto-C, cualquier número de funciones pueden ser especificadas dentro de una sección denominada *function block* de un modelo de proceso. Cabe señalar que el contenido de los bloques de funciones pueden ser editado por el usuario con el fin de modificar el comportamiento de un proceso. Una función definida en el bloque de funciones puede ser llamada desde el STD o desde otra función del mismo bloque de funciones.

Dentro del STD de un proceso, existen elementos básicos e importantes como los que se describen a continuación:

- **Estados:** En términos de un STD, la palabra estado se refiere a un objeto que corresponde a uno de los modos o situaciones en los que un proceso puede caer. Los estados son mutuamente excluyentes y complementarios, lo que significa que un proceso se encuentra siempre en un solo estado. El proceso puede moverse entre los diferentes estados como respuesta a las interrupciones que éste reciba.
- **Directivas:** Diversas especificaciones de acciones pueden ser asociadas con cada estado en Proto-C, estas acciones son denominadas directivas. Las directivas de un estado son divididas en dos secciones, llamadas *directivas de entrada* y *directivas de salida*. Como su nombre lo indica, un directiva de entrada es ejecutada cuando un proceso entra en un estado y una directiva de salida cuando el proceso abandona dicho estado.

- **Transiciones:** Las transiciones describen el posible desplazamiento de un proceso de un estado a otro y las condiciones bajo la cuales tales cambios se llevarán a cabo. Hay cuatro componentes para la especificación de una transición: un estado fuente, un estado de destino, una expresión de condición y una expresión de directiva. La especificación puede ser leída de la siguiente forma: Cuando se esté en el estado fuente y se cumpla la condición, implementar la expresión de directiva y transferir el control al estado de destino.

4.3. Solicitud de recursos en Wimax

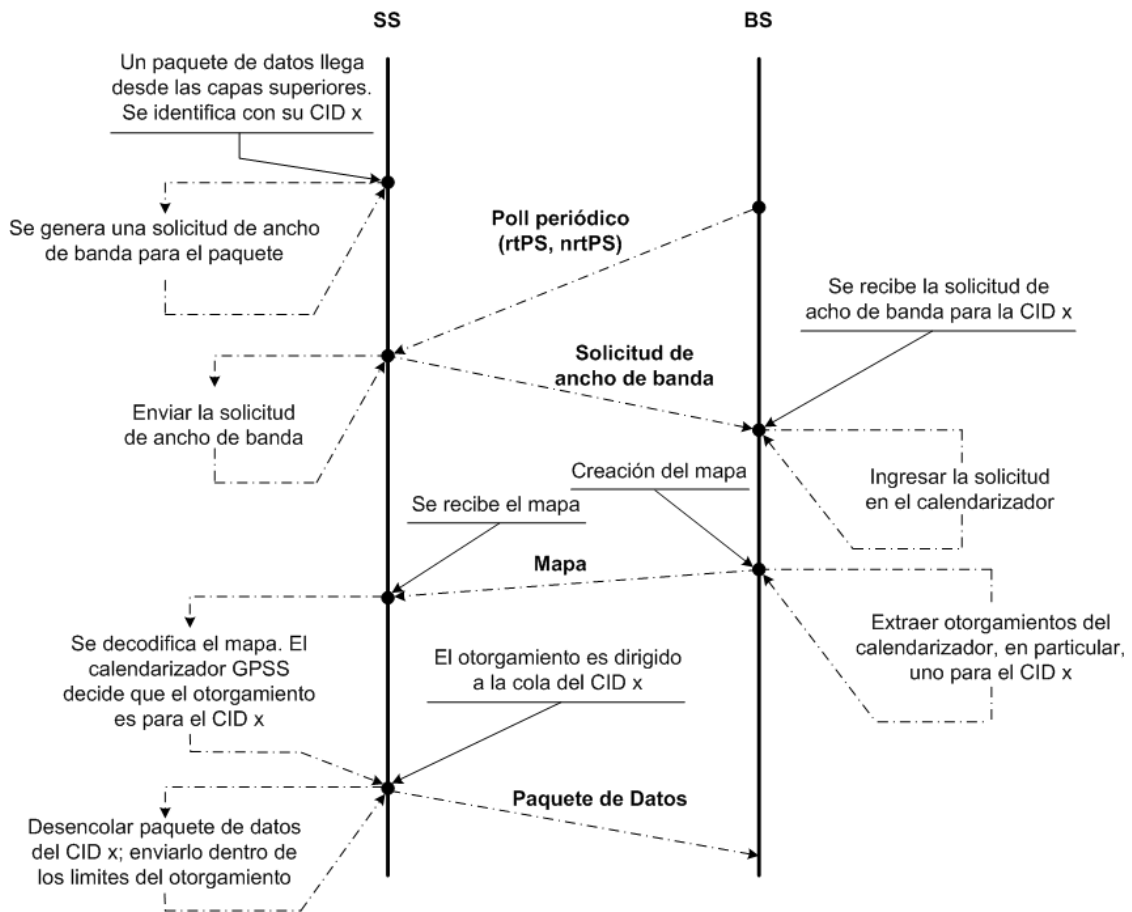


Figura 4.6: Solicitud y Asignación de recursos en UL

En la sección anterior se describió la estructura en OPNET de una red Wimax. Cada una de las partes detalladas anteriormente serán de gran utilidad para comprender a fondo la forma en que se solicitan y se asignan los recursos de ancho de banda en una red Wimax.

La Capa WIMAX MAC es parte primordial de un red de este tipo debido a que es la que se encarga de administrar los recursos de dicha red. Por lo anterior, la descripción presentada en esta sección se centra solamente en el comportamiento del proceso o procesos

que simulan la capa WIMAX MAC. En este punto, es preciso recordar que la SS, al igual que la BS, posee un proceso hijo y un proceso padre a nivel de capa MAC. El proceso padre es el mismo para ambas; sin embargo, el proceso hijo es totalmente diferente como ya se mencionó anteriormente.

En las descripciones y análisis que a continuación se presentan, se mencionarán elementos del STD de un proceso en OPNET, es decir, se indicará en que estado se encuentra el proceso en cuestión, que condiciones se cumplieron y que directivas de entrada o de salida serán ejecutadas. Las invocaciones a funciones también serán indicadas con el fin de obtener un análisis bastante detallado.

Cabe señalar que, como esta sección sólo aborda la solicitud de recursos, los tipos de calendarización involucrados serán rtPS, nrtPS y Best Effort. ertPS y UGS no son abordados en esta sección debido a que dichas clases de servicio poseen un servicio periódico y no es necesaria la solicitud de recursos. Las figuras 4.6 y 4.7 muestran un diagrama general de eventos durante la solicitud de recursos para UL y DL respectivamente.

4.3.1. Un paquete llega de las capas superiores

Imaginemos que un paquete de datos es generado por la capa de aplicación de cualquier nodo de una red Wimax, no importa si se trata de una conexión UL o DL. Como el paquete se encuentra en este momento en la capa más alta de la pila de protocolos, éste tendrá que descender y atravesar las capas inferiores (por ejemplo, TCP, UDP, IP, etc) hasta llegar a la capa WIMAX MAC.

El paquete llega a la capa WIMAX MAC en forma de una interrupción tipo *stream* correspondiente a un flujo de datos. El proceso *wimax_mac* se encontrará en el estado *idle*, figura 4.3, y al sufrir una interrupción tipo *stream*, ejecutará las directivas de salida de dicho estado. Se verificará cual de las posibles condiciones se cumplen para una posibles transición. La única condición que se cumplirá será la siguiente: *HIGHER LAYER DATA PK ARRIVED*, lo que provocará la ejecución de las directivas de entrada del estado *hl_pk*.

4.3.1.1. Directivas de entrada de *hl_pk*

Dentro de esta directiva de entrada, dos operaciones principales son aplicadas al paquete entrante:

- Clasificación, con el fin de encontrar su CID.
- Conducirlo al procesamiento correspondiente a la capa MAC.

La función *wimax_mac_higher_layer_pk_classify ()* encuentra el CID correspondiente a la conexión que el paquete en cuestión debe tomar a lo largo de la capa MAC.

Una vez que el CID para el paquete actual fue encontrado, la función *wimax_support_mac_connection_pk_enqueue ()* permite encolar el paquete en la capa MAC. En este punto, a partir del

CID, la BS encuentra información de la cola que alberga los paquetes de la misma conexión, y el flujo de servicio para dicha conexión.

Como el paquete viene de capas superiores, es considerado un MAC SDU y así es encolado por la función *wimax_support_mac_pk_in_queue_insert ()*, la cual es responsable de encolar el paquete en una cola de datos que se encuentra en un componente de la conexión en cuestión denominado elemento moldeador-cola. Las acciones que se toman, una vez que el paquete es encolado, varían dependiendo de tipo de calendarización.

- Si el tipo de calendarización es UGS, el paquete se inserta en la cola de datos y se hace un control relacionado a la naturaleza de *auto-servicio* de las colas UGS. Por *auto-servicio*, se denomina a la propiedad de una cola UGS para servir paquetes a una tasa constante.
- Si el tipo de calendarización es rtPS, nrtPS o BE, una solicitud de ancho de banda (BWR) es generada e insertada en una cola apropiada para solicitudes de ese tipo. Lo anterior se realiza porque las conexiones de estos tipos de calendarización dependen de respuestas a BWRs para poder desencolar paquetes de sus colas.

Si se considera que el tipo de calendarización del paquete proveniente de capas superiores es rtPS, es necesario invocar la función *wimax_support_bw_request_insert ()* con el fin de crear e insertar una nueva solicitud de ancho de banda en la colas de solicitudes. Esta función inserta una nueva BWR para los siguientes dos casos:

1. Este proceso pertenece a una BS. La BWR es transferida localmente al mecanismo otorgador de recursos.
2. Este proceso NO pertenece a una BS, por lo que pertenece a una SS. La BWR necesita ser encolada en una cola apropiada para solicitudes de ancho de banda. Para colas de BWRs, los paquetes de solicitud son enviados de la SS a la BS donde son dirigidos al mecanismo otorgador de recursos.

4.3.1.1.1. La capa MAC pertenece a una BS

Se invoca al proceso hijo, *wimax_bs_control*. Durante la invocación se le notifica al mecanismo otorgador de recursos que una solicitud de ancho de banda de UL acaba de llegar. El proceso *wimax_mac* entrega el control al proceso *wimax_bs_control*

4.3.1.1.2. La capa MAC pertenece a una SS

Mediante la función *wimax_support_update_bwr_requirements ()*, se actualiza la cantidad de requerimientos de ancho de banda después de la inserción del último SDU. Esta función estima y actualiza los requerimientos reales de ancho de banda para conexiones BE, rtPS y nrtPS. La estimación es usada por el subsistema de solicitudes de ancho de banda con el fin de hacerse de recursos para la conexión correspondiente.

Como se trata de un paquete BWR insertado en a cola de un SS, se debe invocar el proceso *wimax_ss_control* ya que el arribo de esta BWR despertará la SS e interrumpirá su ciclo de ahorro de energía.

4.3.2. Solicitud de ancho de banda: Caso DL

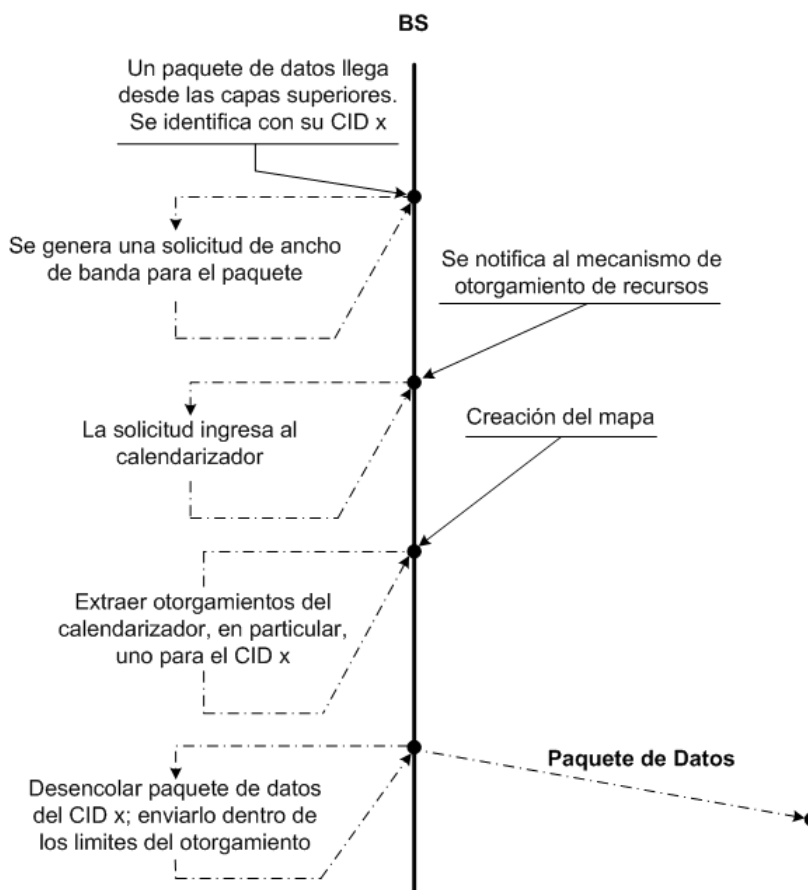


Figura 4.7: Solicitud y Asignación de recursos en DL

En esta sección se aborda el caso en el que una BWR se genera en la capa MAC de una BS. Como la BS es la encargada de asignar los recursos de la red, se dice que esta solicitud es local. Cuando el proceso *wimax_bs_control* es invocado por *wimax_mac* debido a la llegada de una solicitud de ancho de banda local, el plano de control toma el mando de la capa WIMAX MAC. La invocación mencionada llega en forma de un interrupción tipo *stream* y provoca que el proceso *wimax_bs_control* salga del estado *idle* y por tanto se ejecuten sus directivas de salida. La directiva de salida de *idle* sólo tiene la función de recuperar el paquete BWR transferido durante la invocación realizada previamente. Con el fin de identificar claramente las invocaciones de funciones en este caso, el anexo B.1 presenta un diagrama de árbol que muestra también los diferentes estados y procesos involucrados en las solicitudes locales de ancho de banda.

Una vez completada la directiva de salida, se verificará que condición se cumple en ese momento. Para nuestro caso, la única condición que se cumplirá será la siguiente: *LOCAL_BW_REQ_IND*, la cual tiene asociada la directiva denominada *BW_REQ_PROCESS*, que indica el llamado de la función *wimax_bs_control_bw_req_indication_process ()*.

4.3.2.1. Directiva *BW_REQ_PROCESS*

Esta directiva, en realidad, es la invocación de la función *wimax_bs_control_bw_req_indication_process ()*. Esta función procesa una indicación de una BWR local. La primera acción a realizar es notificar al mecanismo de otorgamiento de recursos, lo cual se lleva a cabo llamando a la función *wimax_bs_control_sched_bw_req_notify ()*. En esta función, parámetros como CID, tipo de calendarización y tamaño de BWR asociado, son transferidos al sistema calendarizador de solicitudes. Con dichos parámetros, se actualizan los requerimientos de BW para la conexión correspondiente; la función *wimax_bs_control_bw_req_requirements_update ()* se encarga de esta tarea.

En la función *wimax_bs_control_bw_req_requirements_update ()*, la información acerca de la BWR es recibida. De la misma forma, los requerimientos de ancho de banda para la conexión correspondiente son actualizados fragmentando, si es necesario, la solicitud de BW actual en elementos más pequeños, similares al tamaño de SDU promedio especificado para esta conexión. Cabe señalar que los requerimientos de ancho de banda se incrementarán sólo si la solicitud de ancho de banda es incremental o si es agregada y los nuevos requerimientos son mayores que los actuales. Una BWR es incremental cuando proviene de una solicitud piggyback o de una solicitud hecha por la propia BS; por otra parte, una BWR es agregada cuando es resultado de un proceso de *polling*.

El incremento de los requerimientos de BW implican la inserción de la solicitud de ancho de banda, completa o fragmentada, en la cola correspondiente. Esta tarea es realizada por la función *wimax_bs_control_sched_bw_req_insert ()*. Dentro de esta función, primero se crea el *elemento solicitud de ancho de banda* que es él que posteriormente es insertado en la cola correspondiente dependiendo el tipo de calendarización de la conexión, lo que es realizado por la funciones *wimax_bs_control_sched_bw_req_ps_insert* y *wimax_bs_control_sched_bw_req_be_insert ()* para conexiones Polling Service(PS) y Best Effort respectivamente.

En caso de que la BWR actual sea agregada y sus requerimientos de ancho de banda sean menores a los requerimientos registrados en sistema calendarizador, los nuevos requerimientos serán los indicados por la BWR, lo que que significará una reducción en dichos requerimientos.

4.3.3. Solicitud de ancho de banda: Caso UL

En esta sección se aborda el caso en el que una solicitud de ancho de banda es generada por la capa MAC de una SS. Este tipo de solicitudes se denominan solicitudes remotas y siempre representan peticiones de recursos para conexiones UL. En la sección 4.3.1, se mencionó que cuando una BWR es generada en una SS, se interrumpe el ciclo de ahorro de energía de la SS. A partir de ese momento la SS estará en espera de un *poll* que le dé la oportunidad de notificar a la BS su necesidad de recursos de la red. El anexo B.2 presenta un diagrama de árbol de permite identificar claramente las invocaciones de funciones en este caso.

4.3.3.1. Llegada de un *poll* a la SS

La llegada de un *poll* a la SS está representado por la llegada de un *frame*. Un *frame* puede o no contener *polls* dirigidos a una SS que le permitan informarle a la BS su necesidad de ancho de banda.

Cuando un *frame* llega a la SS, una interrupción tipo *stream* hace que el proceso *wimax_mac* salga del estado *idle* y ejecute las directivas de salida de este estado para luego evaluar la condición que le permita pasar a un estado vecino. La condición que se cumplirá, en este caso, será la siguiente: *CONTROL_PK_ARRIVED*, lo que desencadenará la directivas de entrada del estado *control_pk*, las cuales identificarán la información recibida, verificarán su validez y que pueda ser enviada al plano de control, es decir, al proceso *wimax_ss_control*. Si no hay ningún problema, el proceso es invocado con éxito.

Cuando el proceso *wimax_ss_control* sea invocado, éste saldrá del estado *idle* bajo la condición *UL_MAP* que permitirá la ejecución de la directiva *ul_map_decode*.

4.3.3.2. Directiva *ul_map_decode*

La directiva *ul_map_decode*, en realidad hace referencia a la función *wimax_ss_control_ul_map_decode ()*, la cual decodifica el UL-MAP que se encuentra en el *frame* que acaba de llegar. Dentro del mapa existen *Elementos de Información (IE)* que contienen información de los recursos asignados a una SS en el *frame* actual. Cada IE del mapa corresponde a un *otorgamiento por SS (GPSS)*. Los *polls*, en forma de IE, contenidos en el mapa, son decodificados mediante la llamada de las función *wimax_ss_control_grants_decode ()* usando el argumento *WimaxC_Poll*. En este punto, se descomponen los IEs en otorgamientos por conexión (GPC) y luego se entregan a la conexión correspondiente mediante la función *wimax_ss_control_gpss_decompose ()*.

Una vez que los IEs se descomponen, son entregados al plano de datos, es decir, al proceso padre *wimax_mac* de la SS en cuestión. Lo anterior se realiza mediante una interrupción tipo *remota*.

Una vez que el proceso padre recibe los *polls* transferidos por el proceso hijo, se ejecutan las directivas de salida del estado *idle* de *wimax_mac*. Después, se identificará la condición que se cumple para un posible cambio de estado. La llegada de un *poll* al proceso *wimax_mac* tiene la forma de un otorgamiento. Los otorgamientos pueden ser de otro tipo; sin embargo, en el caso actual, los otorgamiento sólo serán *polls*.

La condición que se cumplirá será la siguiente: *GRANT_ARRIVAL*, lo que desencadenará la ejecución de la directivas de entrada del estado *tx_schedule*.

4.3.3.3. Directiva de entrada de *tx_schedule*

Estas directivas tiene la función de leer los *polls* otorgados que han llegado a la SS. Como los IEs ya fueron descompuestos en GPCs, estos son transferidos a la función *wimax_mac_efficiency_off_grants_process ()* en una sola lista de otorgamientos. Esta función, a

su vez, llama a la función *wimax_mac_efficiency_off_grant_process ()* con el fin de analizar los otorgamientos de forma individual e identificar su tipo, que para el caso actual corresponde a *WimaxC_Poll*.

La identificación de cada GPC permite que el *poll* sea dirigido a la cola de BWR correspondiente, que se encuentra en la SS, y poder aprovecharlos mediante la creación de un paquete BWR que será enviado a la BS. Lo anterior es realizado por la función *wimax_mac_poll_consume*.

Después de haber creado el paquete de BWR, la capa MAC lo envía a la capa Física mediante la función *wimax_support_pk_send*.

