

Universidad Central “Marta Abreu” de Las Villas  
Facultad de Ingeniería Eléctrica  
Departamento de Telecomunicaciones y Electrónica



## TRABAJO DE DIPLOMA

# Mejora del desempeño del *buffer* de de- *jitter* a través de realimentación a nivel de radio en redes LTE

Tesis presentada en opción al grado de  
Ingeniero en Telecomunicaciones y Electrónica

Autor: Rigoberto Valentin López Fernández

Tutor: M.Sc. Carlos A. Rodríguez López  
Ing. Meyli Lucy Guevara Martín

Santa Clara

2015

“Año 57 de la Revolución”

Universidad Central “Marta Abreu” de Las Villas  
Facultad de Ingeniería Eléctrica  
Departamento de Telecomunicaciones y Electrónica



## TRABAJO DE DIPLOMA

# Mejora del desempeño del *buffer* de de-*jitter* a través de realimentación a nivel de radio en redes LTE

Tesis presentada en opción al grado de  
Ingeniero en Telecomunicaciones y Electrónica

Autor: Rigoberto Valentin López Fernández  
E-mail: rlfernandez@uclv.edu.cu

Tutor: M.Sc. Carlos A. Rodríguez López, Prof. Auxiliar  
Dpto. de Telecomunicaciones, Facultad de Ing. Eléctrica, UCLV  
E-mail: crodriguez@uclv.edu.cu

Ing. Meyli Lucy Guevara Martín, Prof. Instructor  
Dpto. de Telecomunicaciones, Facultad de Ing. Eléctrica, UCLV  
E-mail: meyli@edu.cu

Santa Clara

2015

“Año 57 de la Revolución”



Hago constar que el presente trabajo de diploma fue realizado en la Universidad Central "Marta Abreu" de Las Villas como parte de la culminación de estudios de Ingeniería en Telecomunicaciones y Electrónica, autorizando a que el mismo sea utilizado por la Institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos, ni publicados sin autorización de la Universidad.

---

Rigoberto Valentin López Fernández  
Autor

---

Fecha

Los abajo firmantes certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

---

Rigoberto Valentin López Fernández  
Autor

---

Fecha

---

Vitalio Alfonso Reguera, Dr.C  
Jefe del Departamento

---

Fecha

---

Responsable de Información Científico-Técnica

---

Fecha

## PENSAMIENTO

*“Toda la gama de la experiencia humana es tuya, y puedes disfrutarla, si decides aventurarte en territorios que no te ofrecen garantías.”*

*Wayne Dyer*

## DEDICATORIA

A mi mamá,  
*por su amor infinito, por traerme hasta aquí y ser mi ejemplo de fortaleza*

A mi papá,  
*por darme la mejor infancia que cualquier niño pudiera desear*

## AGRADECIMIENTOS

A mis hermanas, mi primo, mi tía y mi abuela, por quererme y siempre poder contar con ellos.

A Ama, mi otra mamá, por el inmenso cariño que me ha dado siempre y por estar en todo momento conmigo.

A Daisy, por compartir conmigo casi cuatro años en los que me ha brindado un gran amor y mucha felicidad.

A Mapi, por preocuparte siempre por mis estudios y ser como otra hermana para mí.

A mis tutores Carlos y Meyli, por su invaluable ayuda, dedicación y disposición.

A Pablo, por ayudarme en los últimos momentos de la tesis, así como a mis otros amigos del aula, en especial Roberto, Chencho, Julio, Carli, Tamara y Hugo por compartir estos cinco años.

A mis mejores amigos: Rosalí, Lucilo, María Antonieta, Barletta, Luzdaimi, Laura y Rosalia, por pasar junto a ellos muchos buenos momentos.

A todos los profesores que han contribuido a mi formación profesional.

A José Omar Espino, Denis Sánchez y José Miguel Delgado, trabajadores de la terminal de ómnibus de Cienfuegos, por facilitarme el transporte a la universidad prácticamente a mi antojo.

Santa Clara, Cuba, 2015

## TAREA TÉCNICA

Para dar cumplimiento a los objetivos propuestos para la culminación de esta tesis, se realizan las siguientes tareas:

1. Realizar una revisión bibliográfica sobre las características generales de la tecnología móvil celular LTE.
2. Identificar los eventos de radio cuya anticipación se pueda utilizar para mejorar el desempeño del *buffer* de de-*jitter*.
3. Estudiar el proceso de *handover* para su utilización en un simulador de eventos discretos.
4. Obtener trazas de voz sobre LTE mediante simulación para evaluar el comportamiento de los algoritmos de reproducción sobre la misma.
5. Diseñar un experimento utilizando simulación para comprobar el impacto de la realimentación a nivel de radio en el mecanismo de ajuste del *buffer*.
6. Discutir los resultados obtenidos en el experimento.
7. Elaborar el informe final del trabajo de diploma.

---

Rigoberto Valentin López Fernández  
Autor

---

M.Sc. Carlos A. Rodríguez López  
Tutor

---

Ing. Meyli Lucy Guevara Martín  
Tutor

## RESUMEN

En este trabajo se analiza la mejora del desempeño de los algoritmos de reproducción del *buffer* de *de-jitter* al utilizar realimentación a nivel de radio. Esta investigación se lleva a cabo debido a la posibilidad de impactar de manera positiva la calidad de servicio en aplicaciones de voz sobre LTE (VoLTE), dado que las mismas enfrentan estrictas restricciones de demora y pérdida de paquetes. Hasta el momento de escribir esta tesis, no se pudo constatar la existencia de un estudio de mecanismos de reproducción que tuviesen en cuenta la anticipación de la ocurrencia de radio eventos. Se obtiene mediante simulación trazas de VoLTE sobre las cuales se evalúa el comportamiento de los algoritmos de la manera tradicional y empleando anticipación, y se comprueba que la calidad percibida por el usuario aumenta en todos los casos cuando se emplea realimentación a nivel de radio.