4.3.3.4. Solicitud Remota llega a la BS

Una solicitud de ancho de banda, proveniente de una SS, llega a la BS por el medio de transmisión y arriba a la capa MAC por medio de la capa física. Esta acción se presenta en el proceso *wimax_mac* de la BS como una interrupción *stream* que provoca la ejecución de las directivas de salida del estado *idle*, donde se identifica que el paquete que acaba de llegar proviene de la capa inferior y que el nombre de su formato es *wimax_mgmt_bw_req*. La evaluación de condiciones mostrará que la condición que se cumple es *CONTROL_PK_ARRIVED*, lo que producirá la ejecución de las directivas de entrada del estado *control_pk*.

4.3.3.5. Directiva de entrada de *control_pk*

Las directivas de este estado tiene la función de verificar si el paquete de BWR es valido para, posteriormente, enviarlo al plano de control, es decir, invocar al proceso *wimax_bs_control*

Cuando llega una interrupción tipo *stream* al proceso *wimax_bs_control*, las directivas de salidas del estado *idle* se ejecutan. Posteriormente, la directiva *PACKET_PROCESS* es ejecutada debido a que la condición que deberá cumplirse es *CONTROL_PACKET*.

4.3.3.6. Directiva *PACKET_PROCESS*

Esta directiva, en realidad, es la invocación de la función *wimax_bs_control_packet_process ()*, quien tiene el objetivo de procesar el paquete de control que llegó a la BS. Como el paquete de control que acaba de llegar es en realidad un paquete BWR remoto, entonces la función *wimax_bs_control_bw_req_process ()* es requerida para obtener del paquete, el CID y la cantidad de ancho de banda solicitada. Con esta información, es posible notificar al mecanismo otorgador de recursos la llegada de la nueva BWR. Lo anterior se realiza con la invocación de la función *wimax_bs_control_sched_bw_req_notify ()*.

Dentro de la función *wimax_bs_control_sched_bw_req_notify ()*, como en el caso UL, parámetros como CID, tipo de calendarización y tamaño de BWR asociado, son transferidos al sistema calendarizador de solicitudes. Con dichos parámetros, se actualizan los requerimientos

de ancho de banda para la conexión correspondiente; la función *wimax_bs_control_bw_req_requirements_update ()* se encarga de esta tarea.

En la función *wimax_bs_control_bw_req_requirements_update ()*, la información acerca de la BWR es recibida. De la misma forma, los requerimientos de ancho de banda para la conexión correspondiente son actualizados fragmentando, si es necesario, la solicitud de BW actual en elementos más pequeños, similares al tamaño de SDU promedio especificado para esta conexión. Cabe señalar que los requerimientos de ancho de banda se incrementarán sólo si la solicitud de ancho de banda es incremental o si es agregada y los nuevos requerimientos son mayores que los actuales. Una BWR es incremental cuando proviene de una solicitud piggyback; por otra parte, una BWR es agregada cuando es resultado de un proceso de *polling*.

El incremento de los requerimientos de BW implican la inserción de la solicitud de ancho de banda, completa o fragmentada, en la cola correspondiente. Esta tarea es realizada por la función *wimax_bs_control_sched_bw_req_insert ()*. Dentro de esta función, primero se crea el *elemento solicitud de ancho de banda* que es él que posteriormente es insertado en la cola correspondiente dependiendo el tipo de calendarización de la conexión, lo que es realizado por la funciones *wimax_bs_control_sched_bw_req_ps_insert* y *wimax_bs_control_sched_bw_req_be_insert ()* para conexiones Polling Service(PS) y Best Effort respectivamente.

En caso de que la BWR actual sea agregada y sus requerimientos de ancho de banda sean menores a los requerimientos registrados en sistema calendarizador, los nuevos requerimientos serán los indicados por la BWR, lo que que significará una reducción en dichos requerimientos.

4.4. Asignación de recursos en Wimax

Una vez que ha hecho solicitudes de ancho de banda, una SS se encuentra en espera de que dichas solicitudes sean concedidas. En esta sección, se explicará, de forma detallada, el procedimiento que una BS ocupa para asignar recursos a las diferentes SSs.

4.4.1. Creación del mapa

Como ya se mencionó anteriormente, una mapa es un mensaje de administración de la capa MAC que contiene información acerca de la estructura del frame. Los mapas pueden ser para uplink y para downlink, denominándose UL-MAP y DL-MAP, respectivamente. El DL-MAP indica los tiempos de inicio de ráfaga en el downlink. De igual forma, el UL-MAP proporciona información que define el acceso de las SSs durante un intervalo determinado.

Los mapas son creados por la BS cada tiempo de frame, es decir, si el frame dura 5[ms], la BS creará los mapas cada 5[ms]. En la BS debe existir un temporizador que indique que es momento de crear un mapa.

En OPNET Modeler, la indicación de que el tiempo de crear un mapa ha llegado, se presenta como una interrupción tipo *self* en el proceso hijo *wimax_bs_control*, lo cual provoca que

se ejecuten las directivas de salida del estado *idle* y posteriormente se cumpla la condición *MAP_TIME* y se ejecuten las directivas *MAP_GENERATE*.

4.4.1.1. Directivas *MAP_GENERATE*

Las directivas *MAP_GENERATE*, en realidad, son la invocación de la función *wimax_bs_control_map_process ()* que tiene la tarea de generar mapas. Dentro de esta función, la función *wimax_bs_control_map_generate ()* es invocada para generar los mapas usando el calendarizador. Cabe señalar que el UL-MAP necesita ser generado antes que el DL-MAP para que el tamaño del UL-MAP sea conocido y pueda ser restado como encabezado del DL. Debido a lo anterior, la función *wimax_bs_control_map_generate ()* es llamada en dos ocasiones, primero para generar el UL-MAP y después, para generar el DL-MAP.

Como la configuración de la red, en nuestro caso, es OFDMA, la función *wimax_bs_control_ofdma_map_generate ()* será necesaria. Es necesario mencionar que esta función puede generar mapas para OFDMA utilizando uno o dos calendarizadores. Se tiene un solo calendarizador cuando el límite UL/DL es variable y dos calendarizadores cuando dicho límite es fijo. Por cuestiones de simplicidad, el presente trabajo sólo contempla el caso en el que el límite UL/DL es fijo.

Como el límite UL/DL ha sido definido como fijo, sólo se podrá generar un mapa por vez. Por ese motivo, se llama a la función *wimax_bs_control_one_ofdma_map_generate ()* la cual se encargará de invocar al calendarizador quien asignará los recursos a las diferentes SSs.

4.4.1.2. Inserción de otorgamientos periódicos: Polls rtPS y nrtPS, ertPS y UGS

El número de acciones realizadas por la función *wimax_bs_control_one_ofdma_map_generate ()* es bastante elevado. Por tal motivo, en esta sección sólo se abordarán las acciones más importantes y relevantes para el presente trabajo.

En el momento en que el temporizador de mapa se activa, la línea de ensamblaje (staging buffer) debe ya contener elementos de solicitud correspondientes a otorgamientos periódicos programados con anticipación. Estos elementos de solicitud periódicos pueden ser de UGS, ertPS y polls de rtPS o nrtPS; estos últimos sólo si se trata de un UL-MAP.

Los primeros elementos de solicitud a ser considerados en el mapa son los que ya se encuentran en la línea de ensamblaje, es decir las asignaciones periódicas de UGS, ertPS y polls rtPS y nrtPS, figura 4.8. El vaciado de la línea de ensamblaje implica un desencolamiento en el que cada elemento de solicitud será procesado por separado. El primer elemento en ser procesado será aquel que se encuentre en la cabeza de la línea de ensamblaje.

El procesamiento por separado de cada elemento de solicitud tiene el objetivo de acomodar dicha solicitud en el mapa. Durante este proceso, los puntos de inserción actual y el total de símbolos libres son actualizados. Esta tarea es realizada por la función *wimax_bs_control_ofdma_req_elem_process ()*.

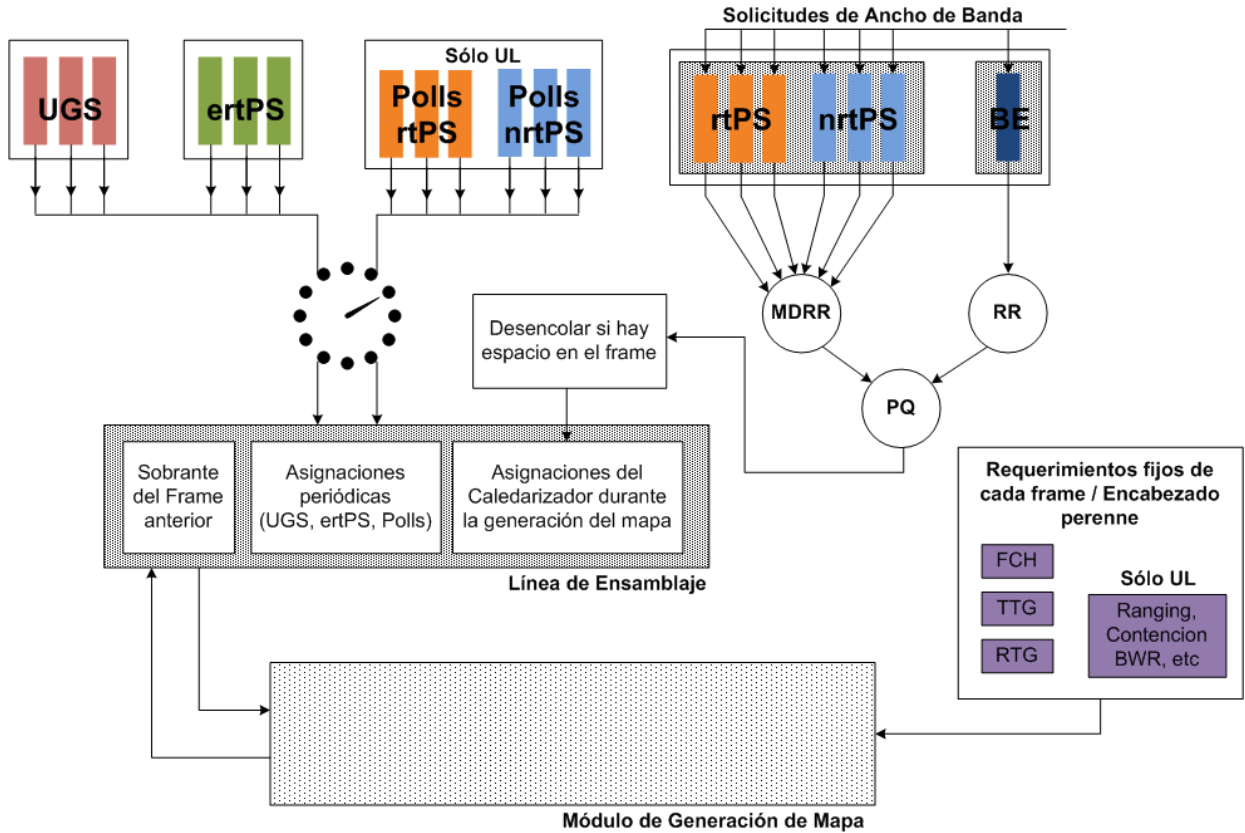


Figura 4.8: Generación de Mapa y Calendarizador de la BS

4.4.1.2.1. Función *wimax_bs_control_ofdma_req_elem_process 0*: Caso UL

Para UL, es posible agrupar varias ráfagas destinadas al mismo CID Básico, es decir una misma SS, en un área contigua y hacer sólo un IE. Si se mantiene sólo un IE por CID Básico, se necesita recordar si la IE ya fue creada anteriormente.

En el primer intento de acomodar el elemento de solicitud en el mapa, se trata de hacer una inserción de forma completa. El procedimiento para *encajar* una solicitud completa, en OFDMA, sigue más o menos el mismo procedimiento que cuando se trabaja con SC. La asignación en UL de OFDMA va de derecha a izquierda en cada fila. Debido a la similitud en los procedimientos de inserción entre SC y OFDMA, la función invocada para hacer esta acción es *wimax_bs_control_sc_fit_whole ()*. Si la solicitud cupo de forma completa en el frame, se crea el IE respectivo y se añade al mapa.

Si la solicitud no cabe de forma completa, se intenta la inserción de forma parcial. Como es UL, esta forma de inserción toma lugar sobre todos los símbolos restantes en el UL. Lo anterior es realizado por la función *wimax_bs_control_sc_partial_fit ()*. Si la solicitud cupo de forma parcial, se crea el IE respectivo y se añade al mapa. La fracción de solicitud que no cupo en el frame se coloca de vuelta en la línea de ensamblaje con ayuda de la función *wimax_bs_sched_staging_buffer_insert ()*. Como la solicitud fue acomodada de forma parcial, todo indica que no queda espacio libre en el frame por lo que es momento de finalizar la construcción del mapa.

4.4.1.2.2. Función *wimax_bs_control_ofdma_req_elem_process 0*: Caso DL

Cuando se construye el DL-MAP, la verificación de inserción completa de la solicitud está influenciada por la condición de que la ráfaga a asignar sea rectangular. La rectangulación usada en este punto no es óptima. La función *wimax_bs_control_dl_ofdma_fit_whole ()* es la encargada de realizar esta acción. En caso de que la solicitud quepa de forma completa en el frame, se crea el IE correspondiente y se añade al mapa con ayuda de la función *wimax_bs_control_dl_ofdma_ie_add ()*. Esta función añade un elemento de información a un DL-MAP OFDMA. Este IE hace referencia a una ráfaga de datos asignada durante la creación del mapa.

Si la solicitud no cabe de forma completa en el frame, se intenta una inserción parcial de ésta mediante la función *wimax_bs_control_ofdma_partial_fit ()*. Si la inserción parcial es posible, se crea un IE para la solicitud parcial acomodada y se añade al mapa. La fracción de solicitud que no cupo en el frame se coloca de vuelta en la línea de ensamblaje con ayuda de la función *wimax_bs_sched_staging_buffer_insert ()*. Como la solicitud fue acomodada de forma parcial, todo indica que no queda espacio libre en el frame por lo que la construcción del mapa finaliza.

4.4.1.3. Inserción de solicitudes de rtPS, nrtPS y BE

En este punto, se ha finalizado de acomodar todas las solicitudes contenidas en la línea ensamblaje. Si hay espacio disponible, se intentará acomodar solicitudes de las colas de rtPS, nrtPS y BE. Se verifica si hay alguna solicitud pendiente en PS o en BE. La lógica que se sigue en UL-MAP es diferente a la utilizada en DL-MAP debido a que en DL-MAP se tratará de unir las solicitudes cuando sea posible. Éste es el último paso de la generación del mapa, lo siguiente es enviarlo a las SS pertenecientes a esta BS.

4.4.1.3.1. Caso UL

La solicitud es extraída con la función *wimax_bs_control_sched_bw_req_dequeue ()*. Debido a su importancia para el presente trabajo, esta función es explicada a fondo en la siguiente sección. Una vez que la solicitud es descolada, la función *wimax_bs_control_ofdma_req_elem_process ()* trata de insertarla en UL, tal y como se explica en la sección 4.4.1.2.

4.4.1.3.2. Caso DL

A diferencia de UL, en DL, se desencolan tantas solicitudes como existan en el sistema calendarizador sin importar si éstas caben en el frame. Cada una de estas solicitudes es seleccionada por la función *wimax_bs_control_sched_bw_req_dequeue ()* y puesta en la línea de ensamblaje, la cual está vacía debido a que fue descargada completamente en acciones anteriores. Una acción importante antes de desencolar la próxima solicitud es estimar si todavía hay espacio restante en el frame considerando la solicitud actual. La función *wimax_bs_control_ofdma_subframe_capacity_estimate ()* realiza esta tarea y en caso de no haber espacio restante la acción es detenida.

Cuando todas las solicitudes posibles fueron extraídas del sistema calendarizador y puestas en la línea de ensamblaje, el siguiente paso es colocar el contenido de la línea de ensamblaje

en el frame, en caso de que no todo quepa de forma completa, como en casos anteriores, se intentará la inserción en forma parcial y el sobrante se devolverá a la línea de ensamblaje.

4.4.2. Función *wimax_bs_control_sched_bw_req_dequeue ()*

Esta función se encarga de obtener la nueva BWR a ser servida y la transfiere a quien invocó la función. Dentro de esta función se encuentra el algoritmo calendarizador que selecciona la cola a servir, lo que implica la aplicación de la QoS y que es parte fundamental de este trabajo. Por tal motivo, la descripción aquí mostrada es sumamente detallada.

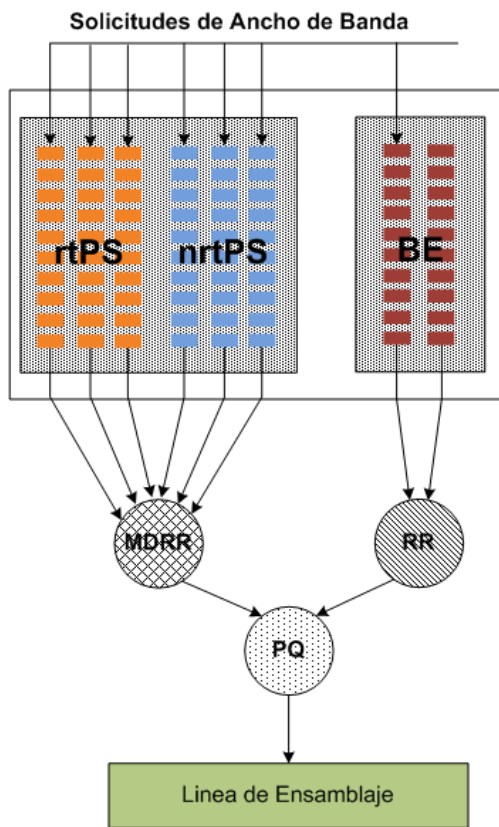


Figura 4.9: Calendarizador de la BS

4.4.2.1. Sistema Calendarizador de la BS

La figura 4.9 muestra el sistema calendarizador de la BS implementado por OPNET Modeler. Este sistema tiene la función de seleccionar de entre las colas de conexiones activas, la solicitud que será atendida. Las colas están identificadas con un número de cola, tienen un CID relacionado y pertenecen a uno de los siguientes grupos:

- **PS (Polling Service):** Está conformado por las colas cuya conexión posee un tipo de calendarización rtPS o nrtPS.

- **BE (Best Effort):** Está conformado por las colas cuya conexión posee un tipo de calendarización Best Effort.

El sistema calendarizador presentado en la figura 4.9 posee en realidad tres algoritmos de calendarización: PQ (Priority Queue), RR (Round Robin), y MDRR (Modified Deficit Round Robin), los cuales son explicados a continuación.

4.4.2.1.1. Algoritmo PQ (Priority Queue)

Este algoritmo aísla conexiones de alta demanda del resto de conexiones, con el fin de ofrecer mayor calidad de servicio. Hay dos modos de servicio en PQ:

- **Modo Alternado:** En este modo, las colas con alta prioridad, PS en nuestro caso, son servidas de forma alternada con las colas de baja prioridad, BE.
- **Modo Prioridad Estricta:** En este modo, ninguna otra cola es atendida hasta que las colas PS sean completamente servidas. Éste es el modo en el que trabaja la implementación de OPNET.

4.4.2.1.2. Algoritmo RR (Round Robin)

Después de FIFO (First In First Out), el algoritmo RR puede ser considerado como el algoritmo calendarizador más básico y menos complejo. RR maneja turnos, en cada turno el apuntador del calendarizador se encuentra en una cola en particular, un paquete es desencolado de dicha cola y luego, el apuntador se desplaza a la siguiente cola.

Una desventaja de RR es que puede asignar recursos a conexiones con colas que pueden no tener nada que transmitir. Por lo tanto, algunas modificaciones son necesarias para que el algoritmo salte las conexiones inactivas y considere solamente las conexiones activas.

Otro inconveniente es que RR no ofrece equidad en sistemas con longitud de paquetes variable, ya que solamente se encarga de desencolar un paquete sin considerar su tamaño, lo que puede provocar que una conexión consuma la mayor parte del ancho de banda disponible debido a la longitud elevada de sus paquetes.

4.4.2.1.3. Algoritmo MDRR (Modified Deficit Round Robin)

El algoritmo MDRR es una extensión del algoritmo DRR (Deficit Round Robin). DRR es una versión mejorada de RR que incluye un *contador de déficit*. El contador es inicializado por un valor llamado *peso* que refleja el ancho de banda reservado para la conexión en cuestión. El DRR visita cada cola y si la cola está vacía y su *contador de déficit* es menor que su *peso*, al *contador de déficit* se le incrementará el valor del *peso*. Bajo cualquier circunstancia, cuando el calendarizador visita una cola vacía, dicha cola es ignorada para la asignación de recursos.

Si la cola no está vacía y su *contador de déficit* es menor a cero, al *contador de déficit* se le incrementará el valor del *peso*. Si el nuevo valor del *contador de déficit* sigue siendo

negativo, la cola es descartada y el apuntador del calendarizador se desplaza a la siguiente cola. De otro modo, la cola actual será servida.