## TABLA DE CONTENIDO

	<u>Página</u>
PENSAMIENTO . . . . .	I
DEDICATORIA . . . . .	II
AGRADECIMIENTOS . . . . .	III
TAREA TÉCNICA . . . . .	IV
RESUMEN . . . . .	V
ÍNDICE DE TABLAS . . . . .	IX
ÍNDICE DE FIGURAS . . . . .	X
INTRODUCCIÓN . . . . .	1
1. REVISIÓN BIBLIOGRÁFICA . . . . .	5
1.1. Necesidad de LTE . . . . .	5
1.2. Arquitectura de la red LTE . . . . .	6
1.2.1. Protocolos . . . . .	7
1.2.2. Interfaces S1 y X2 . . . . .	10
1.3. Planificación . . . . .	12
1.4. HARQ . . . . .	12
1.5. Gestión de sesión del Sistema de Paquetes Evolucionado (EPS) . . . . .	13
1.6. Calidad de servicio (QoS) . . . . .	14
1.6.1. Capacidad de voz . . . . .	15
1.6.2. Algoritmo de estimación de calidad de voz para redes que so- portan VoIP . . . . .	16
1.6.2.1. Métodos de medición no intrusivos . . . . .	17
1.6.2.2. Modelo E . . . . .	17
1.7. Algoritmos de control de demora . . . . .	19

1.8.	Realimentación a nivel de radio . . . . .	20
1.9.	Tipos de <i>handovers</i> (HO) en LTE . . . . .	23
1.9.1.	Configuración de mediciones . . . . .	23
1.9.2.	<i>Handover</i> X2 . . . . .	25
1.10.	Conclusiones parciales . . . . .	27
2.	MATERIALES Y MÉTODOS . . . . .	29
2.1.	Definición del simulador . . . . .	29
2.1.1.	NS-3 . . . . .	29
2.1.2.	LENA . . . . .	30
2.2.	Descripción del escenario . . . . .	30
2.2.1.	Topología . . . . .	30
2.2.2.	Modelo de movilidad . . . . .	30
2.2.3.	Modelo de pérdida de trayectos . . . . .	32
2.2.4.	Modelo de Desvanecimiento . . . . .	33
2.2.5.	Configuración del protocolo RLC . . . . .	33
2.2.6.	Configuración del protocolo RRC . . . . .	34
2.2.7.	Planificador . . . . .	34
2.2.8.	Algoritmo de <i>handover</i> . . . . .	35
2.2.9.	Potencia de transmisión. Modelo de antena. Ancho de banda . . . . .	36
2.2.10.	Modelo de aplicación . . . . .	36
2.3.	Recolección de las trazas . . . . .	37
2.4.	Algoritmos de reproducción . . . . .	38
2.4.1.	Algoritmos de Ramjee . . . . .	38
2.4.2.	Algoritmo 5 . . . . .	40
2.4.3.	Algoritmo eEM . . . . .	41
2.5.	Conclusiones parciales . . . . .	41
3.	RESULTADOS Y DISCUSIÓN . . . . .	43
3.1.	Trazas de VoLTE . . . . .	43
3.2.	Adición de silencios . . . . .	44
3.3.	Evaluación de los algoritmos de reproducción sin detección de picos . . . . .	45

3.4.	Evaluación de los algoritmos de reproducción con detección de picos	53
3.5.	Comparación del desempeño de los algoritmos: detección de picos vs. no detección de picos . . . . .	55
3.6.	Conclusiones parciales . . . . .	59
CONCLUSIONES . . . . .		60
RECOMENDACIONES . . . . .		62
REFERENCIAS BIBLIOGRÁFICAS . . . . .		65

## ÍNDICE DE TABLAS

<u>Tabla</u>		<u>Página</u>
1-1.	Relación del parámetro $R$ con el MOS . . . . .	18
2-1.	Valores asignados a los atributos del modelo SSRWMM . . . . .	32
3-1.	MOS instantáneo de las muestras 7, 8 y 9 de todos los algoritmos . . . . .	51
3-2.	Valores MOS promedio correspondientes a los seis algoritmos . . . . .	52
3-3.	Calidad instantánea (muestra 8 y 10) y calidad promedio de los seis algoritmos . . . . .	58

## ÍNDICE DE FIGURAS

<u>Figura</u>	<u>Página</u>
1-1. Arquitectura de red LTE. . . . .	7
1-2. Pila de protocolos de la interfaz física. . . . .	10
1-3. Esquema lógico de los protocolos de las interfaces S1 y X2. . . . .	11
1-4. Planificación dinámica y semi-persistente para voz. . . . .	12
1-5. Conexión de red de paquetes de datos incluyendo portadoras EPS. . . . .	14
1-6. Conexiones PDN con IMS y APNs de Internet. . . . .	14
1-7. Técnica no intrusiva. . . . .	17
1-8. Esquema funcional del Modelo E. . . . .	18
1-9. Factores para el cálculo de $R$ . . . . .	19
1-10. Configuración de medición básica utilizando una portadora. . . . .	24
1-11. Flujo de señalización de <i>handover</i> basado en X2, intra S-GW. . . . .	26
2-1. Distribución de los emplazamientos de eNodeBs. . . . .	30
2-2. Movimiento en zigzag del proceso RWM. . . . .	31
2-3. Porción de la traza fading_trace_EVA_60kmph.fad. . . . .	33
2-4. Comportamiento de planificación para dos usuarios con calidades promedio de canal diferentes utilizando el planificador de justicia proporcional. . . . .	35
2-5. Pseudocódigos para los algoritmos 1, 2, 3 y 4 de Ramjee . . . . .	40
2-6. Pseudocódigo del algoritmo 5. . . . .	41