Una vez seleccionada la cola que será servida, la solicitud que se encuentra en la cabeza de ésta es desencolada. El tamaño en símbolos de la solicitud de ancho de banda es descontada del *contador de déficit* de la cola en cuestión. Si el nuevo valor del *contador de déficit* es positivo, otra solicitud podrá ser desencolada y así sucesivamente, hasta que el valor del *contador de déficit* se vuelva negativo o la cola quede vacía.

El *peso* que se asigna a cada cola esta definido por la siguiente ecuación (4.1):

$$w_i = MTU + 512 * \lfloor Q_i \rfloor \quad (4.1)$$

$$Q_i = \left(\frac{MRTR}{TotalCapacity} \right) * 100$$

Donde *MRTR* (Minimum Reserved Traffic Rate) es la Tasa de Tráfico Mínima Reservada, en símbolos, de la conexión actual y *TotalCapacity* es la capacidad total en símbolos del frame. Para obtener los valores anteriores en términos de símbolos, una modulación QPSK es considerada.

4.4.2.2. Descripción de la función

Cuando la función *wimax_bs_control_sched_bw_req_dequeue ()* es invocada, lo primero que ésta realiza es verificar que en todo el sistema calendarizador haya al menos una solicitud.

Debido a la presencia de PQ en el modo estricto, primero se ingresa al grupo PS verificando que el número de solicitudes pendientes en el grupo sea mayor a uno. Si esta condición se cumple, la función *wimax_bs_control_sched_ps_bw_req_dequeue ()* es invocada. Esta función tiene el objetivo de entregar una BWR de PS. Para lograr lo anterior, la función *wimax_bs_control_sched_mdrq_select ()* es necesaria. Esta última desencola una BWR de PS usando el algoritmo MDRR, descrito en la sección 4.4.2.1.3. Si por alguna razón la cola seleccionada por el algoritmo se encuentra bloqueada, la función *wimax_bs_control_sched_ps_bw_req_dequeue ()* invocará de nuevo a la función *wimax_bs_control_sched_mdrq_select ()* hasta que la cola seleccionada sea válida.

Si no existen solicitudes pendientes en PS, se verifica que el número de solicitudes pendientes en BE sea mayor uno. Si esta condición se cumple, la función *wimax_bs_control_sched_be_bw_req_dequeue ()* es invocada. Esta función tiene el objetivo de entregar una BWR de BE. Para lograr lo anterior, la función *wimax_bs_control_sched_rr_q_select ()* es necesaria. Esta última desencola una BWR de BE usando el algoritmo RR, descrito en la sección 4.4.2.1.2.

MRTR [Kbps]	Q_i	Q_i truncado	w_i
10	0.085	0	1500
56	0.481	0	1500
96	0.825	0	1500.
140	1.203	1	2012.

Tabla 4.1: Ejemplo del cálculo del *peso* en MDRR

4.4.3. Particularidades y Limitaciones del Sistema Calendarizador de OPNET

Como ya se mencionó con anterioridad, el sistema calendarizador de una red Wimax no está definido en ningún estándar. El estándar IEEE 802.16 no impone ningún algoritmo calendarizador para la asignación de los recursos de una red Wimax. Lo anterior, permite que la investigación y los fabricantes propongan diferentes sistemas calendarizadores para redes Wimax que ofrezcan diversos beneficios y esten enfocados a diversos usos.

El sistema calendarizador de OPNET ofrece la mayoría de las características que un sistema de este tipo debe poseer para ser considerado como bueno. El calendarizador de OPNET esta basado en el algoritmo RR y versiones y modificaciones de éste. RR es considerado como uno de los algoritmos de calendarización más usados debido a su baja complejidad y a su alta equidad entre las conexiones a las que da servicio. La codificación eficiente, utilizada durante la programación del calendarizador, permite que el número líneas utilizadas por el código sea reducido y la comprensión sea relativamente fácil.

Aunque el sistema calendarizador de OPNET satisface las características que se requieren para proporcionar QoS, éste presenta algunas limitaciones que de ser resueltas pueden ofrecer mejores resultados.. A continuación se discuten los inconvenientes encontrados durante el análisis del calendarizador de OPNET.

4.4.3.1. Cálculo del *peso* en MDRR

En el algoritmo MDRR de OPNET Modeler, existe un valor denominado *peso* que es emplea para incrementar el *contador de déficit*. El *peso* es calculado mediante la ecuación 4.1. Considerando un frame de 5[ms] y 11,635[Mbps] de capacidad, se encuentra que w_i es el mismo para conexiones con MRTR inferiores a 110[kbps], provocando que todos los paquetes, sin importar si son datos o voz, reciban la misma prioridad. Q_i es el cociente entero de la división del MRTR y la capacidad total del canal, cuando este valor es menor al 1, el valor tomado es 0, lo que implica que w_i tendrá el mismo valor siempre que este caso se presente. La tabla 4.1 ejemplifica este caso, donde $MTU = 1500[Simbolos]$. La unidad de MTU es simbolos en este caso debido a que el calendarizador otorga recursos, decrece y aumenta el *contador de déficit* en simbolos.

4.4.3.2. rtPS y nrtPS con igual prioridad

Aunado al inconveniente encontrado en el cálculo del *peso*, MDRR presenta otra limitación: rtPS y nrtPS son tratados con la misma prioridad. En la sección 4.4.2.1.3 se mencionó que dentro del grupo PS se encuentran las conexiones rtPS y nrtPS activas, a las cuales se les asigna un peso calculado por la ecuación 4.1. Debido a que para ambos tipos de calendarización se utiliza el mismo procedimiento para el cálculo de este parámetro, dos conexiones, una rtPS y la otra nrtPS, con el mismo MRTR, recibirán el mismo *peso*. Lo anterior no es correcto ya que se omite una de las necesidades primordiales de rtPS: La consideración el retardo. rtPS es un servicio que debe garantizar el retardo ya que debe soportar flujos de datos en tiempo real, por lo que la *Latencia Máxima* debe ser un parámetro imprescindible en el cálculo del parámetro que representa la prioridad de la conexión.

Cuando se dice que rtPS y nrtPS reciben el mismo peso para sus respectivas conexiones, quiere decir que en la práctica, se tiene que una conexión FTP y una MPEG (Moving Picture Experts Group) poseen la misma prioridad en la transmisión, sin considerar que la transmisión MPEG es de tiempo real. Bajo este escenario, la conexión FTP no sufrirá daño alguno. Sin embargo, en la conexión MPEG, se corre el riesgo de que los paquetes no sean recibidos en el momento adecuado.

4.4.3.3. Sistema Calendarizador GPSS Inexistente en la SS

En la sección 3.4.3.3 se indicó que en el modo de asignación GPSS, la SS debe tener la capacidad de administrar los otorgamientos recibidos de la BS. De este modo, si la SS en determinado momento tiene un otorgamiento para la conexión x pero la conexión y necesita enviar lo más pronto posible un paquete, deliberadamente podrá reasignar al otorgamiento de la conexión x a la conexión y .

La implementación en OPNET no cubre esta especificación del estándar IEEE 802.16, es decir, bajo cualquier circunstancia, la SS utilizará los otorgamientos tal y como fueron asignados por la BS.

Diseño y Simulación de una Técnica de Calendarización

El sistema calendarizador, descrito y analizado en el capítulo 4, presenta un algoritmo que ofrece una buena QoS. Sin embargo, dicho sistema también presenta algunas limitaciones que fueron comentadas en el mismo capítulo.

En este capítulo, se presenta el diseño de una técnica de calendarización con el fin de subsanar las deficiencias encontradas en el sistema calendarizador propuesto por OPNET Modeler. El nuevo diseño es incluido en la implementación de OPNET y simulado, obteniéndose los resultados también mostrados en este capítulo.

5.1. Diseño de un algoritmo Calendarizador GPSS en la SS

La posibilidad de que una SS tenga la capacidad de administrar los recursos que la BS le otorgó, es una herramienta que puede resultar muy útil en caso de congestión de la red. De este modo, si una conexión con baja tolerancia a los retardos necesita enviar un paquete de manera urgente debido a que ésta ya espera mucho tiempo en su respectiva cola, la misma SS puede darle prioridad a ese paquete reasignando los otorgamientos que le fueron concedidos. Sin embargo, OPNET Modeler no tiene implementada esa función en sus modelos de procesos. El modelo de la SS, en OPNET, respeta la asignación hecha por la BS y utiliza los recursos otorgados en la conexión que generó la solicitud.

De hecho, una búsqueda de algoritmos de calendarización propuestos para SS en las principales fuentes de información científica, arroja pocos resultados al respecto. Uno de los escasos casos de diseños de calendarizadores GPSS se presenta en [23].

Debido a lo anterior, en este trabajo se propone diseñar un algoritmo calendarizador en la SS que proporcione prioridad a las conexiones rtPS sobre las conexiones nrtPS y BE, de esta forma se tendrá un calendarizador GPSS, que descomponga los otorgamientos proporcionados por la BS como mejor le convenga a la SS.

5.1.1. Introducción del concepto *Satisfacción del retardo*

En la sección 4.4.3.2, se menciona que la *Latencia Máxima* no es considerada dentro del algoritmo MDRR, lo que implica que rtPS y nrtPS tengan la misma prioridad. El presente trabajo, toma en cuenta este importante parámetro y lo utiliza en el cálculo de la *satisfacción del retardo*. La satisfacción del retardo, es un indicador que permite saber si una cola rtPS tiene urgencia por enviar un paquete y es definida como

$$S_i = L_{Max} - T_{guarda} - T_{espera} + 1 \quad (5.1)$$

donde L_{Max} es la *Latencia Máxima* permitida para la conexión rtPS actual, T_{guarda} denota el tiempo de guarda antes de la latencia máxima y T_{espera} es el tiempo de espera del paquete que está en la cabeza de la cola. Esta ecuación es una modificación de la ecuación del *indicador de satisfacción* presentado en [6]. Ahí dicha ecuación es usada para controlar la función de prioridad propuesta en ese trabajo.

- Si $S_i \geq 1$, los requerimientos de retardo de la conexión actual se satisfacen. Valores grandes de $S_i \geq 1$ denotan alta satisfacción del retardo.
- Si $S_i < 1$, los requerimientos de retardo no se satisfacen y el paquete al inicio de la cola debe ser enviado inmediatamente para evitar que la capa aplicación lo deseche debido a la gran cantidad de tiempo que éste ha pasado en la cola. Una cola que no satisface los requerimientos de retardo es marcada como cola con urgencia.

5.1.2. Descripción del Algoritmo

El objetivo del algoritmo calendarizador GPSS propuesto en este trabajo es dar mayor prioridad a las conexiones rtPS sobre las conexiones nrtPS. La SS, aprovechando su capacidad para administrar los otorgamientos proporcionados por la BS, asigna a conexiones rtPS otorgamientos que originalmente eran para conexiones nrtPS en caso de que una cola rtPS se encuentre en estado de *urgencia*. Una cola es declarada como cola con urgencia cuando ésta no satisface los requerimientos de retardos indicados en la sección 5.1.1 y en la lista de otorgamientos no existe ninguno para ella.

A continuación se enumeran las acciones realizadas por el calendarizador propuesto:

1. Obtener el número de colas activas en la SS.
2. Visitar cada una de las colas. Si la cola no está vacía y pertenece a una conexión rtPS:
 - a) Calcular la satisfacción del retardo, S_i . Si $S_i < 1$:
 - 1) Considerar a la cola actual como cola con urgencia.
3. Obtener la lista de otorgamientos (IE) proporcionados por la BS.

4. Procesar cada otorgamiento de forma separada. Si el otorgamiento actual es para una conexión rtPS y su respectiva cola está marcada como cola con urgencia:
 - a) Eliminar la urgencia a la cola actual.
5. Si el otorgamiento pertenece a una conexión nrtPS y el número de colas con urgencia es mayor a 0:
 - a) Encontrar una cola con urgencia y obtener su CID.
 - b) Asignar a la cola con urgencia los recursos otorgados originalmente a la cola nrtPS.
 - c) Actualizar los campos del IE para que apunte cola con urgencia.
 - d) Eliminar la urgencia de la cola.

El diagrama de flujo correspondiente al calendarizador GPSS propuesto se muestra en la figura 5.1.

5.1.3. Tiempo entre polls en conexiones rtPS

Como parte del nuevo diseño de calendarizador GPSS, se identificó que el tiempo entre *polls* dirigidos a una SS es un factor importante en el retardo de encolamiento de un paquete en capa MAC. Como los *polls* no son solicitados por la SS sino otorgados periódicamente por la BS, es necesario e indispensable calcular un tiempo entre *polls* adecuado. Una tasa de *polls* elevada podría ser muy agresiva y resultar en demasiados encabezados y la posibilidad de denegar otras solicitudes más importantes. Por otro lado, un tasa de *polls* muy baja sería un factor limitante en la capacidad de una conexión para alcanzar las tasas configuradas para ésta.

5.1.3.1. Tiempo entre polls en OPNET

Para realizar el calculo de tiempo entre *polls*, OPNET utiliza un concepto denominado *relación polls-paquetes* que indica la relación entre la tasa de polls no solicitados y la tasa estimada de una conexión dada, la *relación polls-paquetes* se expresa en polls/paquetes. Con el fin de evitar las limitaciones comentadas en la sección anterior, OPNET Modeler establece que la relación mencionada no puede ser ni menor a 1 ni mayor a 25.

El *Tiempo entre polls* en OPNET es definido mediante la ecuación 5.2.

$$T_{entrepolls} = \left[\frac{m_{rate}}{SDU_{promedio}} R_{polls-paquetes} \right]^{-1} \quad (5.2)$$

donde $T_{entrepolls}$ es el tiempo entre polls; m_{rate} es la tasa de datos, en [bps], utilizada para el cálculo, MRTR para nrtPS y MSTR (Maximum Sustained Traffic Rate) para rtPS; $SDU_{promedio}$

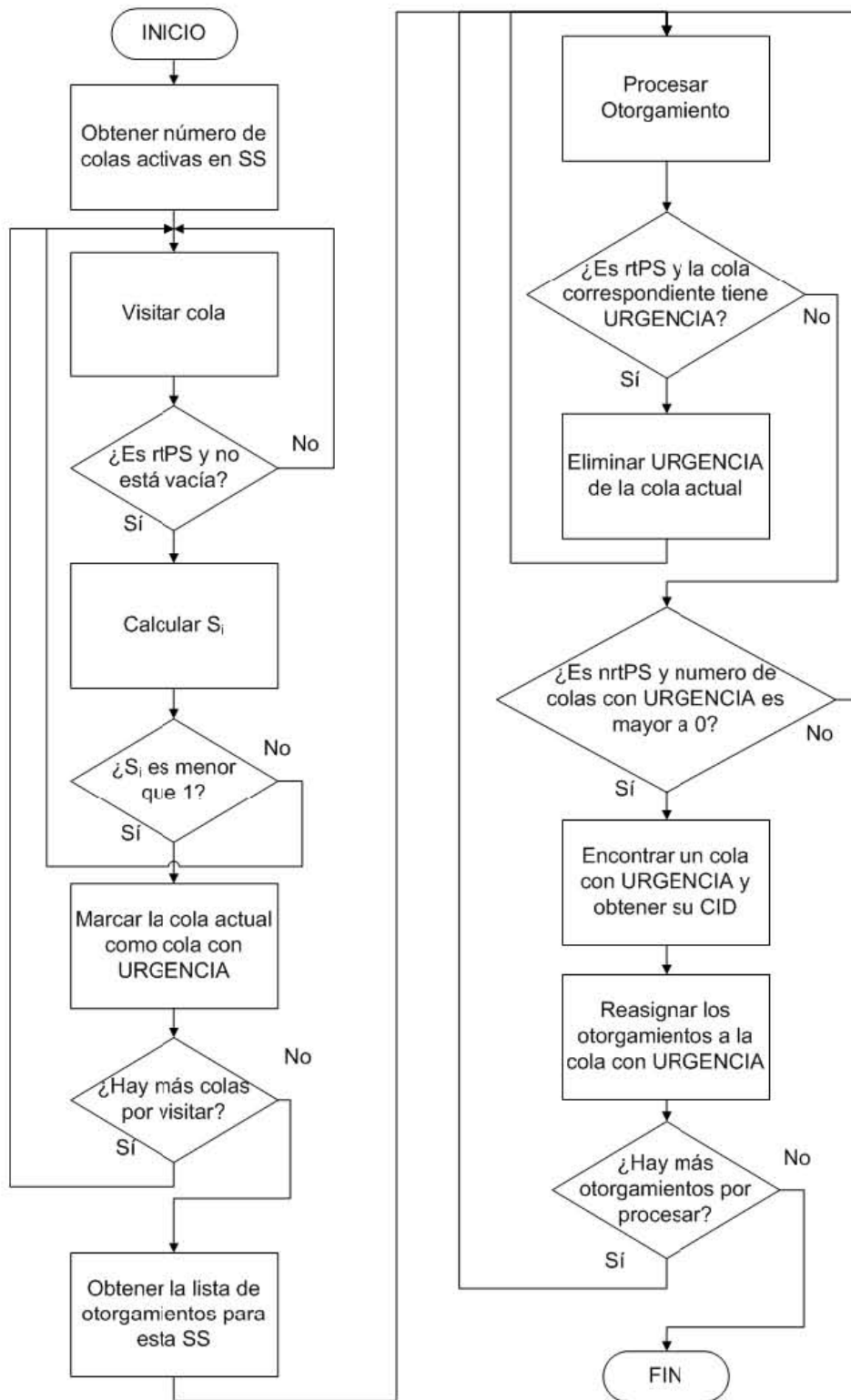


Figura 5.1: Diagrama de Flujo del calendarizador GPSS propuesto

R_i	$T_{entrepolls}$ [ms]	$T_{entrepolls} < L_{Max}$
1	125.00	No
2	62.50	No
3	41.60	No
4	31.25	No
5	25.00	Sí

Tabla 5.1: Ejemplo del nuevo cálculo del *tiempo entre polls* para una conexión rtPS con $MSTR = 96[Kbps]$, $SDU_{promedio} = 12000[bits]$ y $L_{Max} = 30[ms]$

se refiere al tamaño del paquete SDU en bits y $R_{polls-paquetes}$ es la relación polls-paquetes que de forma predeterminada es igual a 1.

La ecuación 5.3 muestra el ejemplo del cálculo de tiempo entre polls para una conexión rtPS con $MSTR = 96[Kbps]$ y $SDU_{promedio} = 12000[bits]$.

$$T_{entrepolls} = \left[\frac{96000 \frac{bits}{seg}}{12000 \frac{bits}{paqueteSDU}} \left(1 \frac{poll}{paqueteSDU} \right) \right]^{-1} = 125[ms] \quad (5.3)$$

Si la conexión en cuestión fuera una aplicación de VoIP, donde el retardo no debe rebasar los $50[ms]$, el *tiempo entre polls* impediría que los paquetes en dirección UL fueran entregados a tiempo. De este modo, para cuando un poll periódico llegara a la SS, el retardo de encolamiento del paquete en la cabeza de la cola correspondiente muy probablemente sería mayor a 50 ms.

5.1.3.2. Nuevo cálculo para el *tiempo entre polls*

Debido a la limitación del cálculo del *tiempo entre polls* propuesto por OPNET, a continuación se presenta un nuevo cálculo para dicho parámetro que considera la *Latencia Máxima*.

Con el fin de privilegiar a las conexiones rtPS sobre las conexiones nrtPS y obedecer las especificaciones de estándar IEEE 802.16, se decidió mantener intacta la ecuación 5.2 para conexiones nrtPS y usar la ecuación 5.4 para conexiones rtPS, donde la *Latencia Máxima* toma importancia.

$$T_{entrepolls} = \left[\frac{m_{rate}}{SDU_{promedio}} R_i \right]^{-1} \quad (5.4)$$

donde $R_i \in \{1, 2 \dots 25\}$ es la relación polls-paquetes. El valor adecuado de R_i será el mínimo valor de éste que haga que se cumpla que $T_{entrepolls} < L_{Max}$. La tabla 5.1 muestra un ejemplo del nuevo cálculo para una conexión rtPS con $MSTR = 96[Kbps]$, $SDU_{promedio} = 12000[bits]$ y $L_{Max} = 30[ms]$. En este caso, se asegura que el *tiempo entre polls* no afecte de forma negativa el retardo de encolamiento de los paquetes en la capa MAC.

5.2. Escenario de Simulación

Con el fin de evaluar y verificar el funcionamiento y la eficiencia del diseño de calendarizador GPSS desarrollado en este trabajo, se implementó un escenario de simulación. Dicho escenario consta de hasta 110 usuarios móviles desplazándose en un celda con sólo una BS. Existen también tres servidores: Web Server, FTP Server y Voice Server como se muestra en la figura 5.2

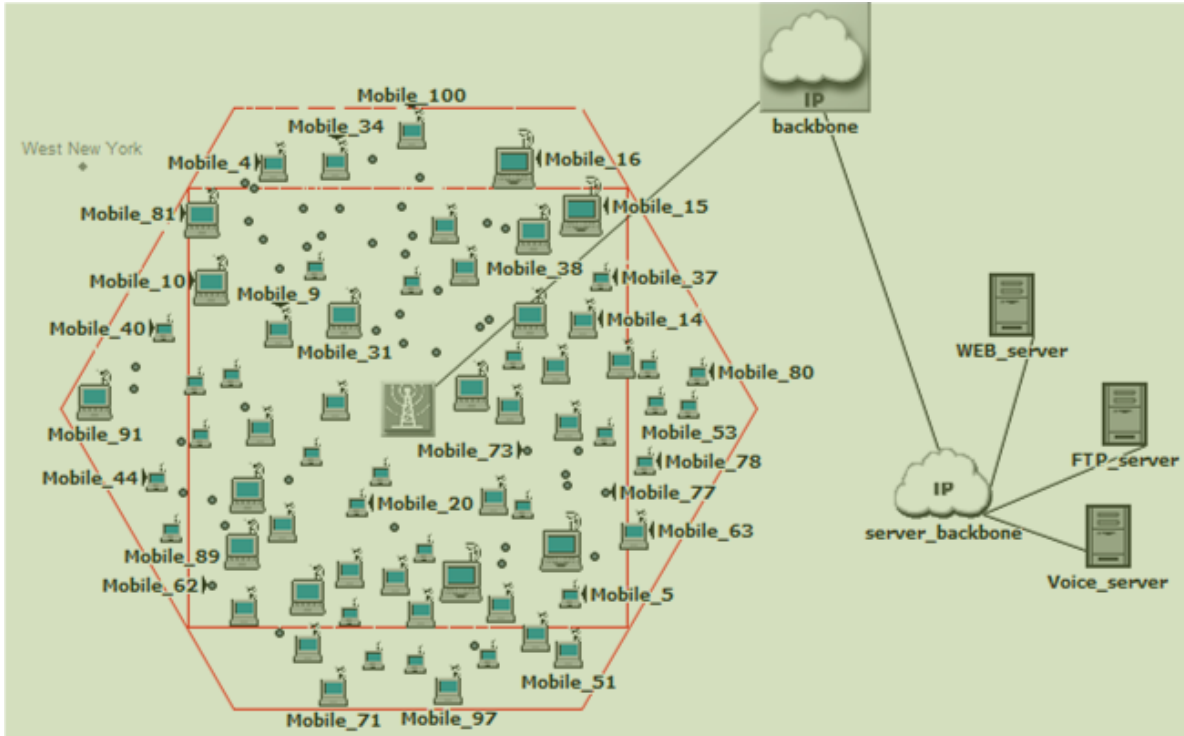


Figura 5.2: Escenario de Simulación

La configuración de red y los parámetros Wimax seleccionados se muestran en la tabla 5.2. Se utilizó y simuló un enlace OFDMA basado en TDD con 2048 subportadoras en la banda de 5[GHz]. El límite UL/DL es fijo y se estableció en 50 %.

Debido a que las SSs empleadas durante la simulación necesitan tener movilidad y con el fin de obtener resultados lo más apegados posibles a la realidad, a cada SS se le configuró un movimiento aleatorio, tal y como se indica en la tabla 5.3.

Este escenario de simulación tuvo dos versiones debido al tipo de aplicaciones usadas en las SSs; las características de cada una de estas versiones se explican en las secciones siguientes.

5.2.1. Escenario A: Aplicaciones diversas en cada SS

En este escenario, cada SS genera y recibe tráfico de 3 aplicaciones diferentes: VoIP, FTP y Web. La tabla 5.4 muestra los parámetros utilizados en cada una de estas aplicaciones.

Parámetro	Valor
Capa Física	OFDMA-TDD
Ancho de Banda del Sistema [MHz]	20
No. de Subcanales	UL: 70 DL: 60
No. de Subportadoras de Datos	UL: 1120 DL: 1440
Límite UL/DL	50 %
Duración de Frame [ms]	5
Duración de Símbolo [μ s]	100.8
ARQ y HARQ	Deshabilitado

Tabla 5.2: Parámetros de Red

Parámetro	Valor
Distancia en x [m]	1267.95
Distancia en y [m]	1267.95
Velocidad [m/s]	5
Tiempo de Inicio [s]	10
Tiempo de Fin [s]	Fin de la Simulación

Tabla 5.3: Configuración de la Movilidad en la SS

Aplicación	Parámetro	Valor
VoIP (G.711)	Tipo de Servicio	Interactive Voice (6)
	Tasa de Datos	64 [Kbps]
	Tamaño del paquete	80 [Bytes]
	Tiempo entre arribos	10 [ms]
FTP	Tipo de Servicio	Excellent Effort (3)
	Tasa de Datos	40 [Kbps]
	Tamaño del archivo	10 000 [Bytes]*
	Tiempo entre arribos	2 [s]*
	Relación Get/Push	50 %
FTP	Tipo de Servicio	Best effort (0)
	Tasa de Datos	5 [Kbps]
	Tamaño de la página	1 875 [Bytes]*
	Tiempo entre arribos	3 s*

* Distribución Exponencial

Tabla 5.4: Parámetros para Aplicaciones en el Escenario de Simulación A

Tipo de Servicio	Nombre	Tipo de Calendarización	MSTR [Kbps]	MRTR [Kbps]	Latencia Máxima [ms]
Interactive Voice(6)	Gold	rtPS	104	104	30
Excellent Effort(3)	Silver	nrtPS	52	52	N/A
Best Effort(3)	Bronze	BE	26	26	N/A

Tabla 5.5: Clases de Servicio en el Escenario de Simulación A

Tipo de Servicio	Clase de Servicio	Modulación y Codificación	SDU Promedio [bytes]	Buffer [bytes]	UL/DL
Interactive Voice(6)	Gold	Adaptiva	1500	64000	DL
Excellent Effort(3)	Silver	Adaptiva	1500	64000	DL
Best Effort(3)	Bronze	Adaptiva	1500	64000	DL
Interactive Voice(6)	Gold	Adaptiva	1500	64000	UL
Excellent Effort(3)	Silver	Adaptiva	1500	64000	UL
Best Effort(3)	Bronze	Adaptiva	1500	64000	UL

Tabla 5.6: Flujos de Servicio en los escenarios de simulación A y B

Los tipos de servicios y aplicaciones de la tabla 5.4 corresponden a una de las *Clases de Servicio* y *Flujos de Servicio* configuradas en la capa Wimax MAC. La tablas 5.5 y 5.6 muestra dicha configuración.

5.2.2. Escenario B: VoIP en BE, nrtPS y rtPS

En este escenario, cada SS genera y recibe tráfico de 3 aplicaciones de VoIP de forma simultánea. Cada una de las aplicaciones en la SS pertenece a un tipo de servicio diferente y, por lo tanto, a un tipo de calendarización diferente. La tabla 5.7 muestra los parámetros utilizados en la aplicación utilizada en este escenario.

Las *clases de servicio* utilizadas en el escenario A sufren una modificación para poder ser empleadas en el escenario B, las *clases de servicio* del escenario B se muestran en la tabla 5.8. Los *Flujos de Servicio* es común para ambos escenarios, véase tabla 5.6.

5.3. Resultados

Las simulaciones de los escenarios A y B fueron realizadas de acuerdo a las configuraciones mostradas en la sección anterior. Aunado a las simulación de cada escenario, se realizó una simulación previa con el fin de caracterizar la red con la cual se trabajó y de conocer sus puntos de saturación.

Aplicación	Parámetro	Valor
VoIP (G.711)	Tipo de Servicio	Interactive Voice (6)
	Tasa de Datos	64 [Kbps]
	Tamaño del paquete	80 [Bytes]
	Tiempo entre arribos	10 [ms]
VoIP (G.711)	Tipo de Servicio	Excellent Effort (3)
	Tasa de Datos	64 [Kbps]
	Tamaño del paquete	80 [Bytes]
	Tiempo entre arribos	10 [ms]
VoIP (G.711)	Tipo de Servicio	Best Effort(0)
	Tasa de Datos	64 [Kbps]
	Tamaño del paquete	80 [Bytes]
	Tiempo entre arribos	10 [ms]

Tabla 5.7: Parámetros para Aplicación en el escenario de simulación B

Tipo de Servicio	Nombre	Tipo de Calendarización	MSTR [Kbps]	MRTR [Kbps]	Latencia Máxima [ms]
Interactive Voice(6)	Gold	rtPS	104	104	30
Excellent Effort(3)	Silver	nrtPS	104	104	N/A
Best Effort(3)	Bronze	BE	104	104	N/A

Tabla 5.8: Clases de Servicio en el escenario de simulación B

Número de Simulación	Tiempo de Simulación [min]	Número de Usuarios en la celda
1	6.5	10
2	6.5	30
3	6.5	35
4	6.5	40
5	6.5	50
6	6.5	70
7	6.5	90
8	6.5	110

Tabla 5.9: Número de usuarios y tiempo de simulación para la caracterización de la red

5.3.1. Caracterización de la Red

La caracterización de la red Wimax, utilizada en este trabajo, se obtuvo a partir de una serie de simulaciones con diferente número de SSs desplazándose en forma aleatoria. Cada simulación tuvo una duración de 6,5[*min*]. La tabla 5.9 muestra el número de usuarios utilizados en cada simulación.

Un factor importante de este trabajo es la movilidad de la SS. Para verificar que las SS realizaban un movimiento aleatorio se generó la gráfica de SNR de cada usuario. La figura 5.3 muestra el SNR de la SS No.4 en cada una de las simulaciones realizadas. El hecho de que cada curva sea diferente implica que, en cada simulación, la SS se movió de forma aleatoria.

5.3.1.1. Carga y Tráfico Enviado

Como se ha venido comentando a lo largo de este trabajo, el tráfico de datos puede darse en dos direcciones UL y DL. Para el caso UL y considerando como referencia a la capa Wimax MAC, la *Carga* representa el tráfico ingresado a Wimax MAC por las capas superiores. Por otra parte, el *Tráfico Enviado* se refiere al tráfico transmitido por la capa Wimax MAC y Física. Los encabezados de estas capas son considerados como tráfico enviado.

La figura 5.4 presenta la *carga* y el *tráfico enviado* totales en la red Wimax utilizada en las diferentes simulaciones. La diferencia entre ella se debe a que la primera es una estadística tomada al ingresar a la capa Wimax MAC, en la dirección UL, y por tanto no contabiliza los encabezados de PHY ni de la propia Wimax MAC. La gráfica muestra también que la red soporta un máximo de 40 usuarios con tráfico de las aplicaciones indicadas en la tabla 5.4. Con el fin de no colapsar la red, el mecanismo de control de admisión impide el acceso a más conexiones una vez que ésta ha llegado a su límite, lo anterior se puede constatar en la gráfica gracias al hecho de que las curvas dejan de crecer y se vuelven planas para valores mayores a 40 usuarios.

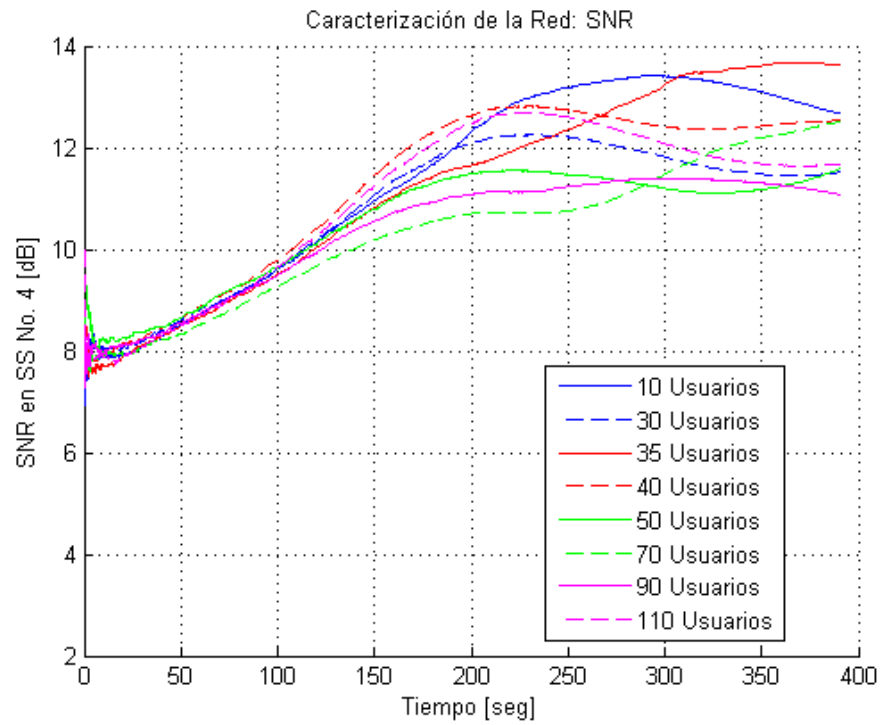


Figura 5.3: Gráfica de SNR en la SS No. 4

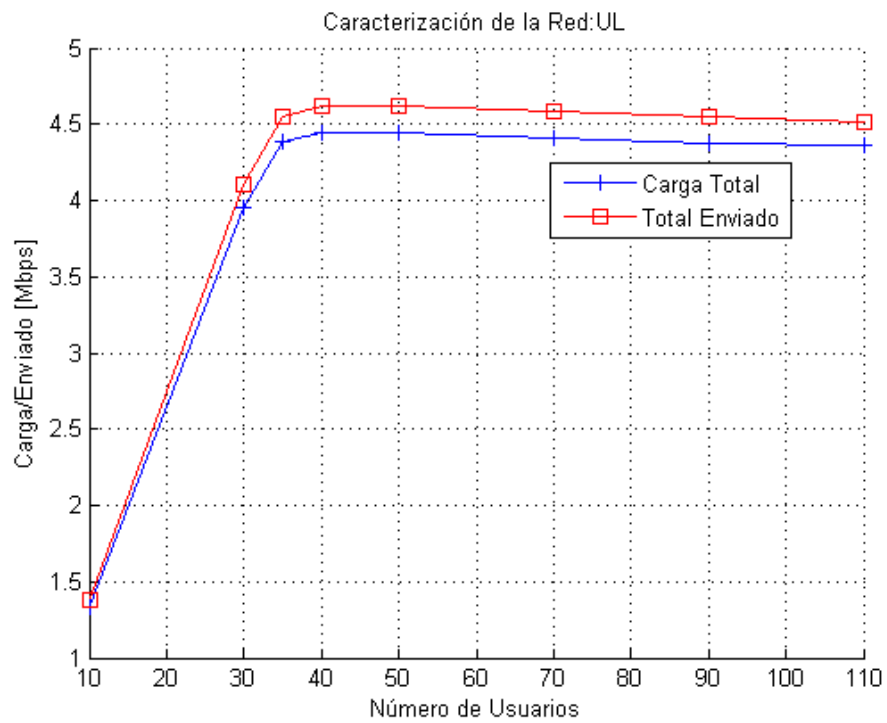


Figura 5.4: Carga y Tráfico enviado Totales en la red Wimax

5.3.1.2. Throughput y Tráfico Recibido

En el caso UL y tomando también como referencia a la capa Wimax MAC, el *Throughput* representa el tráfico de datos recibido satisfactoriamente por la capa Wimax MAC y remitido a las capas superiores. Similar al caso anterior, el *Tráfico Recibido* se refiere al tráfico recibido satisfactoriamente por Wimax MAC y proveniente de la capa física. Los encabezados de PHY y Wimax MAC son considerados tráfico recibido.

En la figura 5.5 se observa la gráfica de *throughput* y *tráfico recibido* totales en la red Wimax simulada. Esta figura, al igual que la figura 5.4, muestra que la red se ha saturado con 40 usuarios. Una vez superado este número, las curvas se vuelven planas debido al mecanismo de control de admisión.

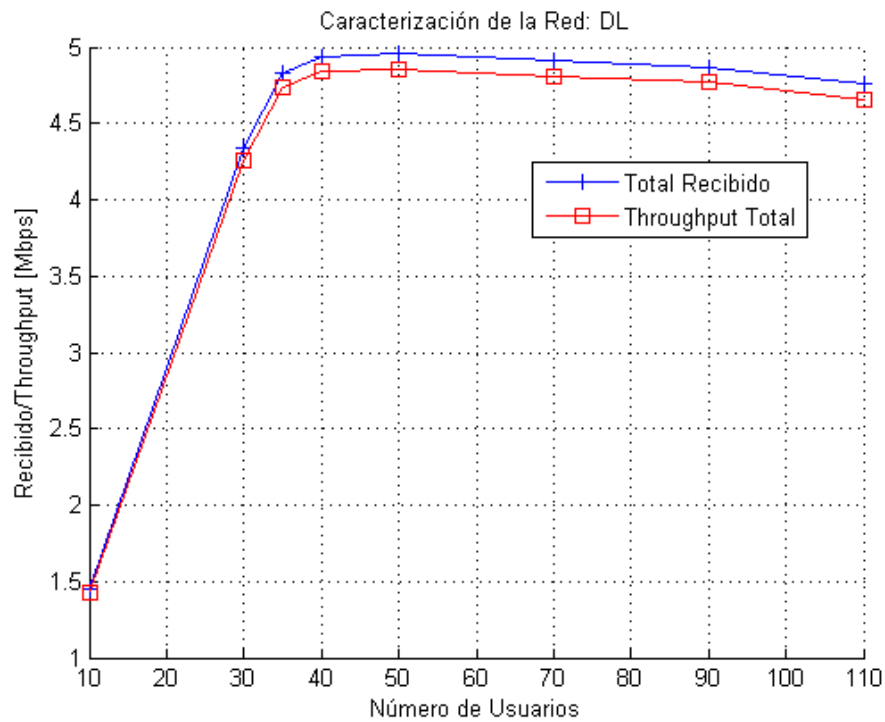


Figura 5.5: Tráfico recibido y Throughput Totales en la red Wimax

5.3.1.3. Tráfico Descartado

Debido a que en la dirección UL las colas de capa MAC tienen un límite, es posible descartar paquetes de datos en esta capa. Se considera *Tráfico Descartado* al tráfico que no fue aceptado de forma satisfactoria en la capa MAC debido a que la cola correspondiente se encontraba completamente llena. La figura 5.6 presenta el tráfico descartado durante la simulación de una red Wimax. La gráfica es consistente con las dos gráficas anteriores ya que se observa que la red empieza a descartar paquetes una vez que ésta ha superado los 40 usuarios, valor que ha sido considerado como punto de saturación de la red simulada.

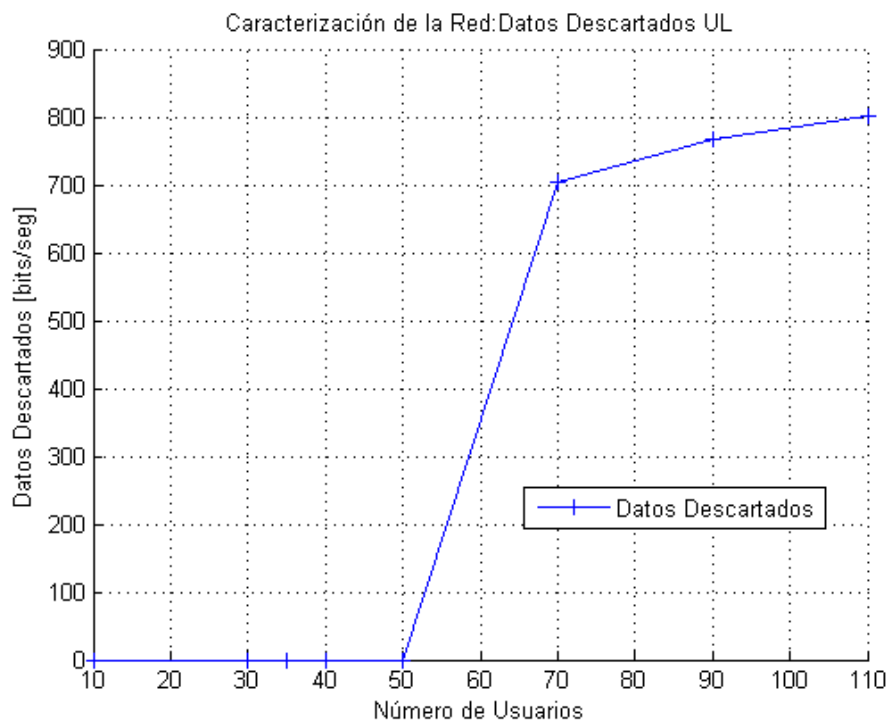


Figura 5.6: Tráfico total descartado en la red Wimax

5.3.1.4. Espacio Libre en el Frame

El espacio libre en el frame representa también un indicador del comportamiento de la red simulada. La gráfica 5.7 muestra el espacio libre en los subframes UL y DL durante las simulaciones realizadas. Cuando hay sólo 10 usuarios en la red, el espacio libre en los frame es bastante alto, 3,73 y 5,23[Mbps] para UL y DL respectivamente. Para 40 usuarios, el subframe UL ya no presenta espacio libre e indica que la red ha llegado a su límite. Aunque el límite UL/DL fue establecido en 50 %, UL y DL no tienen la misma capacidad debido a que cada uno tiene diferentes encabezados.