2-7. Muestra de la estrategia de reproducción del algoritmo 6. . . . .	41
3-1. Traza de VoLTE obtenida mediante simulación. . . . .	44
3-2. Traza con períodos de silencio insertados. . . . .	45
3-3. Comportamiento de los 6 algoritmos de reproducción evaluados sin utilizar detección de picos. . . . .	45
3-4. Comportamiento del algoritmo 1 de Ramjee. . . . .	46
3-5. Amplificación de la figura 3-4. . . . .	46
3-6. Comportamiento del algoritmo 3 de Ramjee. . . . .	46
3-7. Amplificación de la figura 3-6. . . . .	47
3-8. Comportamiento del algoritmo eEM. . . . .	47
3-9. Amplificación de la figura 3-8. . . . .	47
3-10. Comportamiento del algoritmo 2 de Ramjee. . . . .	48
3-11. Comportamiento del algoritmo 4 de Ramjee. . . . .	49
3-12. Amplificación de la figura 3-11. . . . .	49
3-13. Comportamiento del algoritmo 5. . . . .	50
3-14. Amplificación de la figura 3-13. . . . .	50
3-15. MOS instantáneo para los 6 algoritmos evaluados. . . . .	51
3-16. MOS instantáneo para los 6 algoritmos evaluados. . . . .	52
3-17. Valor reportado de RSRP para celda de servicio y objetivo antes de un <i>handover</i> . . . . .	53
3-18. Desempeño de los algoritmos de reproducción utilizando detección de picos. . . . .	54
3-19. Desempeño del algoritmo 1 de Ramjee empleando detección de picos. . . . .	54
3-20. Desempeño del algoritmo 2 de Ramjee empleando detección de picos. . . . .	54

3-21.Desempeño del algoritmo 3 de Ramjee empleando detección de picos. . .	55
3-22.Desempeño del algoritmo 4 de Ramjee empleando detección de picos. . .	55
3-23.Desempeño del algoritmo 5 empleando detección de picos. . . . .	56
3-24.Desempeño del algoritmo eEM empleando detección de picos. . . . .	56
3-25.Comparación del valor instantáneo de la octava y décima muestra del MOS. . . . .	57
3-26.Comparación de la calidad percibida por el usuario para el algoritmo 5.	58

## INTRODUCCIÓN

Las redes inalámbricas desde sus primeros desarrollos comerciales en la década de los 80 se han afianzado en la preferencia de los usuarios debido, entre muchas otras cualidades, a la movilidad que les confiere, y son especialmente su demanda y popularidad lo que las hacen en la actualidad centro de investigación y continuo desarrollo. El indetenible avance en estas esferas se ha manifestado en prácticamente todas las implementaciones, sin embargo una sobre la que más impacta la evolución tecnológica es el sistema celular (Ati and Geovanny, 2011).

El sistema celular surgió en sus inicios con el fin de proveer telefonía móvil, definiéndose como tal a aquel que permite la comunicación entre usuarios que se desplazan libremente en lugares geográficos diferentes, en estos sistemas un transmisor cubre una zona geográfica denominada celda, las cuales son las unidades básica de la cobertura del sistema (Ati and Geovanny, 2011). En nuestros días no se puede decir que brindar telefonía sea la única finalidad de los sistemas celulares, a medida que las redes han ido evolucionando y los nuevos servicios para los usuarios han entrado en el panorama social, los sistemas celulares han ido creando nuevas capacidades con el objetivo de integrarse y ser parte activa.

El sistema celular utilizó la infraestructura de la Red Telefónica Pública Conmutada (*Public Switched Telephone Network*, PSTN), que emplea conmutación de circuitos y sistemas de señalización, para la implementación de los servicios de voz; así fue desde la primera generación que estaba basada en tecnología analógica, donde sus servicios eran únicamente de voz y se caracterizaban por su baja calidad y velocidad, y continuó en las generaciones que le sucedieron. En los años 90 hizo su aparición la segunda generación de telefonía celular basada esta vez totalmente en tecnología digital. Esta nueva generación poseía novedad en estándares como el Sistema Global para Comunicaciones Móviles (*Global System for Mobile Communications*, GSM), y el Acceso Múltiple por División del Código (*Code Division Multiple Access*, CDMA) que hicieron posible que a partir de aquí existiera no solo la comunicación telefónica, sino también el envío de mensajes de texto y el *roaming* internacional, brindándole a los usuarios comunicaciones de alta calidad de voz, bajo costo de terminales, flexibilidad y capacidad para migración, bajo el mismo principio de conmutación de circuitos (Ati and Geovanny, 2011). Sin embargo,

con el amplio esparcimiento del uso de Internet, los servicios de conmutación de paquetes fueron introducidos con el fin de mejorar el sistema celular. Con el surgimiento del estándar Servicio de Radio de Paquetes General (*General Packet Radio Service*, GPRS), el que constituyó una extensión de la tecnología GSM, se realizó por primera vez la implementación de envío de paquetes de información por la red celular, posibilitando el establecimiento para los usuarios de servicios como la mensajería multimedia, el correo electrónico y el acceso a Internet; de esta manera el sistema celular constaba con tecnología de conmutación de paquetes para transmisión de datos. Las posteriores evoluciones del sistema celular se han mantenido sobre este mismo principio y todas las variaciones que se han realizado hasta la implementación de Acceso de Paquetes de Alta Velocidad (*High Speed Packet Access*, HSPA), relativa a la generación 3.5, han ido encaminadas, entre otros factores, a mejorar las capacidades de estos canales de datos con el objetivo de soportar las demandas de las aplicaciones disponibles para los usuarios; de esta manera estas generaciones constaban con ambos tipos de conmutación, por circuitos para la voz y por paquetes para los datos (Ahmad, 2005).

En diciembre de 2009 vio la luz el primer estándar Evolución de Largo Plazo (*Long Term Evolution*, LTE) el cual devendría la primera versión de la que comercialmente se conoce como cuarta generación de sistemas celulares. Esta tecnología fue estandarizada por el Proyecto de Asociación de Tercera Generación (*3rd Generation Partnership Project*, 3GPP) y constituye una evolución del Sistema de Comunicaciones Móviles Universal (*Universal Mobile Telecommunication System*, UMTS) (MAHMUD, 2013). Los estándares de Long Term Evolution (LTE)/evolved packet core (EPC) han sido diseñados como una tecnología móvil basados puramente en IP. Desde la perspectiva del despliegue LTE, el servicio de voz puede ser considerado como una aplicación de datos, pero con requerimientos específicos para tráfico en tiempo real, calidad de servicio e interoperación con la infraestructura de voz existente (núcleos de red de circuitos conmutados) (Poikselkä *et al.*, 2012).

El desempeño de aplicaciones en redes inalámbricas es parcialmente dependiente de la configuración del enlace. Las características del enlace varían con la persistencia de retransmisión de tramas, demora de retransmisión de tramas, estrategias de modulación adaptativa, codificación, y más. La configuración del enlace y las condiciones del canal pueden conducir a pérdidas de paquetes, demora y variaciones de la demora (Alfredsson *et al.*, 2008).

Para enfrentar las variaciones de demora inherentes de redes IP, se utiliza en redes LTE un *buffer* de eliminación de demora o *buffer* de de-jitter en los equipos de usuarios

(UE). Este *buffer* almacena los paquetes que arriban, los cuales no siempre llegan en orden y, luego de recuperar el sincronismo, los reproduce.

El *buffer* de de-*jitter* en un receptor de voz sobre LTE (VoLTE) puede utilizar eventos de radio, cuya ocurrencia puede anticipar, para mejorar su desempeño y ajustarse ante los mismos sin que ocurra una elevada pérdida de paquetes y sin introducir una demora excesiva, lo cual repercute positivamente en la calidad de servicio percibida por el usuario. Hasta el momento de escribir esta tesis, no se pudo constatar la existencia de un estudio de mecanismos de reproducción que tuviesen en cuenta la anticipación de la ocurrencia de radio eventos.

Teniendo en cuenta lo planteado anteriormente, se define la siguiente interrogante científica: ¿Cómo se puede mejorar el mecanismo de ajuste del *buffer* de eliminación de *jitter* mediante el conocimiento previo de eventos de radio en redes LTE?

Para responder a esta interrogante se plantea como objetivo general:

- Proponer un mecanismo de ajuste del *buffer* de reproducción que tenga en cuenta el conocimiento previo de eventos de radio.

Para organizar el estudio y dar cumplimiento al objetivo general, se definen los siguientes objetivos específicos:

1. Realizar un estudio bibliográfico de las tecnologías involucradas en la investigación.
2. Describir los eventos de radio que influyen en el comportamiento del *jitter* cuyo conocimiento previo pueda ser usado para mejorar el mecanismo de ajuste del *buffer* de de-*jitter*.
3. Diseñar un experimento mediante simulación que permita evaluar la mejora del mecanismo de ajuste.
4. Presentar un informe donde se exponga una valoración crítica de los resultados obtenidos.

**Organización del informe:** El informe de la investigación se organiza de la siguiente forma: resumen, introducción, capitulario, conclusiones y recomendaciones, referencias bibliográficas.

**Introducción:** Se realiza una reseña donde se define la necesidad, actualidad e importancia del tema que se aborda, así como la situación problemática y objetivos del trabajo.

**Capítulo 1:** Se describen las principales características de una red LTE, los requerimientos de calidad de los servicios de VoIP, el modelo utilizado para la evaluación de la calidad y los radio eventos, en particular el *handover*.

**Capítulo 2:** Se exponen las principales características del simulador NS-3, se describen los parámetros y los valores asignados a estos en las simulaciones realizadas y se hace una descripción del funcionamiento de los seis algoritmos de *buffer* de reproducción tratados en la tesis.

**Capítulo 3:** Se realiza una comparación del comportamiento de los algoritmos en términos de calidad percibida por el usuario en situaciones donde no se emplea la realimentación a nivel de radio y en el caso contrario.

**Conclusiones:** Se describen los resultados obtenidos a partir de los objetivos trazados inicialmente.

# CAPÍTULO 1

## REVISIÓN BIBLIOGRÁFICA

En este capítulo se realiza un análisis bibliográfico de las redes LTE, haciendo énfasis en su arquitectura, protocolos de la interfaz de radio y requerimientos de las aplicaciones de voz sobre LTE, entre otros. Se exponen los eventos de radio cuya anticipación se puede utilizar para mejorar la calidad, eligiéndose el *handover* como evento a estudiar en la tesis.

### 1.1. Necesidad de LTE

La evolución de sistemas 3G a 4G es necesario debido a la creación y desarrollo de nuevos servicios para dispositivos móviles, y es posible gracias al avance de la tecnología disponible para estos sistemas. Además, también ha ocurrido una evolución del ambiente en el que los sistemas móviles son desplegados y operados, en términos de competencia entre operadores de otras tecnologías móviles, y nuevas regulaciones del uso del espectro y aspectos del mercado (Dahlman *et al.*, 2013).

La rápida evolución de tecnologías usadas en sistemas de telecomunicaciones, electrónica de consumo, y específicamente en dispositivos móviles ha sido notable en los últimos 20 años. La telefonía fija (POTS) y las generaciones anteriores de tecnología móvil fueron construidas para servicios de circuitos conmutados, principalmente voz. Los primeros servicios de datos sobre GSM fueron de conmutación de circuitos, con GPRS basado en paquetes en una adición posterior. Esto también influyó el desarrollo de 3G, el cual estaba basado en transmisión de datos por conmutación de circuitos, con servicios de conmutación de paquetes como un agregado (Dahlman *et al.*, 2013).

No fue hasta la evolución de 3G en HSPA y luego en LTE/LTE-*Advanced* (Evolución de Largo Plazo) que los servicios por conmutación de paquetes e IP fueron considerados el principal objetivo de diseño (Dahlman *et al.*, 2013).

Con una mayor utilización del espectro para servicios móviles de banda ancha, existe una necesidad de operar sistemas móviles en diferentes bandas de frecuencias, en asignaciones del espectro de diferentes tamaños y a veces en espectro fragmentado. Esto requiere una alta flexibilidad del espectro con la posibilidad de un ancho de banda del

canal variante, lo cual fue uno de los parámetros impulsores y esenciales en el diseño de LTE (Dahlman *et al.*, 2013). La demanda de nuevos servicios móviles y la evolución de la interfaz de radio a LTE han causado la evolución del núcleo de la red. El núcleo desarrollado para GSM en la década de 1980 fue extendido para soportar GPRS, EDGE, y WCDMA en la década de 1990, pero aún estaba la mayor parte desarrollada sobre el dominio de conmutación de circuitos. Se inició una Evolución de la Arquitectura del Sistema (SAE) al mismo tiempo que comenzó el desarrollo de LTE y resultó en un Núcleo de Paquetes Evolucionado (*Evolved Packet Core*, EPC), desarrollado para soportar HSPA y LTE/LTE-*Advanced*, concentrándose en el dominio de conmutación de paquetes (Dahlman *et al.*, 2013).