5.3.2. Escenario A

Como ya se mencionó anteriormente, en el escenario de simulación A, las SS generan y reciben tráfico de tres aplicaciones diferentes: WEB, FTP y VoIP, según la tabla 5.4. La implementación de calendarizador GPSS ofrece los resultados mostrados en las gráficas 5.8 - 5.13.

Las gráficas mencionadas muestran que gracias al calendarizador GPSS implementado, las conexiones rtPS son priorizadas por encima de nrtPS y BE, disminuyendo su retardo de encolamiento hasta 80 %, al pasar de 0.55 a 0.1 ms aproximadamente. Esta disminución representa un gran avance ya que un retardo de encolamiento mayor a 50 ms implica que el paquete pueda ser descartado por la aplicación correspondiente debido a que éste supera

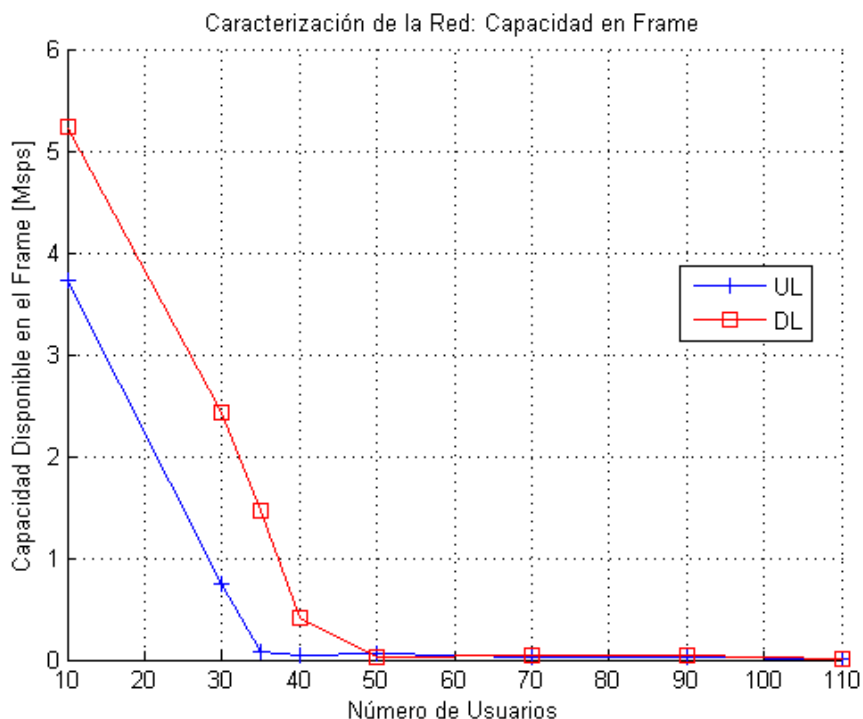


Figura 5.7: Espacio libre en el Frame

la latencia máxima soportada.

Las gráficas 5.8 - 5.13 también muestran que las conexiones nrtPS no son afectadas significativamente con la implementación del nuevo calendarizador GPSS, éstas mantienen un retardo más o menos constante a lo largo de las diferentes simulaciones. Las conexiones que sí se ven afectadas son las de BE, que incrementan su retardo de encolamiento a medida que el número de usuarios en la red aumenta. Si se considera que la clase de servicio *Bronze*, que contiene al tipo de calendarización BE, está diseñada para no ofrecer ningún tipo de garantía durante la transmisión de datos, entonces es posible concluir que las afectaciones en estas conexiones no son realmente importantes.

Cabe señalar que en las gráficas se observa que la conexión BE posee menor retardo de encolamiento que nrtPS, esto no implica que la prioridad de BE esté por encima de la de nrtPS. El retardo de encolamiento mínimo de BE se debe a la naturaleza de la aplicación utilizada (tabla 5.4). Por otra parte, la prioridad inferior de BE se puede observar en la vulnerabilidad de su retardo de encolamiento cuando el número de usuarios en la red incrementa.

La figura 5.14 confirma lo mostrado por las figuras 5.8 - 5.13. La nueva implementación ofrece una mejora importante en términos de retardo de encolamiento para las conexiones rtPS. La inclusión de la *Latencia Máxima* como parámetro importante tiene un efecto positivo en el tiempo que tarda un paquete de datos en la cola correspondiente.

La figura 5.14 permite observar que el retardo pasa de $53,65ms$ en la implementación actual de OPNET a $11,27ms$ en promedio en la nueva implementación del calendarizador GPSS.

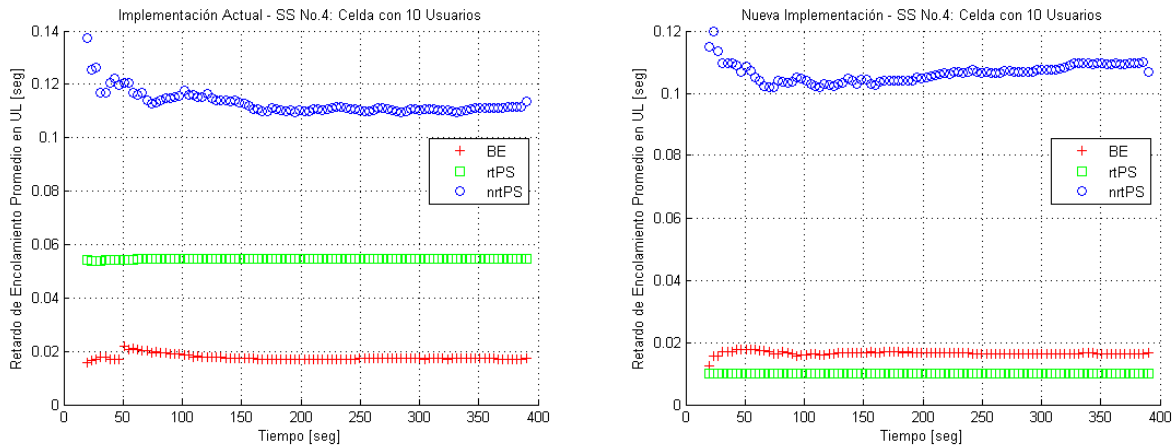


Figura 5.8: Comparación del retardo promedio de encolamiento en el UL: rtPS, nrtPS y BE con 10 usuarios en la celda

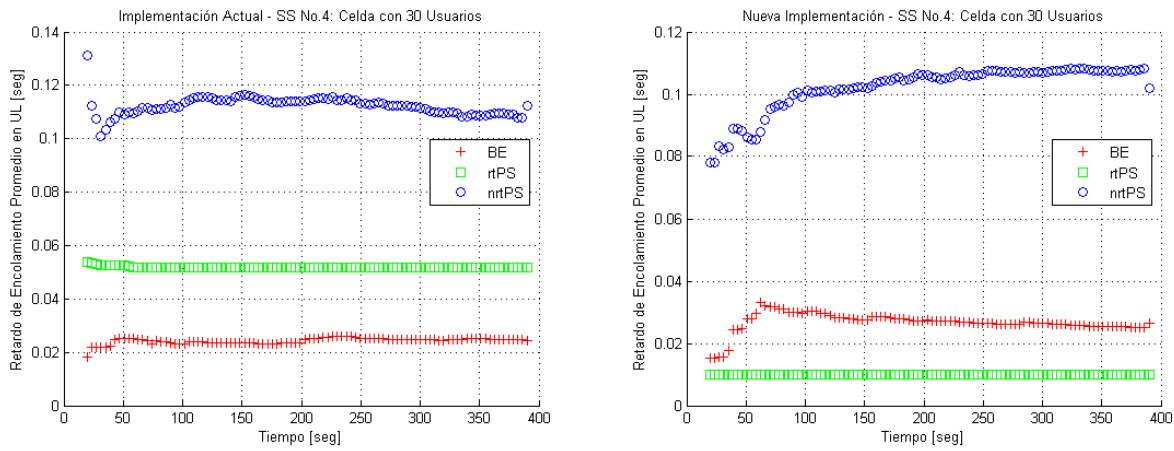


Figura 5.9: Comparación del retardo promedio de encolamiento en el UL: rtPS, nrtPS y BE con 30 usuarios en la celda

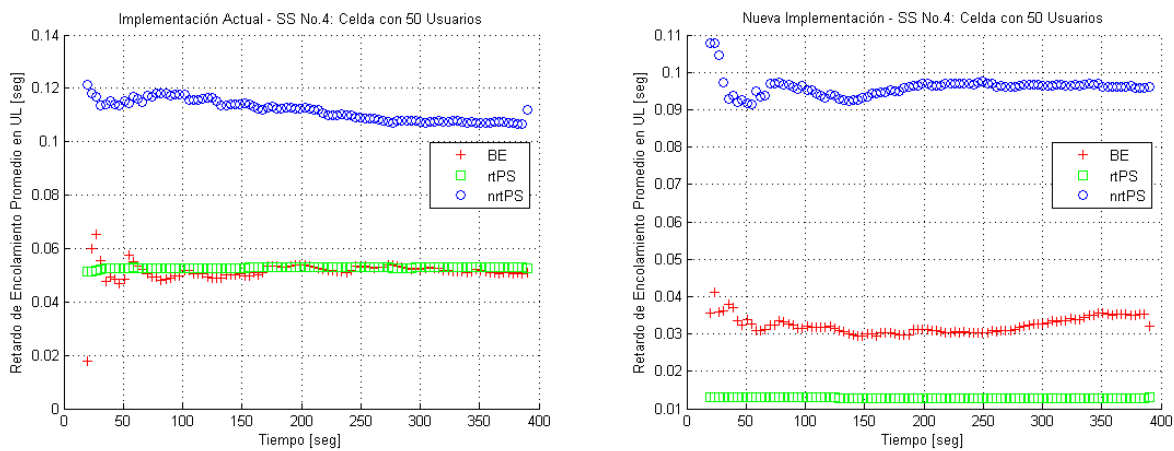


Figura 5.10: Comparación del retardo promedio de encolamiento en el UL: rtPS, nrtPS y BE con 50 usuarios en la celda

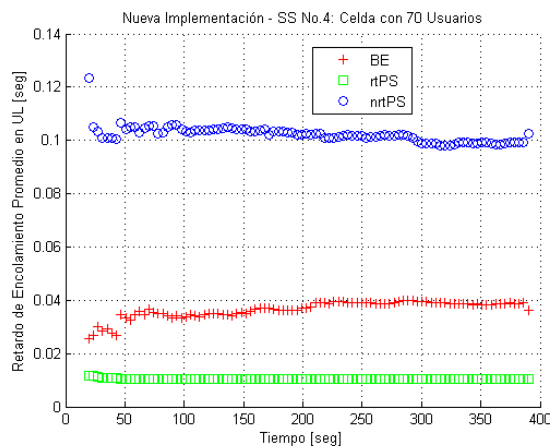
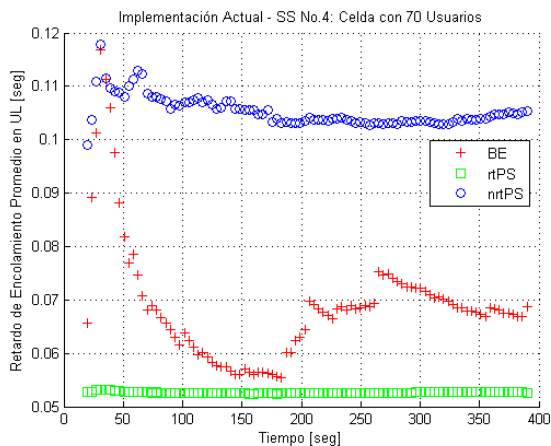


Figura 5.11: Comparación del retardo promedio de encolamiento en el UL: rtPS, nrtPS y BE con 70 usuarios en la celda

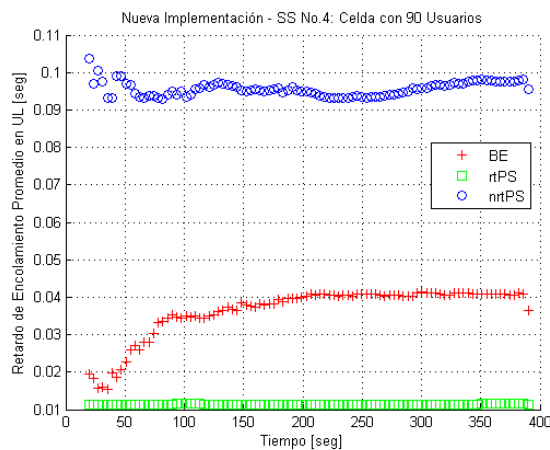
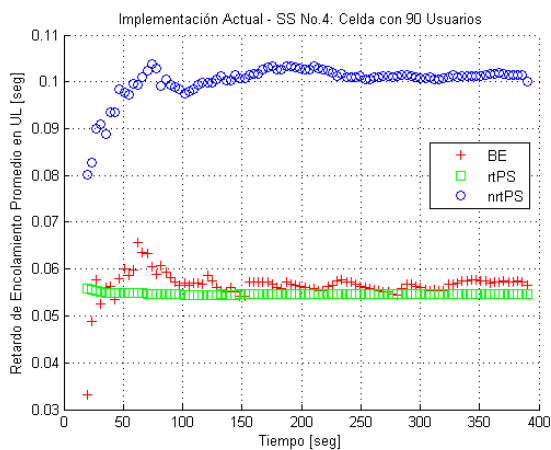


Figura 5.12: Comparación del retardo promedio de encolamiento en el UL: rtPS, nrtPS y BE con 90 usuarios en la celda

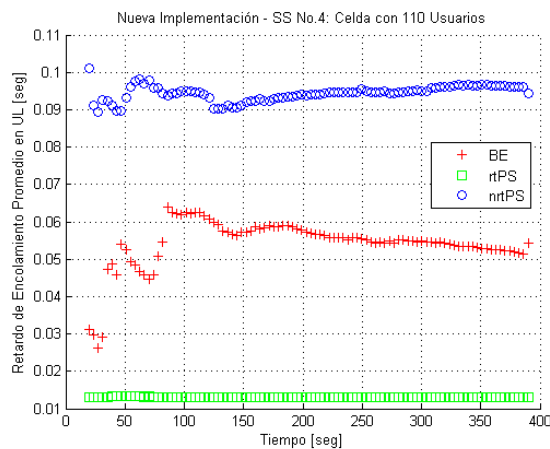
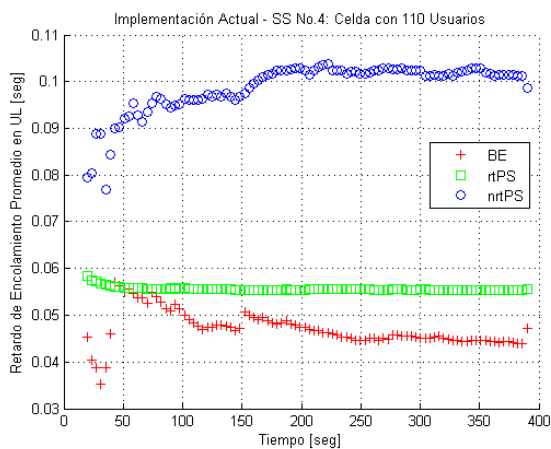


Figura 5.13: Comparación del retardo promedio de encolamiento en el UL: rtPS, nrtPS y BE con 110 usuarios en la celda

Lo anterior representa una disminución de aproximadamente 80 %, como ya se había mencionado anteriormente.

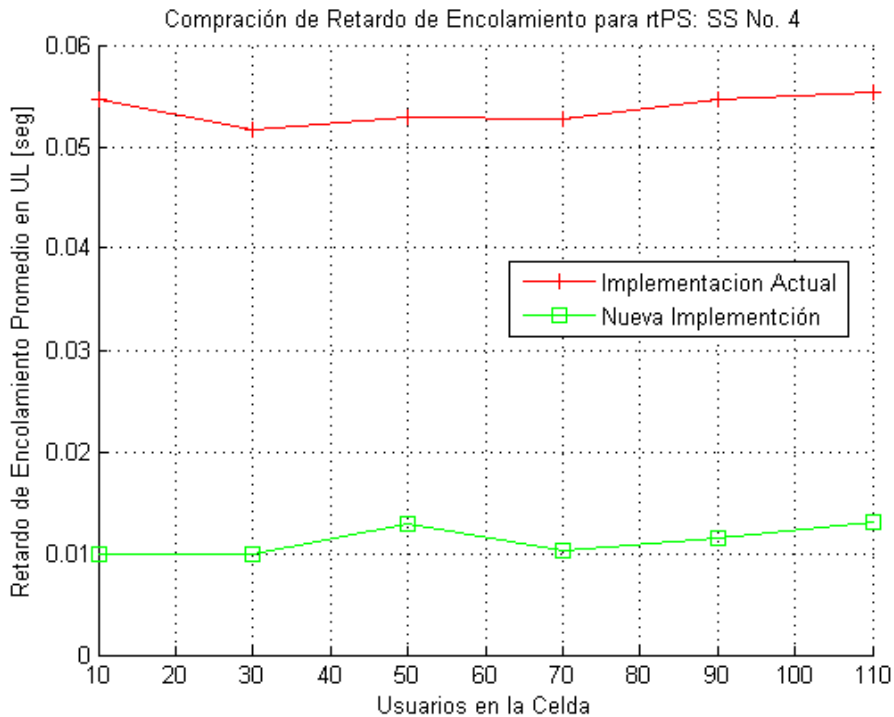


Figura 5.14: Comparación del retardo promedio de encolamiento en el UL para rtPS

5.3.3. Escenario B

En el escenario de simulación B, cada SS tiene tres conexiones activas de la misma aplicación pero diferente clase de servicio. Tres conexiones VoIP G.711 fueron configuradas para trabajar con BE, nrtPS y rtPS respectivamente. El objetivo de esta simulación fue observar el comportamiento de la aplicación en los diferentes tipos de calendarización y el comportamiento, en términos de prioridad, de éstos cuando se tiene un flujo de tráfico idéntico en cada uno de ellos.

Este escenario fue simulado en una sola ocasión durante 2,5[*min*] con 25 usuarios móviles y tres aplicaciones ejecutándose simultáneamente en cada uno de ellos. Lo anterior permitió que la simulación se realizara sin que la red llegara a la saturación y, por lo tanto, ninguna conexión fuera rechazada por el mecanismo de control de admisión.

5.3.3.1. Carga

La figura 5.15 presenta la carga, tráfico ingresado a la capa Wimax MAC en dirección UL, para conexiones rtPS, nrtPS y BE. Sólo se muestra la dirección UL porque es en ésta donde el calendarizador GPSS propuesto tiene efectos directos. En la gráfica se observa que las tres

conexiones inyectan la misma cantidad de tráfico a la capa Wimax MAC, 96 Kbps, lo cual es congruente ya que las tres conexiones contienen la misma aplicación.

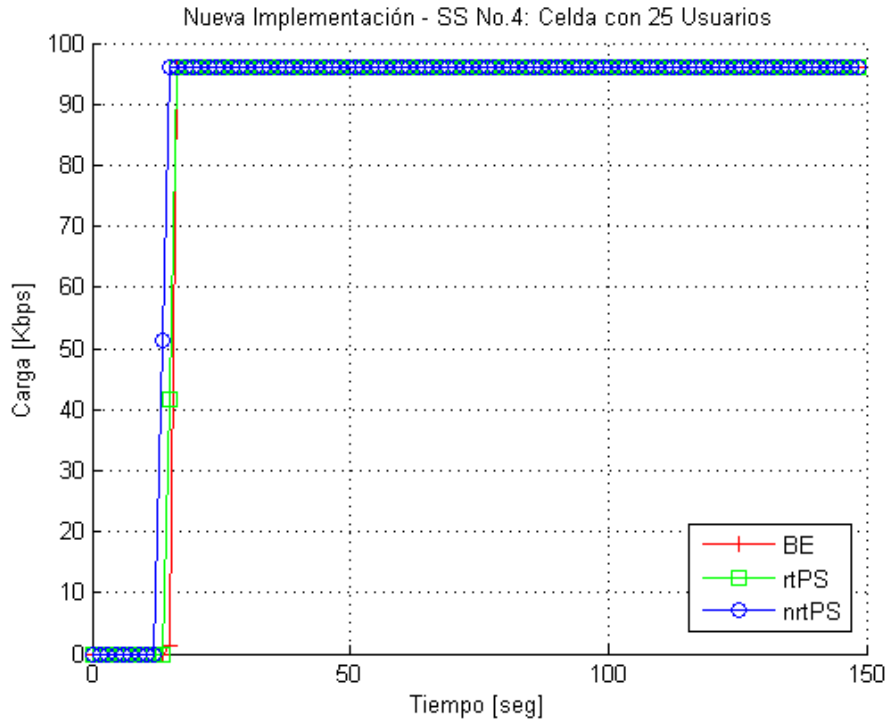


Figura 5.15: Tráfico ingresado a la capa MAC en el UL: VoIP utilizando rtPS, nrtPS y BE

5.3.3.2. Tráfico enviado y descartado

En la figura 5.16 se muestra el tráfico enviado y descartado por Wimax MAC en dirección UL. Es preciso recordar que el tráfico descartado representa el tráfico que es rechazado por la capa Wimax MAC debido que la cola de la conexión a la que pertenece dicho tráfico se encuentra llena. Cabe señalar que esta situación sólo se presenta en UL. El tráfico enviado es aquel que fue encolado en capa Wimax MAC y fue transmitido satisfactoriamente por la capa física.