Además de lo mencionado anteriormente, un operador de 2G o 3G tiene que mantener dos núcleos de red: el dominio de circuitos conmutados para voz, y el dominio de paquetes conmutados para datos. Sin embargo, también es posible transportar llamadas de voz sobre redes de paquetes conmutados usando técnicas como VoIP. Haciendo esto, los operadores pueden mover todo al dominio de paquetes conmutados, reduciendo sus gastos operacionales y capitales (Cox, 2012).

## 1.2. Arquitectura de la red LTE

La arquitectura de red LTE está diseñada con el objetivo de soportar tráfico de paquetes conmutados con movilidad sin irregularidades, calidad de servicio (*Quality of Service*, QoS) y latencia mínima. Una aproximación de paquetes conmutados permite soportar todos los servicios incluyendo voz a través de conexiones de paquetes. Resulta una arquitectura plana altamente simplificada con solo dos tipos de nodos llamados Node-B evolucionado (eNB o eNodeB) y entidad de gestión de movilidad/pasarela (*Mobility Management Entity/Gateway*, MME/GW). Alguno de los beneficios de tener un solo nodo en la red de acceso son reducción de la latencia y distribución de la carga de procesamiento entre múltiples eNBs (Farooq, 2009).

Todas las interfaces de red están basadas en protocolos IP. Las eNBs se interconectan a través de la interfaz X2 y a la entidad MME/GW a través de la interfaz S1 como se muestra en la figura 1-1. Se definen dos pasarelas lógicas llamadas Pasarela de Servicio (*Serving Gateway*, S-GW) y Pasarela de Red de Paquetes de Datos (*Packet Data Network*, P-GW). La S-GW actúa como un ancla de movilidad local enviando y recibiendo paquetes hacia y desde el equipo de usuario (*User Equipment*, UE) y la eNB. La P-GW sirve de interfaz con redes de paquetes de datos externas como la Internet y el Subsistema Multimedia IP (*IP Multimedia Subsystem*, IMS). La P-GW también desempeña varias funciones IP como asignación de direcciones, filtrado de paquetes y enrutamiento. La MME es una entidad solo para señalización y por

tanto los paquetes IP de usuario no pasan por esta. Una ventaja de una entidad de señalización separada es que la capacidad de la red para señalización y tráfico pueden crecer independientemente. Las principales funciones de la MME son gestión de la lista del área de rastreo, *roaming*, autenticación, autorización, selección de P-GW/S-GW, gestión de portadores incluyendo el establecimiento de portadores dedicados (Farooq, 2009).

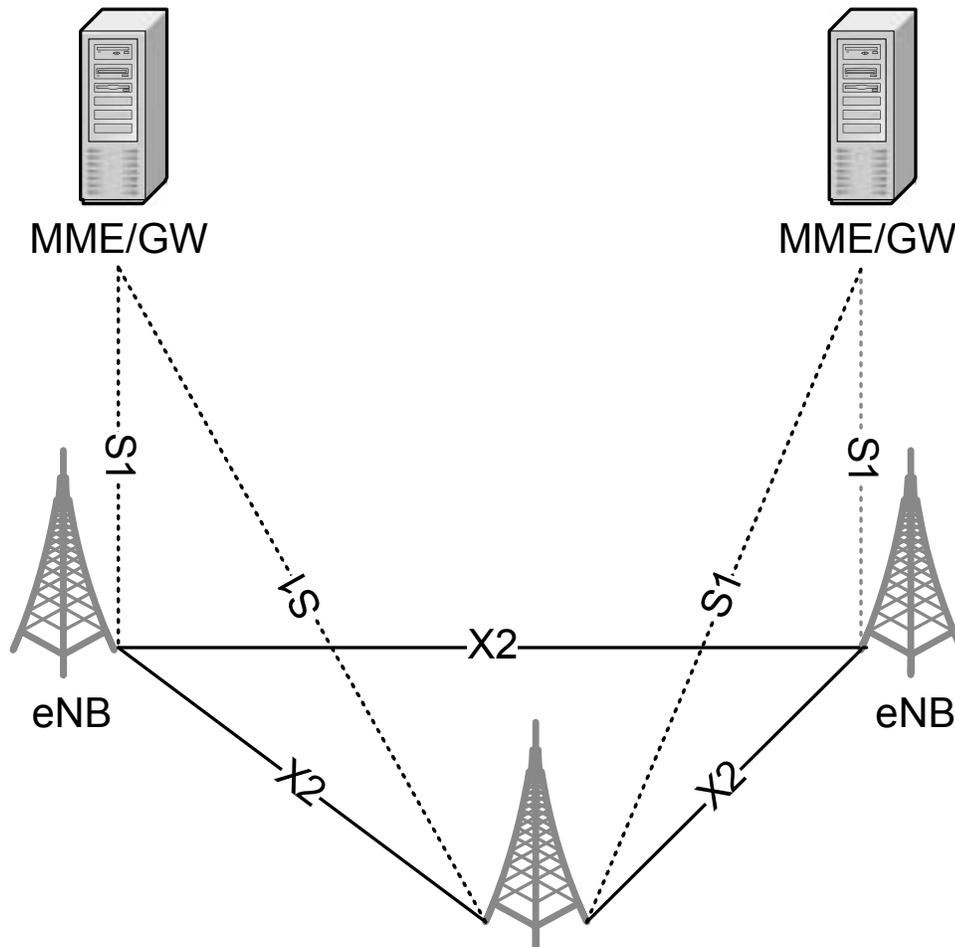


Figura 1-1: Arquitectura de red LTE (Dahlman *et al.*, 2013).