La gráfica de *tráfico enviado* muestra que las conexiones rtPS y nrtPS lograron transmitir todo el tráfico inyectado por las capas superiores. Sin embargo, la conexión BE sólo logró transmitir entre 20 y 40 % del tráfico ingresado a Wimax MAC. El tráfico no enviado de la conexión BE fue descartado debido a la saturación de su cola, cuestión que se constata en la gráfica de *tráfico descartado*. Los resultados mostrados en la gráficas obedecen a que BE posee la prioridad más baja entre las conexiones activas, lo que implica oportunidades de transmisión mínimas.

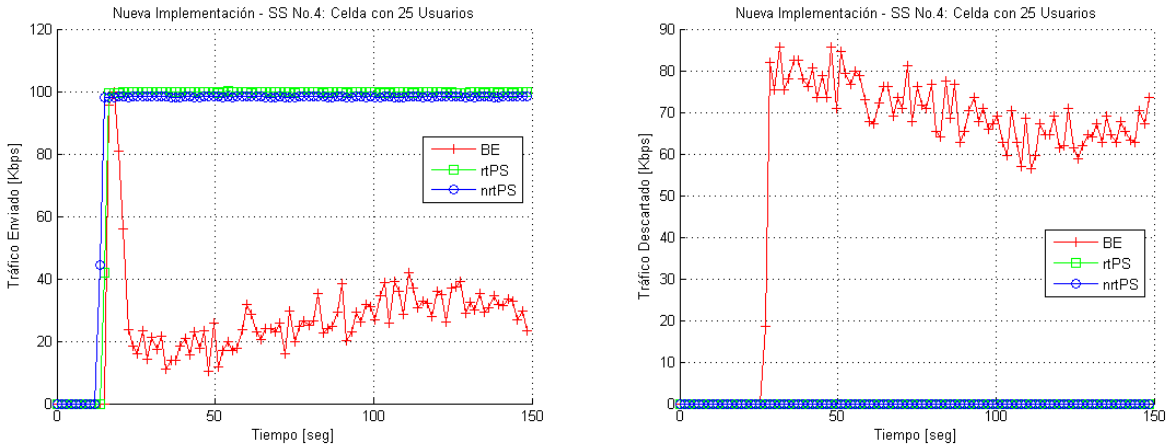


Figura 5.16: Tráfico enviado y descartado en el UL: VoIP utilizando rtPS, nrtPS y BE

5.3.3.3. Retardo de encolamiento

El retardo de encolamiento es el tiempo que tarda un paquete desde que entra a la cola correspondiente a su conexión hasta que sale de ella para ser transmitido por la capa física. La gráfica 5.17 muestra que a pesar de que las tres aplicaciones ejecutándose en la SS son idénticas, su comportamiento en Wimax MAC es diferente debido al tipo de calendarización asignado a cada una de ellas.

En la gráfica, nrtPS y rtPS presentan un retardo de encolamiento promedio de 54,30[ms] y 11,91[ms] respectivamente. Estos valores se aproximan a los obtenidos en la simulación del escenario A. rtPS mantiene una ventana en el retardo de encolamiento con respecto a nrtPS y BE debido a que es un tipo de calendarización en la que la *latencia máxima* es un parámetro importante a considerar. BE obtiene retardos de encolamiento que cambian con el tiempo y que llegan a ser hasta de 24[s], esto es congruente ya que BE no ofrece prácticamente ninguna garantía en el servicio y por lo tanto no es recomendable para aplicaciones de tiempo real como VoIP G.711.

5.3.4. Comparación Final

Esta sección presenta la comparación de estadísticas generales entre la implementación actual de OPNET y la nueva implementación propuesta en este trabajo. Asimismo, se describe el balance y el impacto de la implementación en la red Wimax simulada.

5.3.4.1. Retardo y Tráfico Descartado

El retardo total de una red Wimax representa el retardo end-to-end de todo los paquetes de datos que fueron recibido exitosamente en la capa Wimax MAC.

La figura 5.18 presenta resultados generales satisfactorios con respecto al retardo total de la red Wimax. El retardo total se redujo aproximadamente 88 % al pasar de 79,55[ms], promedio

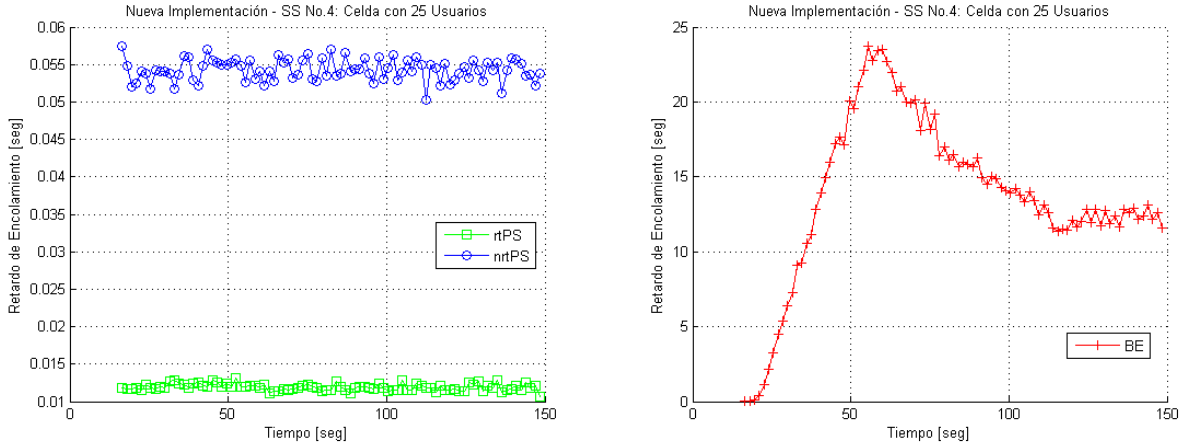


Figura 5.17: Retardo de encolamiento en el UL: VoIP utilizando rtPS, nrtPS y BE

de las simulaciones realizadas, a 36,89[ms].

El tráfico descartado, como en situaciones anteriores, representa el tráfico descartado por Wimax MAC debido a la saturación de las colas. La figura 5.18 muestra una disminución importante del tráfico descartado con la nueva implementación, este resultado se debe a que el retardo en general disminuyó, lo que representa que el retardo de encolamiento también lo hizo y los paquetes fueron servidos más rápidamente y se evitó descartar paquetes ya que las colas no se saturaron con mucha frecuencia.

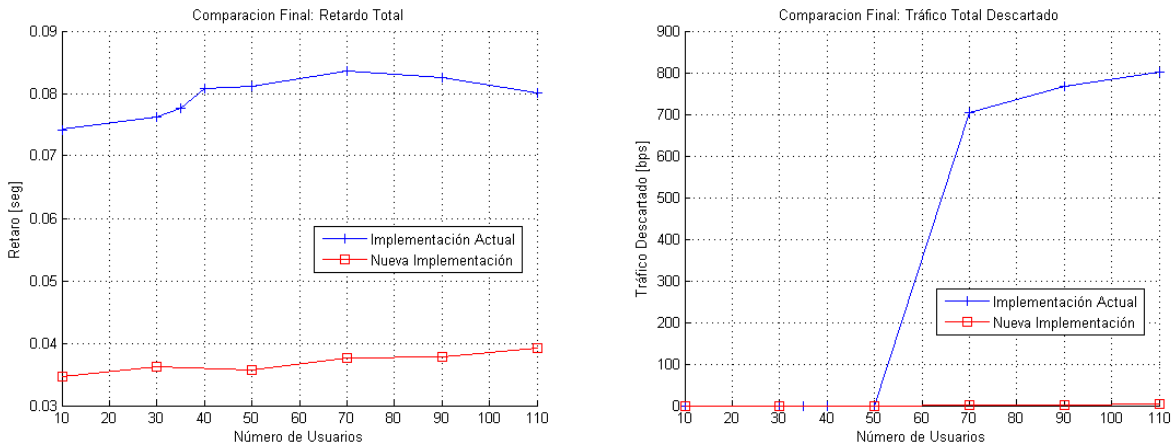


Figura 5.18: Comparación de retardo y tráfico descargado totales.

5.3.4.2. Carga y Tráfico Enviado

Los resultados en términos de *carga total* y *Tráfico total enviado* no fueron tan satisfactorios como en casos anteriores. La figura 5.19 muestra que existe una disminución en tráfico total enviado cuando se implementa el nuevo calendarizador GPSS. Lo anterior implica que la

red se saturó mas rápido y por lo tanto se pudo enviar menos información a través de la red. Esta disminución es de aproximadamente 12 %.

Si bien la reducción del tráfico total enviado provocado por la nueva implementación no es insignificante, el beneficio obtenido en el retardo de encolamiento de las conexiones rtPS y, por consiguiente, en el retardo total de la red Wimax es una muy buena compensación y un buen resultado. Cabe señalar que de nada serviría tener la posibilidad de enviar más tráfico en un red, si este tráfico tendrá un retardo elevado que impida que las aplicaciones funcionen correctamente.

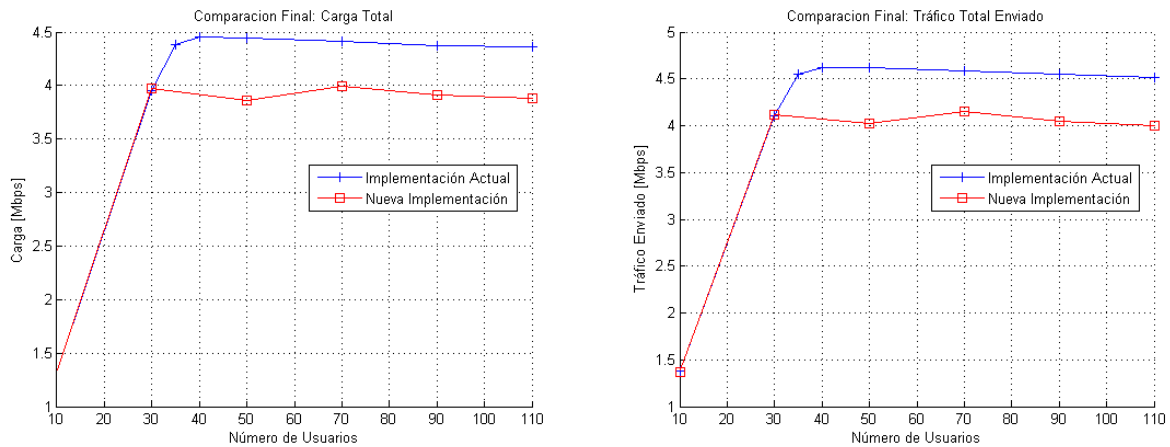


Figura 5.19: Comparación de carga y tráfico enviado totales.

6.1. Discusiones Finales

En el presente trabajo se propuso un algoritmo de calendarización de GPSS para redes Wimax basadas en el estándar IEEE 802.16. Este diseño demostró ser capaz de complementar y subsanar algunas limitaciones del algoritmo de calendarización propuesto por el software OPNET Modeler. De este modo, la solución presentada considera la *latencia máxima* en la calendarización, como lo indica el estándar IEEE 802.16, y logra disminuir el retardo de encolamiento de las conexiones con tipo de calendarización rtPS diseñando un algoritmo calendarizador para la SS, introduciendo el concepto de *satisfacción del retardo* y creando una nueva forma de calcular el tiempo entre *polls* para el tipo de calendarización mencionado.

El algoritmo calendarizador GPSS propuesto en este trabajo permite que una SS, en una red Wimax, tenga la capacidad de reasignar los recursos proporcionados por la BS como mejor le convenga en un momento determinado. Lo anterior implica que si una conexión debe enviar un paquete de inmediato porque éste está llegando a su tiempo límite (*deadline*), la SS distribuirá los recursos que le fueron concedidos de tal forma que dicho paquete sea enviado como se necesita con el fin de evitar el descarte. Este escenario conlleva a ofrecer una mejor QoS en las conexiones sensibles al retardo, sin afectar significativamente a las demás conexiones.

6.2. Contribuciones

Como parte de las contribuciones de este trabajo, se obtuvo un modelo de simulación de red Wimax Móvil, el cual está basado en una arquitectura punto multipunto. Con este modelo es posible simular diferentes escenarios de tráfico y observar los efectos de estos en la red en cuestión. Lo anterior permite predecir el comportamiento dinámico del sistema bajo diferentes demandas de tráfico, antes de implementar físicamente un sistema de este tipo. Con ello se pueden reducir grandes costos de instalación, mantenimiento, operación y actualización.

Parte importante del modelo de simulación propuesto es que permite simular usuarios

móviles desplazándose de forma aleatoria dentro de una sola celda. Asimismo, el modelo está adaptado para poder realizar simulaciones donde se va incrementando el número de usuarios en la red, esta importante herramienta no se encuentra de forma predeterminada en OPNET Modeler.

Una segunda contribución es la introducción del concepto *Satisfacción del retardo*, en la sección 5.1.1, que permite saber si una cola rtPS tiene urgencia por enviar un paquete. La ecuación 5.1 es una adaptación, para un calendarizador GPSS de redes Wimax, del *indicador de satisfacción* presentado en [6].

La nueva forma de calcular el tiempo entre *polls* presentada en la sección 5.1.3 es otra de las contribuciones hechas por este trabajo. Esta nueva forma de cálculo permite que el tiempo entre cada arribo de *poll* de rtPS sea tal que no afecte en el retardo de encolamiento de los paquetes con este tipo de calendarización.

Finalmente, la mayor aportación de este trabajo consiste en la propuesta de un algoritmo de calendarización GPSS para redes Wimax basadas en el estándar IEEE 802.16 que asegura QoS para conexiones rtPS y subsana las limitaciones presentadas por el algoritmo de calendarización propuesto por OPNET Modeler sin afectar de forma significativa las demás conexiones.

6.3. Trabajo a Futuro

Se tiene la intención de que este trabajo sirva de referencia para posteriores investigaciones y aportaciones en el área del diseño de calendarizadores para redes de banda ancha, fijas y/o móviles, basadas en el estándar IEEE 802.16. Entre las posibles líneas de trabajo que esta tesis deja abiertas, se encuentran las siguientes:

- Diseño e implementación de otros algoritmos de calendarización para redes Wimax.
- Diseño de un modelo matemático y analítico para el algoritmo de calendarización propuesto en este trabajo.
- Considerar la modulación adaptativa y la codificación en el diseño de una técnica de calendarización.
- Diseño e implementación de calendarizadores GPSS para la SS de una red Wimax.

El diseño e implementación de algoritmo de calendarización es un área de investigación abierta actualmente, tanto fabricantes como investigadores enfocan su esfuerzo en desarrollar algoritmos de baja complejidad que ofrezca mejor QoS que los algoritmos propuestos hasta el día de hoy. Un mejor algoritmo implica una mejor asignación de recursos en una red.

La construcción de modelos analíticos es también parte importante en la investigación de algoritmos de calendarización. En ocasiones los diseños basados y demostrados solamente

en simulaciones son considerados como investigaciones sin análisis teórico profundo. Un modelo analítico siempre respalda un diseño de forma significativa.

En [14], se deja en claro que el uso de la modulación adaptiva y la codificación tiene un impacto directo en la QoS. El diseño de algoritmos de capa cruzada (MAC - PHY) es también un campo de investigación abierto actualmente. Generalmente, los algoritmos de calendarización asumen un canal perfecto y sin pérdidas y fuente de potencia ilimitada. Sin embargo, debido a la naturaleza del medio inalámbrico y a la movilidad de usuario, estas suposiciones no son validas. Por ejemplo, una SS puede recibir otorgamientos pero quizás no sea capaz de transmitir exitosamente debido a una tasa de pérdida alta.

El estándar IEEE 802.16 indica que la SS debe tener la capacidad de administrar los recursos que le fueron otorgados. Típicamente ésta usará el ancho de banda para la conexión que la solicitó, pero no necesariamente. El diseño de calendarizadores GPSS se vuelve entonces importante en la toma de decisiones de una SS para aprovechar de mejor manera los recursos de la red.

Código del Nuevo Algoritmo calendarizador GPSS

A continuación se presenta el código en proto-C del algoritmo de calendarización propuesto en este trabajo.

A.1. Función *wimax_ss_control_gpss_ie_decompose()*

```
static List*
wimax_ss_control_gpss_ie_decompose (WimaxT_IE* gpss_ie_ptr, WimaxT_Request_Purpose purpose)
{
    List* ie_lptr = OPC_NIL;

    /* This function represents the decoder that takes a per-SS */
    /* IE and decomposes it in a list of per-connection IEs. */
    /* This is where the model can be modified to fit any */
    /* custom GPSS scheduler. For now, we are using a trivial */
    /* GPSS scheduler which decomposes the GPSS IE in the same */
    /* way as it was composed at the BS. */
    FIN (wimax_ss_control_gpss_ie_decompose (gpss_ie_ptr));

    if (purpose == WimaxC_Data)
    {
        int list_size;
        int no_shaper_queue_elems;
        int i, j;
        WimaxT_IE* temp_ie_ptr = OPC_NIL;
        WimaxT_Shaper_Queue_Elem* sq_elem_ptr = OPC_NIL;
        WimaxT_Shaper_Queue_Elem* sq_elem_tmp_ptr = OPC_NIL;
        double insertion_time;
        double deadline;
        double time_at_queue;
        double guard_time = 0.005;
        double satisfaction;
        int urgency_count = 0;

        no_shaper_queue_elems = op_prg_list_size (data_plane_config_ptr->mux_ptr->shaper_queue_elems_lptr);

        for (i = 0; i < no_shaper_queue_elems ; i++)
        {
            sq_elem_ptr = (WimaxT_Shaper_Queue_Elem*) op_prg_list_access
                (data_plane_config_ptr->mux_ptr->shaper_queue_elems_lptr, i);

            if (sq_elem_ptr->service_flow_ptr->params_ptr->scheduling_type == WimaxC_rtPS &&
                !oms_buffer_is_empty (sq_elem_ptr->queue_ptr->buffer))
```

```

{
deadline = sq_elem_ptr->service_flow_ptr->params_ptr->max_latency;
oms_buffer_access (sq_elem_ptr->queue_ptr->buffer, 0, &insertion_time);
time_at_queue = op_sim_time() - insertion_time;
satisfaction = deadline - guard_time - time_at_queue + 1.0;

if (satisfaction < 1.0)
{
sq_elem_ptr->urgency = OPC_TRUE;
urgency_count++;

/* ODB trace. */
if (op_prg_odb_ltrace_active ("wimax_gpss_sched"))
{
printf ("\n Cola rtPS con urgencia, CID [%d] Satisfaction [%f] deadline [%f]
time_at_queue [%f]", sq_elem_ptr->cid, satisfaction, deadline, time_at_queue);
}
}
else
{
sq_elem_ptr->urgency = OPC_FALSE;
}
}
}

/* Get the list size of the list of grants. */
ie_lptr = (List*) gpss_ie_ptr->extra_info_ptr;
list_size = op_prg_list_size (ie_lptr);

/* Go over the list and process each and every grant in the */
/* list. */
for (i=0 ; i < list_size; i++)
{
/* Access the first grant from the list. */
temp_ie_ptr = (WimaxT_IE*) op_prg_list_access (ie_lptr, i);
sq_elem_ptr = (WimaxT_Shaper_Queue_Elem*) wimax_sup_mux_conn_get
(data_plane_config_ptr->mux_ptr, temp_ie_ptr->cid);
if (sq_elem_ptr->service_flow_ptr->params_ptr->scheduling_type == WimaxC_rtPS)
{
if (sq_elem_ptr->urgency == OPC_TRUE)
{
sq_elem_ptr->urgency = OPC_FALSE;
urgency_count--;
}
}
else if (sq_elem_ptr->service_flow_ptr->params_ptr->scheduling_type == WimaxC_nrtPS)
{
/* ODB trace. */
if (op_prg_odb_ltrace_active ("wimax_gpss_sched"))
{
printf ("\n Grant para CID [%d] scheduling [%d]",
temp_ie_ptr->cid, sq_elem_ptr->service_flow_ptr->params_ptr->scheduling_type);
}
}
if(urgency_count>0)
{
for (j = 0; j < no_shaper_queue_elems ; j++)
{
sq_elem_tmp_ptr = (WimaxT_Shaper_Queue_Elem*) op_prg_list_access
(data_plane_config_ptr->mux_ptr->shaper_queue_elems_lptr, i);
if(sq_elem_tmp_ptr->urgency == OPC_TRUE)
{
WimaxT_Admission_Element* ith_flow_elem_ptr = OPC_NIL;
WimaxT_Service_Flow* flow_ptr = OPC_NIL;
WimaxT_Request_Element* bwr_ptr = OPC_NIL;
int num_ul_svc_flows = 0;
int k;
WimaxT_IE* ie_extra = OPC_NIL;

```

```

num_ul_svc_flows = op_prg_list_size (data_plane_config_ptr->ul_svc_flows_lptr);
k = 0;
while (flow_ptr == OPC_NIL && k < num_ul_svc_flows)
{
    ith_flow_elem_ptr = (WimaxT_Admission_Element*) op_prg_list_access
        (data_plane_config_ptr->ul_svc_flows_lptr, k);
    if (ith_flow_elem_ptr->service_flow_ptr->cid == sq_elem_tmp_ptr->cid)
    {
        flow_ptr = ith_flow_elem_ptr->service_flow_ptr;
    }
    k++;
}

/* ODB trace. */
if (op_prg_odb_ltrace_active ("wimax_gpss_sched"))
{
    printf ("\n Grant for CID [%d] stolen by CID[%d]: bwr_size_bits [%d]
        bwr_size_symbols [%d]", temp_ie_ptr->cid, sq_elem_tmp_ptr->cid,
        temp_ie_ptr->req_elem_ptr->bwr_size_bits,
        temp_ie_ptr->req_elem_ptr->bwr_size_symbols);
}

temp_ie_ptr->cid = sq_elem_tmp_ptr->cid;
bwr_ptr = (WimaxT_Request_Element *) wimax_support_request_element_create
    (temp_ie_ptr->req_elem_ptr->bwr_size_bits,
    temp_ie_ptr->req_elem_ptr->bwr_size_symbols,
    temp_ie_ptr->req_elem_ptr->purpose,
    temp_ie_ptr->req_elem_ptr->region, flow_ptr);
bwr_ptr->transmission_delay = temp_ie_ptr->req_elem_ptr->transmission_delay;
bwr_ptr->mgmt_info.module_ptr = temp_ie_ptr->req_elem_ptr->mgmt_info.module_ptr;
bwr_ptr->mgmt_info.module_id = temp_ie_ptr->req_elem_ptr->mgmt_info.module_id;

temp_ie_ptr->req_elem_ptr = bwr_ptr;
ie_extra = (WimaxT_IE*) op_prg_list_remove (ie_lptr,i);
op_prg_list_insert (ie_lptr, temp_ie_ptr, i);
urgency_count--;
sq_elem_tmp_ptr->urgency = OPC_FALSE;
}
}
}

FRET (ie_lptr);
}
else
{
    ie_lptr = (List*) gpss_ie_ptr->extra_info_ptr;

    FRET (ie_lptr);
}
}

```


A.2. Función *wimax_ss_control_one_svc_flow_parse ()* – Fragmento

```

static void
wimax_ss_control_one_svc_flow_parse (Objid sf_row_objid, Objid harq_params_objid, int row_index,
                                     WimaxT_Service_Flow_Direction direction,
                                     PrgT_String_Hash_Table* insertion_hash_ptr)
{
    WimaxT_Service_Class* service_class_ptr = OPC_NIL;
    WimaxT_Service_Flow* service_flow_ptr = OPC_NIL;
    char service_class_name [256];
    int mrate = 0;
    double estimated_pps_rate = 0.0;
    double poll_rate = 0.0;
    int toggle_int = 0;
    int ack_type;
    int moducoding = 0;
    WimaxT_Region region;
    /* HARQ parameters for the service flow. */
    Boolean is_harq_enabled;
    int max_rxmt;
    int num_acid;
    int ack_delay;
    int buffer_const;
    Boolean buffer_aggr_flag;

    /* This function is responsible for parsing one service flow */
    FIN (wimax_ss_control_one_svc_flow_parse ());

    /* Allocate a new service flow structure */
    /* The service flow allocated here will not be de-allocated */
    /* elsewhere, because this information will be useful up */
    /* until endsim. The memory will not increase, because this */
    /* allocation is done once for each service flow. There are */
    /* only a small number of service flows defined on the SS. */
    service_flow_ptr = (WimaxT_Service_Flow*) op_prg_cmo_alloc
        (wimax_catmem_handle, sizeof (WimaxT_Service_Flow));

    /* Store a reference to the data plane configuration. */
    service_flow_ptr->data_plane_info_ptr = data_plane_config_ptr;

    /* First, we parse the service class name. In case the service */
    /* class referenced in the name cannot be found, we don't have */
    /* to continue parsing this flow. */
    op_ima_obj_attr_get_str (sf_row_objid, "Service Class Name", 256, service_class_name);

    /* The average SDU size in bytes needs to be parsed now, so it */
    /* can be accessible when setting the default inter-poll time */
    /* for rtPS or nrtPS service flows. */
    op_ima_obj_attr_get (sf_row_objid, "Average SDU Size", &service_flow_ptr->avg_sdu_size);

    /* Retain the average SDU size in bits */
    service_flow_ptr->avg_sdu_size *= 8;

    /* Get the service flow parameters that correspond to the service class name */
    service_class_ptr = (WimaxT_Service_Class*) oms_data_def_entry_access ("Wimax Service Class",
        service_class_name);

    /* Check if a service class was obtained. */
    if (service_class_ptr == OPC_NIL)
    {
        /* Log that a service class definition is missing. */
        /* Make exception for the "Specify..." name. */
        if (0 != strcmp(service_class_name, "Specify..."))
        {

```

```

        wimax_undefined_service_class_name_log_write (service_class_name, "service flow");
    }

    /* Do not allocate any memory for this service flow */
    op_prg_mem_free (service_flow_ptr);

    FOUT;
}
else
{
    /* Set the service flow's attributes to the service flow parameters retrieved from */
    /* oms_data repository */
    service_flow_ptr->params_ptr = (WimaxT_SF_Params*) op_prg_cmo_alloc
        (wimax_catmem_handle, sizeof (WimaxT_SF_Params));
    service_flow_ptr->params_ptr->scheduling_type = service_class_ptr->scheduling_type;
    service_flow_ptr->params_ptr->traffic_priority = service_class_ptr->traffic_priority;
    service_flow_ptr->params_ptr->max_sustained_traffic_rate =
        service_class_ptr->max_sustained_traffic_rate;
    service_flow_ptr->params_ptr->min_reserved_traffic_rate =
        service_class_ptr->min_reserved_traffic_rate;
    service_flow_ptr->params_ptr->max_traffic_burst = service_class_ptr->max_traffic_burst;
    service_flow_ptr->params_ptr->max_latency = service_class_ptr->max_latency;
    service_flow_ptr->class_id = service_class_ptr->class_id;
    /* For rtPS and nrtPS flows, we need to set the */
    /* time between consecutive unsolicited polls. */
    if ((WimaxC_rtPS == service_class_ptr->scheduling_type) ||
        (WimaxC_nrtPS == service_class_ptr->scheduling_type))
    {
        /* Compute the estimated packet/second equivalent to */
        /* Max. Sustained Traffic Rate (for rtPS) or to Min. */
        /* Guaranteed Traffic Rate (for nrtPS). We call it */
        /* "estimated" packet/second rate, because its */
        /* accuracy depends on how close the average SDU size*/
        /* is to the true average of SDUs coming in through */
        /* the service flow. */
        /* The pps rate is used in the cases below to find */
        /* an appropriate default value for the time between */
        /* the unsolicited polls. */
        mrate = (WimaxC_rtPS == service_class_ptr->scheduling_type ?
            service_class_ptr->max_sustained_traffic_rate:
            service_class_ptr->min_reserved_traffic_rate);
        estimated_pps_rate = (double) mrate/(double) service_flow_ptr->avg_sdu_size;

        /* Check for default setting or invalid setting */
        if ((WIMAX_POLL_INTERVAL_DEFAULT == service_class_ptr->inter_poll_time)
            || (service_class_ptr->inter_poll_time <= 0.0))
        {
            int aux;
            /* The bits/second rate considered is */
            /* set above as follows: */
            /* 1. Maximum Sustained Traffic Rate, if rtPS class */
            /* 2. Minimum Guaranteed Traffic Rate, if nrtPS. */
            service_flow_ptr->params_ptr->inter_poll_time =
                1.0/(estimated_pps_rate * WIMAX_DEFAULT_POLL_TO_PACKET_RATIO);

            /* We consider max_lantecy as an argument for rtPS inter_poll_time calculation */
            if (WimaxC_rtPS == service_class_ptr->scheduling_type &&
                service_flow_ptr->params_ptr->inter_poll_time >
                service_flow_ptr->params_ptr->max_latency)
            {
                aux = 1;
                while (service_flow_ptr->params_ptr->inter_poll_time >
                    service_flow_ptr->params_ptr->max_latency)
                {
                    service_flow_ptr->params_ptr->inter_poll_time =
                        1.0/(estimated_pps_rate * WIMAX_DEFAULT_POLL_TO_PACKET_RATIO*aux);
                    aux++;
                }
            }
        }
    }
}

```

```

    }

    if (WIMAX_TRACE_POLLING)
    {
        char msg [256];
        char sched_type [64];

        sprintf (msg, "At service flow parsing, Auto-Calculated/NOT_USED Unsolicited Poll Time
                    translates into %.6f seconds", service_flow_ptr->params_ptr->inter_poll_time);
        op_prg_odb_print_major (msg, OPC_NIL);

        sprintf (msg, "The %.6f value is derived as follows:",
                service_flow_ptr->params_ptr->inter_poll_time);
        op_prg_odb_print_minor (msg, OPC_NIL);

        if (WimaxC_rtPS == service_flow_ptr->params_ptr->scheduling_type)
            sprintf (sched_type, "For this rtPS flow, the Maximum Sustained Traffic Rate of");
        else
            sprintf (sched_type, "For this nrtPS flow, the Minimum Guaranteed Traffic Rate of");

        sprintf (msg, "%s (%d bits/second)", sched_type, mrate);
        op_prg_odb_print_minor (msg, OPC_NIL);

        sprintf (msg, "divided by avg SDU size (%d bytes) results in avg (%.2f pks/second)",
                service_flow_ptr->avg_sdu_size, estimated_pps_rate);
        op_prg_odb_print_minor (msg, OPC_NIL);

        sprintf (msg, "with %d polls per packet yields %.2f polls per second rate, which
                    inverted gives %.6f seconds above.", WIMAX_DEFAULT_POLL_TO_PACKET_RATIO*aux,
                    1/service_flow_ptr->params_ptr->inter_poll_time,
                    service_flow_ptr->params_ptr->inter_poll_time);

        op_prg_odb_print_minor (msg, OPC_NIL);
    }

}

else
{
    ...
}

}
}

```

Apéndice B

Diagramas de Árbol

B.1. Solicitud de ancho de banda para conexiones DL

Rastreo de los estados, directivas y funciones involucradas en la solicitud de ancho de banda para conexiones rtPS, nrtPS y BE en dirección DL.

```
-----Invoking process ID (73)-----
+- process ("wimax_mac")
|
|-----state (idle): exit executives-----
|
+- op_intrpt_type ()
| | intrpt type      (stream intrpt)
+-----
|
|-----computing transition conditions-----
|
|-----evaluating condition "LOWER_LAYER_DATA_PK_ARRIVED"-----
|
|-----evaluated false-----
|
|-----evaluating condition "GRANT_ARRIVAL"-----
|
|-----evaluated false-----
|
|-----evaluating condition "CONTROL_PK_ARRIVED"-----
|
|-----evaluated false-----
|
|-----evaluating condition "NO_BS_FOR_THIS_SS"-----
|
|-----evaluated false-----
|
|-----evaluating condition "HIGHER_LAYER_DATA_PK_ARRIVED"-----
|
|-----evaluated true-----
|
|-----evaluating condition "SS_DISCONNECT"-----
|
|-----evaluated false-----
|
|-----transition from state (idle) to state (hl_pk)-----
|
* condition: "HIGHER_LAYER_DATA_PK_ARRIVED"
```

```

| _____state (hl_pk): enter executives_____
|
| +- wimax_mac_load_stat_only_node_update ()
| |
| +-----
|
| +- wimax_mac_higher_layer_pk_classify (pkptr)
| |
| | +- wimax_mac_service_class_get (pkptr, class_str, class_id_ptr)
| | |
| | | +- wimax_mac_service_class_find (socket_info_ptr, class_ptr)
| | | |
| | | | +-----
| | | |
| | | | +-----
| | |
| | +- wimax_mac_dest_addr_get ()
| | |
| | | +-----
| | |
| | +-----
|
| +- wimax_support_mac_connection_pk_enqueue ()
| |
| | +- wimax_support_packet_role_get ()
| | |
| | | +-----
| |
| | +- wimax_support_mac_load_stat_update ()
| | |
| | | +-----
| |
| | +- wimax_support_mac_pk_in_queue_insert (pkptr, sq_elem_ptr, service_flow_ptr,
| | |                                     data_plane_config_ptr, contention_ptr)
| | |
| | |
| | | +- oms_buffer_enqueue (buffer_handle, element, insert_data_ptr)
| | | |
| | | | +-----
| | |
| | | +- wimax_mac_queue_size_stat_update ()
| | | |
| | | | +-----
| | |
| | | +- wimax_support_mac_pk_in_queue_efficiency_off (pkptr, sq_elem_ptr, service_flow_ptr)
| | | |
| | | | +- wimax_support_bw_request_insert ()
| | | | |
| | | | | +- wimax_support_mac_to_control_bw_req_indicate ()
| | | | | |
| | | | | | +- prg_cmo_alloc (handle, size)
| | | | | | |
| | | | | | |   handle      (0x038C4158)
| | | | | | |   size        (32)
| | | | | | |   addr         (0x096944E0)
| | | | | | |   +-----
| | | | | |
| | | | | | +- op_pro_self ()
| | | | | | |
| | | | | | |   process ID   (73)
| | | | | | |   +-----
| | | | | |
| | | | | | +- op_pro_equal (handle1, handle2)
| | | | | | |
| | | | | | |   handle 1's id (73)
| | | | | | |   handle 2's id (1541)
| | | | | | |   equal         (false)
| | | | | | |   +-----
| | | | | |
| | | | | | +- op_pro_invoke (pro_handle, argmem_ptr)

```

```

| | | | | process ID      (1541)
| | | | | arg mem       (0x09694E0)
| | | | | process name  (wimax_bs_control)
| | | | |
| | | | | * process ID (1541), of type (wimax_bs_control) invoked indirectly
| | | | |
| | | | | +- process ("wimax_bs_control")
| | | | | |
| | | | | | _____state (idle): exit executives_____
| | | | | |
| | | | | | +- op_pro_self ()
| | | | | | | process ID      (1541)
| | | | | | +-----
| | | | | |
| | | | | | +- op_pro_invoker (pro_handle, invmode_ptr)
| | | | | | | process ID      (1541)
| | | | | | | invoker proc. ID (73)
| | | | | | +-----
| | | | | |
| | | | | | +- op_intrpt_type ()
| | | | | | | intrpt type      (stream intrpt)
| | | | | | +-----
| | | | | |
| | | | | | _____computing transition conditions_____
| | | | | |
| | | | | | _____evaluating condition "CONTROL_PACKET"_____
| | | | | |
| | | | | | _____evaluated false_____
| | | | | |
| | | | | | _____evaluating condition "MAP_TIME"_____
| | | | | |
| | | | | | _____evaluated false_____
| | | | | |
| | | | | | _____evaluating condition "LOCAL_BW_REQ_IND"_____
| | | | | |
| | | | | | _____evaluated true_____
| | | | | |
| | | | | | _____transition from state (idle) to state (idle)_____
| | | | | |
| | | | | | * condition: "LOCAL_BW_REQ_IND"
| | | | | |
| | | | | | * executive: "BW_REQ_PROCESS"
| | | | | |
| | | | | | +- wimax_bs_control_bw_req_indication_process (sf_ptr, bw_req_size)
| | | | | | |
| | | | | | | +- wimax_bs_control_sched_bw_req_notify ()
| | | | | | | |
| | | | | | | | +- wimax_support_by_phy_overhead_rate_augment ()
| | | | | | | | |
| | | | | | | | +-----
| | | | | | | |
| | | | | | | | +- wimax_bs_control_bw_req_requirements_update ()
| | | | | | | | |
| | | | | | | | | +- wimax_support_by_phy_overhead_rate_augment ()
| | | | | | | | | |
| | | | | | | | | +-----
| | | | | | | | |
| | | | | | | | | +- wimax_bs_control_sched_bw_req_insert ()
| | | | | | | | | |
| | | | | | | | | | +- wimax_support_request_element_create
| | | | | | | | | | |
| | | | | | | | | | +-----
| | | | | | | | | |
| | | | | | | | | | +- wimax_bs_control_sched_bw_req_ps_insert ()
| | | | | | | | | | |
| | | | | | | | | | | +- wimax_bs_control_sched_poll_serv_q_access_by_conn_id()
| | | | | | | | | | | |
| | | | | | | | | | | +-----

```



```

+-----
+- op_pk_get (instrm_index)
|   strm. index      (1)
|   packet ID       (1457175)
+-----

_____computing transition conditions_____
_____evaluating condition "LOWER_LAYER_DATA_PK_ARRIVED"_____
_____evaluated false_____
_____evaluating condition "GRANT_ARRIVAL"_____
_____evaluated false_____
_____evaluating condition "CONTROL_PK_ARRIVED"_____
_____evaluated false_____
_____evaluating condition "NO_BS_FOR_THIS_SS"_____
_____evaluated false_____
_____evaluating condition "HIGHER_LAYER_DATA_PK_ARRIVED"_____
_____evaluated true_____
_____evaluating condition "SS_DISCONNECT"_____
_____evaluated false_____
_____transition from state (idle) to state (hl_pk)_____

* condition: "HIGHER_LAYER_DATA_PK_ARRIVED"
_____state (hl_pk): enter executives_____

+- wimax_mac_load_stat_only_node_update ()
|
+-----

+- wimax_mac_higher_layer_pk_classify (pkptr)
|
| +- wimax_mac_service_class_get (pkptr, class_str, class_id_ptr)
| |
| | +- wimax_mac_service_class_find (socket_info_ptr, class_ptr)
| | |
| | | +-----
| | | |
| | | +-----
| | |
| +- wimax_mac_dest_addr_get ()
| |
| | +-----
| |
+-----

+- wimax_support_mac_connection_pk_enqueue ()
|
| +- wimax_support_mac_load_stat_update ()
| |
| | +-----
| |
| +- wimax_support_mac_pk_in_queue_insert (pkptr, sq_elem_ptr, service_flow_ptr,
| | |                                     data_plane_config_ptr, contention_ptr)
| | |

```



```

| -----evaluated false-----
| -----evaluating condition "HIGHER_LAYER_DATA_PK_ARRIVED"-----
| -----evaluated false-----
| -----evaluating condition "SS_DISCONNECT"-----
| -----evaluated false-----
| -----transition from state (idle) to state (control_pk)-----
| * condition: "CONTROL_PK_ARRIVED"
| -----state (control_pk): enter executives-----
| +- wimax_mac_pkt_accepted()
| |
| +-----
| +- wimax_mac_send_to_control_plane()
| |
| | +- prg_cmo_alloc (handle, size)
| | |   handle      (0x038C4158)
| | |   size        (32)
| | |   addr         (0x09692708)
| | +-----
| | +- op_pro_invoke (pro_handle, argmem_ptr)
| | |   process ID   (1542)
| | |   arg mem      (0x09692708)
| | |   process name (wimax_ss_control)
| | |
| | | * process ID (1542), of type (wimax_ss_control) invoked indirectly
| | |
| | | +- process ("wimax_ss_control")
| | | |
| | | | -----state (idle): exit executives-----
| | | | +- wimax_ss_control_interrupts_process (void)
| | | | |
| | | | +- op_pro_self ()
| | | | |   process ID   (1542)
| | | | +-----
| | | | +- op_pro_invoker (pro_handle, invmode_ptr)
| | | | |   process ID   (1542)
| | | | |   invoker proc. ID (85)
| | | | +-----
| | | | +- op_pro_argmem_access ()
| | | | |   arg mem      (0x09692708)
| | | | +-----
| | | | +-----
| | | | -----computing transition conditions-----
| | | | -----evaluating condition "INITIATE_HO"-----
| | | | -----evaluated false-----
| | | | -----evaluating condition "START_SCAN_INTERVAL"-----
| | | | -----evaluated false-----
| | | | -----evaluating condition "UL_MAP"-----
| | | |
| | | |