### 1.2.1. Protocolos

Las torres de protocolos utilizadas en las tres interfaces de E-UTRAN (radio, S1 y X2) se estructuran en torno a un plano de usuario y un plano de control. El plano de usuario abarca los protocolos utilizados para el envío del tráfico (paquetes IP) correspondiente a los servicios a los que acceden los terminales a través de la red. El plano de control se refiere a los protocolos necesarios para sustentar las funciones y procedimientos en las diferentes interfaces (Dahlman *et al.*, 2013).

#### Interfaz de radio

El envío de paquetes IP entre la eNB y un equipo de usuario a través de la interfaz radio, concentrado en el plano de usuario, se sustenta en una torre de protocolos formada por una capa de enlace (o capa de nivel 2) y una capa física. La capa física es la encargada de realizar la transmisión, propiamente dicha, a través del canal radio. La capa de enlace se desglosa a su vez en tres subcapas:

- Subcapa Protocolo de Convergencia de Paquetes de Datos (*Packet Data Convergence Protocol*, PDCP), esta constituye la capa superior de la torre de protocolos encargada de proporcionar el punto de acceso al servicio portador radio (*Radio Bearer*, RB), o sea los paquetes IP del tráfico de usuario se entregan y se reciben a través del servicio de transferencia proporcionado por esta capa (Dahlman *et al.*, 2013).
- Subcapa Control del Enlace de Radio (*Radio Link Control*, RLC) este permite enviar de forma fiable los paquetes PDCP entre el eNB y equipo de usuario. Las funcionalidades básicas de esta capa son: implementar procedimientos de segmentación/ concatenación de los paquetes IP recibidos de capas superiores para adaptar su tamaño a las capacidades de transmisión de la interfaz aire; implementar mecanismos de retransmisión de los paquetes recibidos erróneamente y así como evitar duplicidades y realizar la entrega ordenada a las capas superiores (Dahlman *et al.*, 2013).

Existe un mapeo directo entre los flujos de información y/o control recibidos por la subcapa RLC y los canales lógicos definidos entre las capas RLC y MAC.

Existen tres modos de operación de la subcapa RLC: el modo transparente, que es el más simple ya que la subcapa RLC entrega directamente a la subcapa MAC los SDU RLC recibidos de las capas superiores y no se introducen cabeceras al mensaje ni se implementan procesos de retransmisión; el modo de no-reconocimiento, que detecta si los paquetes SDU RLC (*Service Data Units* recibidos contienen errores, pero no implementa mecanismos de retransmisión, y el modo de reconocimiento que es el modo de operación más robusto, donde además de las funciones de segmentación/reensamblado y de ordenación de los SDU RLC también proporciona mecanismos de retransmisión tipo ARQ para eliminar posibles errores introducidos por el canal radio (Iñiguez~Chavarría *et al.*, 2014).

- Subcapa Control de Acceso al Medio (*Medium Access Control*, MAC): esta es la subcapa encargada de controlar el acceso al canal radio, para lo que soporta varias funciones de planificación dinámica entre equipos de usuario atendiendo a prioridades establecidas por la calidad demandada en cada una de ellas, multiplexa los paquetes RLC de diferentes servicios portadores radio en los canales de transporte ofrecidos por la capa física y realiza un control de errores mediante el mecanismo de Solicitud de Retransmisiones Automáticas Híbrido (*Hybrid Automatic Repeat Request*, HARQ). Cada

capa/subcapa de la torre de protocolos se ocupa de un conjunto de funciones concreto y define el formato de los paquetes de datos que se intercambian entre entidades remotas (Dahlman *et al.*, 2013).

Respecto al plano de control entre el equipo de usuario y la red, éste se soporta sobre la misma capa de enlace (protocolos PDCP, RLC, MAC) y la misma capa física utilizadas en el plano de usuario. Los protocolos de nivel de red específicos de este plano son:

- Subcapa Control de Recursos de Radio (*Radio Resource Control*, RRC): permite establecer una conexión de control entre la eNB y un equipo de usuario a través de la cual se llevan a cabo un número importante de funciones relacionadas con la gestión de la operativa de la interfaz radio (Dahlman *et al.*, 2013). Este protocolo es el encargado de controlar el comportamiento del móvil cuando está en el modo de operación “estado conectado”, mientras que especifica la señalización de aviso (*paging*) y del Sistema de Información cuando el sistema todavía no tiene establecidos los servicios portadores radio (“estado inactivo”) (Iñiguez~Chavarría *et al.*, 2014). El protocolo RRC habilita la transmisión de información de no acceso a capa (*Non-Access Stratum*, NAS) común que es aplicable a todos los UE, así como la información NAS dedicada, aplicada solo a UE particulares. El protocolo RRC cubre todos los procedimientos relacionados con el establecimiento, modificación y liberación de una conexión RRC, incluyendo el aviso, activación inicial segura, establecimiento de los portadores de radio de señalización (*Signaling Radio Bearers*, SRB) y los portadores de radio que portan la información del usuario (*Data Radio Bearers*, DRB) (Agusti~Comes *et al.*, 2010). El protocolo RRC también soporta una serie de funciones relacionadas con la movilidad del usuario final que se encuentre en el estado conectado. Entre estas se encuentra el control de mediciones, el que se refiere a la configuración con que debe proceder el terminal así como el método para realizar los reportes a la eNB (Iñiguez~Chavarría *et al.*, 2014).

Como ha sido mencionado anteriormente, el protocolo establece dos estados para el terminal: el estado RRC-conectado y el estado RRC-inactivo. En el estado inactivo, creado para ahorrar tanta energía como sea posible, aunque a nivel de aplicación el usuario está en este modo, en la interfaz de radio ocurren procesos en los que este está envuelto. De esta manera el terminal tendrá que decodificar la información del sistema que se transmite en el canal de difusión, además realizará la decodificación de los mensajes de aviso o *paging* de modo que se pueda conectar a la red en caso de que surja una nueva sesión para él. Por último el terminal en estado inactivo debe implementar la reelección de celdas, donde periódicamente serán evaluadas las celdas con el objetivo de pertenecer a la que mejores condiciones posea. En el estado conectado existe una conexión activa entre el terminal y la eNB, en este caso ambos lados intercambiarán

datos de usuario o mensajes de señalización sobre canales lógicos (Iñiguez~Chavarría *et al.*, 2014).