```

```

| | | | _____evaluated true_____
| | | | _____evaluating condition "OTHER_CONTROL_PACKET"_____
| | | | _____evaluated false_____
| | | | _____evaluating condition "T4_EXPIRED || (T3_EXPIRED && !RETRIES_EXCEEDED)"_____
| | | | _____evaluated false_____
| | | | _____evaluating condition "(T3_EXPIRED && RETRIES_EXCEEDED) || (RESOURCE_RETAIN_EXPIRED)
| | | | | (INITIATE_MAC_RESET)"_____
| | | | _____evaluated false_____
| | | | _____evaluating condition "RNG_RSP"_____
| | | | _____evaluated false_____
| | | | _____evaluating condition "TO_SLEEP"_____
| | | | _____evaluated false_____
| | | | _____evaluating condition "TO_LISTEN"_____
| | | | _____evaluated false_____
| | | | _____evaluating condition "TO_NORMAL || BE_NRTPS_BWR"_____
| | | | _____evaluated false_____
| | | | _____transition from state (idle) to state (idle)_____
| | | | * condition: "UL_MAP"
| | | | * executive: "ul_map_decode"
| | | | +- wimax_ss_control_ul_map_decode (pkptr)
| | | | |
| | | | | +- wimax_phy_power_ctrl_open_loop_uplink_loss_update ()
| | | | | |
| | | | | +-----
| | | | |
| | | | | +- wimax_ss_control_grants_decode (map_ptr)
| | | | | |
| | | | | +-----
| | | | |
| | | | | +- wimax_ss_control_grants_decode (map_ptr)
| | | | | |
| | | | | +-----
| | | | |
| | | | | +- wimax_ss_control_grants_decode (map_ptr)
| | | | | |
| | | | | +-----
| | | | |
| | | | | +- wimax_ss_control_grants_decode (map_ptr)
| | | | | |
| | | | | | +- wimax_support_map_hash_element_get ()
| | | | | | |
| | | | | | +-----
| | | | | |
| | | | | | +- wimax_ss_control_gpss_ie_decompose (gpss_ie_ptr)
| | | | | | |
| | | | | | +-----
| | | | | |
| | | | | | +- wimax_cdma_bwr_grant_arrival ()
| | | | | | |
| | | | | | +-----

```

```

| | | | | +- wimax_support_grant_to_data_plane_give (grant_ptr)
| | | | | |
| | | | | +- op_ev_state_install (state_ptr)
| | | | | |   state_ptr      (0x0394B108)
| | | | | |   previous state_ptr (0x00000000)
| | | | | | +-----+
| | | | | |
| | | | | +- op_id_self ()
| | | | | |   object ID      (10847)
| | | | | | +-----+
| | | | | |
| | | | | +- op_sim_time ()
| | | | | |   sim time       (20.00548384 sec. [20s . 005ms 483us 840ns])
| | | | | | +-----+
| | | | | |
| | | | | +- op_intrpt_schedule_remote (time, code, mod_objid)
| | | | | |   time          (20.00548384 sec. [20s . 005ms 483us 840ns])
| | | | | |   code           (6,000)
| | | | | |   module objid   (10847)
| | | | | |   module        (top.Logical Network.Mobile_1.wimax_mac [Objid=10847])
| | | | | |   event ID     (2992430)
| | | | | | +-----+
| | | | | |
| | | | | +- op_ev_state_install (state_ptr)
| | | | | |   state_ptr      (0x00000000)
| | | | | |   previous state_ptr (0x0394B108)
| | | | | | +-----+
| | | | | |
| | | | | +-----+
| | | | |
| | | | | +- wimax_ss_control_grants_decode (map_ptr)
| | | | | |
| | | | | +- wimax_ss_control_contention_rng_consume (contention_ie_ptr)
| | | | | |
| | | | | | +-----+
| | | | | |
| | | | | +- wimax_ss_control_contention_bwr_consume (contention_ie_ptr)
| | | | | |
| | | | | +- wimax_ss_control_contention_bwr_consume_efficiency_phy ()
| | | | | |
| | | | | | +- wimax_ss_control_process_cdma_bwr_retx ()
| | | | | | |
| | | | | | | +-----+
| | | | | | |
| | | | | | +- wimax_ss_control_pending_cdma_bwr_update ()
| | | | | | |
| | | | | | | +-----+
| | | | | | |
| | | | | | +- wimax_ss_control_pending_bwr_grant_update ()
| | | | | | |
| | | | | | | +-----+
| | | | | | |
| | | | | | +- wimax_ss_control_get_bwr_from_be_queues ()
| | | | | | |
| | | | | | | +- wimax_support_bw_req_conn_total_requirements_get ()
| | | | | | | |
| | | | | | | | +- wimax_support_update_bwr_requirements (sq_elem_ptr)
| | | | | | | | |
| | | | | | | | | +-----+
| | | | | | | | |
| | | | | | | | | +-----+
| | | | | | | | |
| | | | | | | | | +- wimax_support_bw_req_conn_total_requirements_get ()
| | | | | | | | | |
| | | | | | | | | | +- wimax_support_update_bwr_requirements (sq_elem_ptr)

```

```

| | | | | | | |
| | | | | | | +-----
| | | | | | | |
| | | | | | | +-----
| | | | | | | +- wimax_support_bw_req_conn_total_requirements_get ()
| | | | | | | +- wimax_support_update_bwr_requirements (sq_elem_ptr)
| | | | | | | |
| | | | | | | +-----
| | | | | | | |
| | | | | | | +-----
| | | | | | | +- wimax_support_bw_req_conn_total_requirements_get ()
| | | | | | | +- wimax_support_update_bwr_requirements (sq_elem_ptr)
| | | | | | | |
| | | | | | | +-----
| | | | | | | |
| | | | | | | +-----
| | | | | | | +- wimax_support_bw_req_conn_total_requirements_get ()
| | | | | | | +- wimax_support_update_bwr_requirements (sq_elem_ptr)
| | | | | | | |
| | | | | | | +-----
| | | | | | | |
| | | | | | | +-----
| | | | | | | +- wimax_ss_control_pending_intial_cdma_codes_process ()
| | | | | | | |
| | | | | | | +-----
| | | | | | | +- wimax_ss_control_cdma_bwr_send ()
| | | | | | | |
| | | | | | | +-----
| | | | | | | |
| | | | | | | +-----
| | | | | | | |
| | | | | | | +-----
| | | | | | | +- wimax_ss_control_cqich_allocation_consume
| | | | | | | |
| | | | | | | +-----
| | | | | | | +- wimax_ss_control_grants_decode (map_ptr)
| | | | | | | |
| | | | | | | +-----
| | | | | | | +- wimax_ss_control_grants_decode (map_ptr)
| | | | | | | |
| | | | | | | +-----
| | | | | | | |
| | | | | | | +-----
| | | | | | | _____state (idle): enter executives_____
| | | | | | | |
| | | | | | | +-----
| | | | | | | +-----
| | | | | | | +-----
| | | | | | | _____state (control_pk): exit executives_____

```

```

| _____transition from state (control_pk) to state (idle)_____
| * condition: "default"
| _____state (idle): enter executives_____
|
+-----
|
| _____Returning from process ID (85)_____
|
| _____Invoking process ID (85)_____
|
+- process ("wimax_mac")
|
| _____state (idle): exit executives_____
|
+- op_intrpt_type ()
| |   intrpt type      (remote intrpt)
+-----
|
+- op_intrpt_code ()
| |   code              (6,000)
+-----
|
| _____computing transition conditions_____
|
| _____evaluating condition "LOWER_LAYER_DATA_PK_ARRIVED"_____
|
| _____evaluated false_____
|
| _____evaluating condition "GRANT_ARRIVAL"_____
|
| _____evaluated true_____
|
| _____evaluating condition "CONTROL_PK_ARRIVED"_____
|
| _____evaluated false_____
|
| _____evaluating condition "NO_BS_FOR_THIS_SS"_____
|
| _____evaluated false_____
|
| _____evaluating condition "HIGHER_LAYER_DATA_PK_ARRIVED"_____
|
| _____evaluated false_____
|
| _____evaluating condition "SS_DISCONNECT"_____
|
| _____evaluated false_____
|
| _____transition from state (idle) to state (tx_schedule)_____
|
| * condition: "GRANT_ARRIVAL"
|
| _____state (tx_schedule): enter executives_____
|
+- wimax_mac_efficiency_off_grants_process (grant_lptr)
| |
| | +- wimax_mac_efficiency_off_grant_process (grant_ptr)
| | |
| | | +- wimax_mac_poll_consume(grant_ptr, sq_elem_pptr, bw_req_pkpptr)
| | | |
| | | | +- wimax_support_bw_req_conn_total_requirements_get ()
| | | | |
| | | | +- wimax_support_update_bwr_requirements (sq_elem_ptr)
| | | | |
| | | | +-----
| | | |
| | | |

```

```

| | | | +-----
| | | |
| | | | +- wimax_support_bw_req_request_requirements_by_cid_pk_get ()
| | | | |
| | | | +- wimax_support_update_bwr_requirements (sq_elem_ptr)
| | | | |
| | | | +-----
| | | |
| | | | +- wimax_support_bw_request_pk_generate ()
| | | | |
| | | | +-----
| | | | |
| | | | +-----
| | | | |
| | | | +-----
| | | |
| | | +- wimax_support_pk_send ()
| | | |
| | | | +-----
| | | |
| | | +-----
| | |
| | +- wimax_support_request_element_destroy ()
| | |
| | +-----
| |
| +-----
|
|-----state (tx_schedule): exit executives-----
|-----transition from state (tx_schedule) to state (idle)-----
|
| * condition: "default"
|-----state (idle): enter executives-----
|-----
|-----Returning from process ID (85)-----

```

B.2.3. La solicitud de ancho de banda llega a la BS

```

|-----Invoking process ID (73)-----
|
+- process ("wimax_mac")
|
|-----state (idle): exit executives-----
|
+- op_intrpt_type ()
| | intrpt type (stream intrpt)
| +-----
|
+- op_pk_get (instrm_index)
| | strm. index (0)
| | packet ID (1457618)
| +-----
|
+- wimax_mac_pkt_accepted()
| |
| +-----
|-----computing transition conditions-----
|-----evaluating condition "LOWER_LAYER_DATA_PK_ARRIVED"-----

```

```

| -----evaluated false-----
| -----evaluating condition "GRANT_ARRIVAL"-----
| -----evaluated false-----
| -----evaluating condition "CONTROL_PK_ARRIVED"-----
| -----evaluated true-----
| -----evaluating condition "NO_BS_FOR_THIS_SS"-----
| -----evaluated false-----
| -----evaluating condition "HIGHER_LAYER_DATA_PK_ARRIVED"-----
| -----evaluated false-----
| -----evaluating condition "SS_DISCONNECT"-----
| -----evaluated false-----
| -----transition from state (idle) to state (control_pk)-----
| * condition: "CONTROL_PK_ARRIVED"
| -----state (control_pk): enter executives-----
| +- wimax_mac_send_to_control_plane()
| |
| | +- prg_cmo_alloc (handle, size)
| | |   handle      (0x038C4158)
| | |   size        (32)
| | |   addr         (0x040869B8)
| | | +-----+
| | |
| | | +- op_pro_invoke (pro_handle, argmem_ptr)
| | | |   process ID   (1541)
| | | |   arg mem      (0x040869B8)
| | | |   process name  (wimax_bs_control)
| | | |
| | | | * process ID (1541), of type (wimax_bs_control) invoked indirectly
| | | |
| | | | +- process ("wimax_bs_control")
| | | | |
| | | | | -----state (idle): exit executives-----
| | | | |
| | | | | +- op_pro_self ()
| | | | | |   process ID   (1541)
| | | | | | +-----+
| | | | |
| | | | | +- op_pro_invoker (pro_handle, invmode_ptr)
| | | | | |   process ID   (1541)
| | | | | |   invoker proc. ID (73)
| | | | | | +-----+
| | | | |
| | | | | +- op_intrpt_type ()
| | | | | |   intrpt type   (stream intrpt)
| | | | | | +-----+
| | | | |
| | | | | -----computing transition conditions-----
| | | | |
| | | | | -----evaluating condition "CONTROL_PACKET"-----
| | | | |
| | | | | -----evaluated true-----
| | | | |
| | | | | -----evaluating condition "MAP_TIME"-----

```



```
| | | | |
| | | | | +-----
| | | | |
| | | | | +-----
| | | | |
| | | | | _____state (idle): enter executives_____
| | | | |
| | | | | +-----
| | | | | +-----
| | | | |
| | | | | +-----
| | | | |
| | | | | +-----
| | | | |
| | | | | _____state (control_pk): exit executives_____
| | | | |
| | | | | _____transition from state (control_pk) to state (idle)_____
| | | | |
| | | | | * condition: "default"
| | | | |
| | | | | _____state (idle): enter executives_____
| | | | |
| | | | | +-----
| | | | |
| | | | | _____Returning from process ID (73)_____
```


Lista de Acrónimos

AAS	Adaptive Antena System
ACK	Acknowledgment
AMC	Adaptive Modulation and Coding
ARQ	Automatic Repeat Request
ASE	Average Spectral Efficiency
BS	Base Station
BSN	Block Sequence Number
BPSK	Binary Phase Shift Keying
BWA	Broadband Wireless Access
BWR	Bandwidth Request
CID	Connection Identifier
CP	Cyclic Prefix
CRC	Cyclic Redundancy Check
CS	Subcapa de Convergencia
DAMA	Demand Assigned Multiple Access
DC	Direct Current
DFPQ	Deficit Fair Priority Queue
DL	DownLink
DOCSIS	Data Over Cable Service Interface Specification
DRR	Deficit Round Robin
DSL	Digital Subscriber Line
DVB	Digital Video Broadcasting
EDF	Early Deadline First
ETSI BRAN	European Telecommunications Standards Institute Broadband Radio Access Networks
FCFS	First Come First Served
FCH	Frame Control Header
FDD	Frequency Division Duplex
FEC	Forward Error Correction

FFT	Fast Fourier Transform
FIFO	First In First Out
FSM	Finite State Machine
FTP	File Transfer Protocol
GPC	Grant per Connection
GPSS	Grant per SS
GT	Guard Time
HTTP	Hypertext Transfer Protocol
IE	Information Element
IEEE	Institute of Electrical and Electronics Engineers
IFFT	Inverse Fast Fourier Transform
IP	Internet Protocol
ISP	Internet Service Provider
ITU	International Telecommunications Union
LAN	Local Area Network
LLC	Logical Link Control
LOS	Line of Sight
MAC	Media Access Control
MAN	Metropolitan Area Network
MCS	Modulation and Coding Scheme
MDRR	Modified Deficit Round Robin
MIMO	Multiple Input Multiple Output
MMS	Multimedia Messaging Service
MPEG	Moving Picture Experts Group
MRTR	Minimum Reserved Traffic Rate
MSTR	Maximum Sustained Traffic Rate
NewMDRR	New Modified Deficit Round Robin
NLOS	Non Line of Sight
nrtps	Non Real Time Polling Service
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
OPNET	Optimized Network Engineering Tool
OSI	Open System Interconnection
PDU	Protocol Data Unit
PHY	Capa Física
PMP	Point to Multiple Point
PQ	Priority Queue
PS	Physical Slot
PS	Pollins Service: nrTPS y rtPS
PSTN	Public Switched Telephone Network

PTP	Point to Point
QAM	Quadrature Amplitude Modulation
QoS	Calidad de Servicio
QPSK	Quadrature Phase Shift Keying
RFCs	Request for Comments
RR	Round Robin
rtPS	Real Time Polling Service
SC	Single Carrier
SDU	Service Data Unit
SNR	Signal Noise Ratio
SOFDMA	Scalable Orthogonal Frequency Division Multiple Access
SP	Strict Priorization
SS	Suscriber Station
STD	State Transition Diagrams
TCP	Transmission Control Protocol
TCP/IP	Transmission Control Protocol/Internet Protocol
TDD	Time Division Duplex
UDP	User Datagram Protocol
UL	UpLink
VoIP	Voice over IP
WAN	Wide Area Network
WFQ	Weighted Fair Queueing
Wimax	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network

Bibliografía

- [1] R. Attar, et al, *Evolution of cdma2000 cellular networks: multicarrier EV-DO*, IEEE Communications Magazine, Vol. 44, Issue 3, 2006.
- [2] IEEE, *IEEE Standard for Local and Metropolitan Area Networks Part 11, Amendment 4: Further Higher Data Rate Extension in the 2.4 GHz Band*, 2006.
- [3] IEEE, *IEEE Standards for Local and Metropolitan Area Networks - Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems - Amendment 3: Management PLANE Procedure and Services*, IEEE Std 802.16g 2007 (Amendment to IEEE Std 802.16-2004), 2007.
- [4] Cable Television Laboratories, Inc., *Data Over Cable Service Interface Specification 1.0* (Abril 1999).
- [5] Molina, Francisco J, *Redes de Área Local*. Editorial Alfaomega, México, 2004.
- [6] Qingwen Liu, Xin Wang, y G. B Giannakis, *A cross-layer scheduling algorithm with QoS support in wireless networks*, IEEE Transactions on Vehicular Technology 55, n°. 3 (Mayo 2006): 839-847.
- [7] H. Fattah y C. Leung, *An overview of scheduling algorithms in wireless multimedia networks*, IEEE Wireless Communications 9, n°. 5 (Octubre 2002): 76- 83.
- [8] Chakchai So-In, R. Jain, y A. -K Tamimi, *Scheduling in IEEE 802.16e mobile WiMAX networks: key issues and a survey*, IEEE Journal on Selected Areas in Communications 27, n°. 2 (Febrero 2009): 156-171.
- [9] Q. Liu, X. Wang, and G. Giannakis, *A Cross-Layer Scheduling Algorithm with QoS Support in Wireless Networks*, in IEEE Transactions on Vehicular Technologies, Vol. 55, No.3, pp. 839 - 847,2006.
- [10] Xiaojuan Xie, Haining Chen, y Hongyi Wu, *Simulation studies of a Fair and Effective Queueing algorithm for WiMAX resource allocation*, en Communications and Networking in China, 2008. ChinaCom 2008. Third International Conference on, 2008, 281-285.
- [11] M. Hawa, *Stochastic Evaluation of Fair Scheduling with Applications to Quality-of-Service in Broadband Wireless Access Networks*, en Ph.D Thesis, University of Kansas, 2.

- [12] K. Wongthavarawat y A. Ganz, ***IEEE 802.16 based last mile broadband wireless military networks with quality of service support***, en IEEE Military Communications Conference, 2003. MILCOM 2003, vol. 2 (presented at the IEEE Military Communications Conference, 2003. MILCOM 2003, IEEE, 2003), 779- 784 Vol.2.
- [13] Jianfeng Chen, Wenhua Jiao, y Hongxi Wang, ***A service flow management strategy for IEEE 802.16 broadband wireless access systems in TDD mode***, en 2005 IEEE International Conference on Communications, 2005. ICC 2005, vol. 5 (presented at the 2005 IEEE International Conference on Communications, 2005. ICC 2005, IEEE, 2005), 3422- 3426 Vol. 5.
- [14] M. Fathi y H. Taheri, ***Queuing analysis for dynamic bandwidth allocation in IEEE 802.16 standard***, en Wireless Pervasive Computing, 2008. ISWPC 2008. 3rd International Symposium on, 2008, 616-619.
- [15] Qingwen Liu, Shengli Zhou, y G.B. Giannakis, ***Queuing with adaptive modulation and coding over wireless links: cross-Layer analysis and design***, Wireless Communications, IEEE Transactions on 4, n^o. 3 (2005): 1142-1153.
- [16] Howon Lee y Dong-Ho Cho, ***Multi-User Queuing Analysis Considering AMC for Wireless VoIP Services***, en IEEE International Conference on Communications Workshops, 2009. ICC Workshops 2009 (presented at the IEEE International Conference on Communications Workshops, 2009. ICC Workshops 2009, IEEE, 2009), 1-5.
- [17] Sheng Zhou et al., ***Queuing Analysis on MIMO Systems with Adaptive Modulation and Coding***, en Communications, 2008. ICC '08. IEEE International Conference on, 2008, 3400-3405.
- [18] G. D Castellanos y J. Y Khan, ***Performance of WiMAX packet schedulers for multi-class traffic***, en 2010 IEEE Latin-American Conference on Communications (LAT-INCOM) (presented at the 2010 IEEE Latin-American Conference on Communications (LATINCOM), IEEE, 2010), 1-6.
- [19] C. Ravichandiran, P. Raj, y Vaithyanathan, ***Analysis and Modification of Scheduling Algorithm for IEEE 802.16e (Mobile WiMAX)***, en Second International Conference on Communication Software and Networks, 2010. ICCSN 2010 (presented at the Second International Conference on Communication Software and Networks, 2010. ICCSN 2010, IEEE, 2010), 434-438.
- [20] Gang Zhang et al., ***Quality of Service Scheduling Based on GPSS in IEEE 802.16 WiMax Networks***, en 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM 2008 (presented at the 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM 2008, IEEE, 2008), 1-4.
- [21] Loutfi Nuaymi, ***Wimax: Technology for Broadband Wireless Access***, John Wiley & Sons Ltd, United Kingdom , 2007.

-
- [22] Jeffrey, G. Andrews et al. , *Fundamentals of Wimax: Understanding Broadband Wireless Networking*, Prentice Hall, USA, 2007.
- [23] Gang Zhang et al., *Quality of Service Scheduling Based on GPSS in IEEE 802.16 WiMax Networks*, en 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08 (presented at the 4th International Conference on Wireless Communications, Networking and Mobile Computing, 2008. WiCOM '08, IEEE, 2008), 1-4.