- Señalización de los protocolos NAS: los protocolos NAS se extienden entre la entidad de red MME en la red troncal y el equipo de usuario. Los mensajes de estos protocolos se transportan de forma transparente en la interfaz, radio encapsulados dentro de la parte de datos de los mensajes RRC. Las principales funciones de los protocolos NAS son: autenticación, autorización, gestión de movilidad de los terminales que no tienen una conexión RRC establecida y gestión de los servicios portadores de la red EPS (Dahlman *et al.*, 2013).

La siguiente figura muestra la pila de protocolos de esta interfaz representada de forma esquemática.

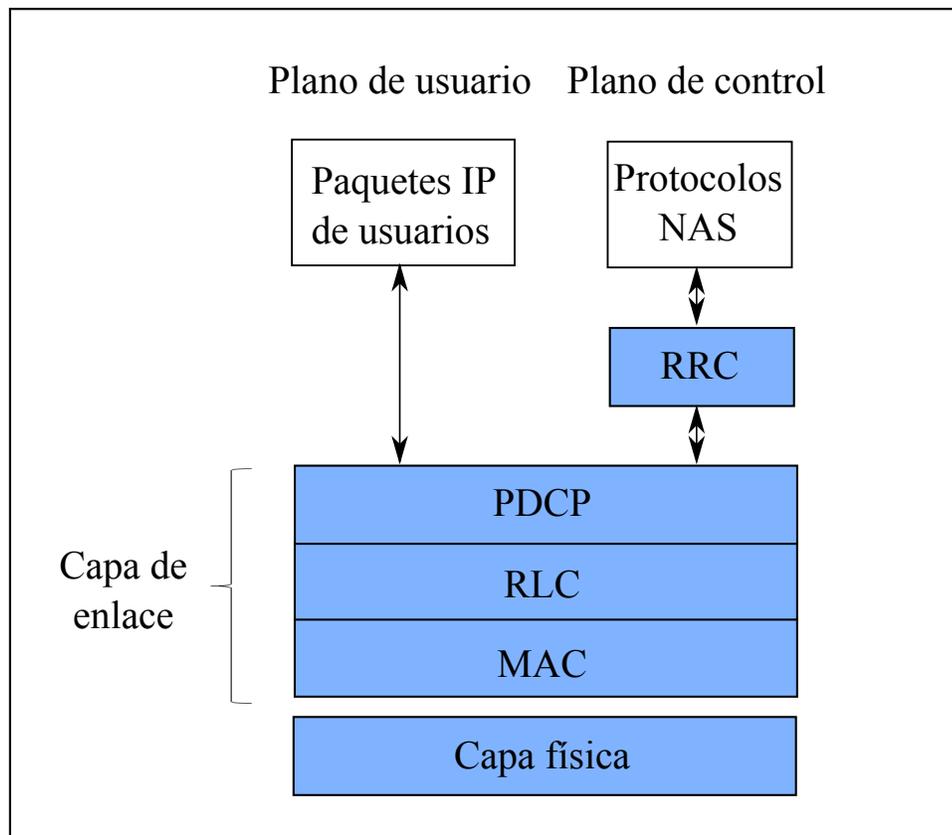


Figura 1–2: Pila de protocolos de la interfaz física (Agusti~Comes *et al.*, 2010).

### 1.2.2. Interfaces S1 y X2

La estructura de protocolos utilizada en E-UTRAN para soportar las interfaces S1 y X2 establece una separación entre la capa de red radio (*Radio Network Layer*, RNL) y la capa de red de transporte (*Transport Network Layer*, TNL). Esta descomposición tiene como objetivo aislar las funciones que son específicas del sistema de comunicaciones

móviles de aquellas otras que dependen de la tecnología de transporte utilizada como IP. De esta forma, los protocolos específicos de la red de acceso de radio constituyen la capa RNL, mientras que la capa TNL alberga los protocolos utilizados para el transporte de la información de la capa RNL entre las entidades de la red (Agusti~Comes *et al.*, 2010).

Tanto el plano de usuario de la interfaz S1 (S1-U) como el de la interfaz X2 utilizan el protocolo de encapsulado GPRS (*GPRS Tunneling Protocol - User Plane*, GTP-U) para el envío de paquetes IP de usuario, el protocolo GTP-U se transporta sobre UDP/IP y fundamentalmente se utiliza para multiplexar los paquetes IP de múltiples usuarios (Agusti~Comes *et al.*, 2010).

Respecto al plano de control de las interfaces S1 y X2, la capa de red de radio consiste en los protocolos S1-AP (S1 - *Application Part*) y X2-AP (*X2 Application Part*) respectivamente (Agusti~Comes *et al.*, 2010).

La transferencia de los mensajes de señalización del protocolo S1-AP entre eNBs y MMEs y X2-AP se realiza mediante el servicio de transferencia fiable que ofrece el Protocolo de Transporte de Control de Flujo (*Stream Control Transmission Protocol*, SCTP). Este es un protocolo de transporte (al igual que otros protocolos como TCP y UDP) de propósito general estandarizado por la Fuerza de Trabajo de Ingeniería de Internet (*Internet Engineering Task Force*, IETF) en (RFC 4960, 2007) que fue concebido originariamente para el envío de señalización de redes telefónicas sobre redes IP. SCTP hereda muchas de las funciones contempladas en TCP a la vez que introduce importantes mejoras encaminadas a proporcionar mayor robustez y versatilidad en la transferencia de diferentes tipos de información (Agusti~Comes *et al.*, 2010). La figura 1–3 muestra un esquema lógico de los protocolos anteriormente mencionados.

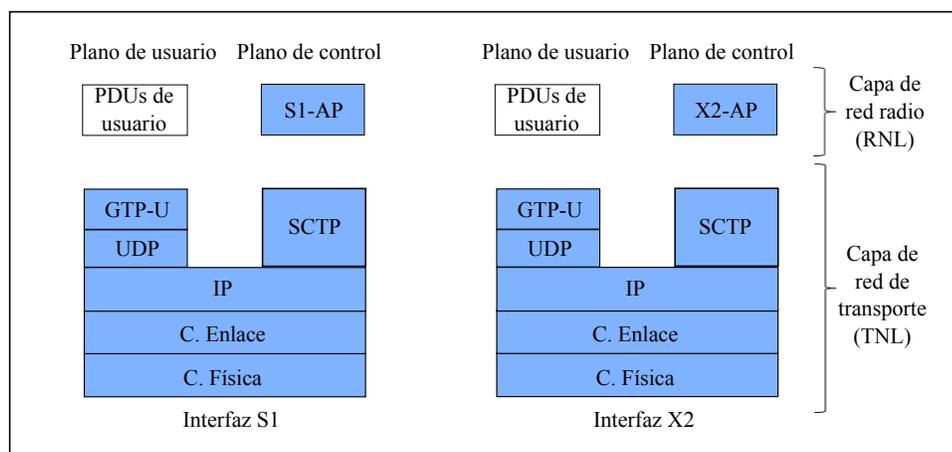


Figura 1–3: Esquema lógico de los protocolos de las interfaces S1 y X2 (Agusti~Comes *et al.*, 2010).

### 1.3. Planificación

La función de planificación se encuentra en el planificador de red, y las decisiones de planificación son señalizadas al UE. El planificador en el eNodeB realiza la asignación de bloques de recursos (*Resource Blocks*, RB) a bloques de transporte individuales en una subtrama y la selección de la razón de la modulación y codificación para bloques de transporte (Rumney *et al.*, 2013).

Se definen varios medios para señalar las decisiones de planificación al UE. El método más flexible es el de planificación dinámica en el cual se envía una asignación del DL para cada canal compartido del DL (*DL Shared Channel*, DL-SCH) y para cada transmisión del canal compartido del UL (*UL Shared Channel*, UL-SCH) la eNodeB envía un consentimiento del UL. Aunque este esquema provee flexibilidad total en cada subtrama, puede generar información de control excesiva (Rumney *et al.*, 2013).

Un método más eficiente es la planificación semi-persistente en el que una asignación del DL o un consentimiento del UL son válidos para N transmisiones o recepciones en lugar de ser válido para una sola transmisión o recepción. El valor de N es configurado y señalizado por el RRC. El resultado reduce la información de control pero también la flexibilidad. Otra opción es la agrupación de intervalos de tiempo de transmisión (*Transmission Time Interval*, TTI), en el UL en el cual se combinan un número de subtramas y solo una realimentación HARQ es enviada para todo el grupo. Esto es especialmente útil para limitar la potencia requerida para la transmisión, por ejemplo, en el borde de la celda (Rumney *et al.*, 2013).

En la figura 1–4 se muestra la planificación dinámica y la semi-persistente para voz.

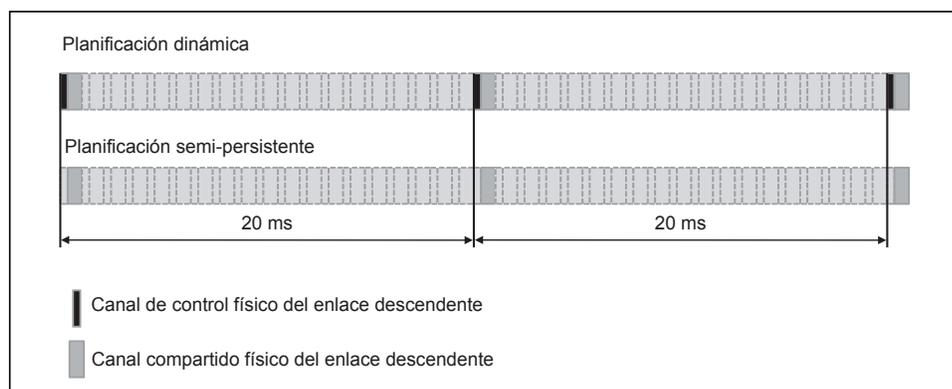


Figura 1–4: Planificación dinámica y semi-persistente para voz (Poikselkä *et al.*, 2012).

### 1.4. HARQ

Las transmisiones sobre canales inalámbricos están sujetas a errores, por ejemplo, debido a variaciones en la calidad de la señal recibida. Por lo tanto, la mayoría de los sistemas

de comunicaciones inalámbricos emplean alguna forma de Corrección de Error hacia Adelante (*Forward Error Correction*, FEC). El principio básico detrás de la codificación FEC es introducir redundancia en la señal transmitida. Esto se logra añadiendo bits de paridad a los bits de información antes de la transmisión (Dahlman *et al.*, 2013).

Otra aproximación para manejar errores de transmisión es la Solicitud de Repetición Automática (*Automatic Repeat Request*, ARQ). En un esquema ARQ, el receptor utiliza un código de detección de errores, típicamente un Chequeo de Redundancia Cíclico (*Cyclic Redundancy Check*, CRC), para detectar si el paquete recibido tiene errores o no. Si no se detectan errores en el paquete recibido, se declara los datos recibidos como libres de errores y se notifica al transmisor enviando un reconocimiento positivo. Por otra parte, si se detecta un error, el receptor descarta los datos recibidos y notifica al transmisor enviando un reconocimiento negativo. En respuesta a este reconocimiento, el transmisor retransmite la misma información (Dahlman *et al.*, 2013).

Virtualmente todos los sistemas de comunicaciones modernos, emplean una combinación de FEC y ARQ, conocida como HARQ. HARQ utiliza codificación FEC para corregir un subconjunto de todos los errores y confía en la detección de errores para detectar errores incorregibles. Los paquetes recibidos erróneamente son descartados y el receptor solicita retransmisiones de los mismos (Dahlman *et al.*, 2013).

La operación HARQ descrita anteriormente descarta paquetes con errores y solicita retransmisiones. Sin embargo, a pesar de no ser posible decodificar el paquete, la señal recibida aún contiene información, la cual se pierde al descartar el paquete. Este defecto se evita con HARQ con *soft combining*. En HARQ con *soft combining*, los paquetes recibidos con errores son almacenados en una memoria *buffer* y luego combinados con las retransmisiones para obtener un único y combinado paquete que es más confiable que sus constituyentes. La decodificación del código de corrección de errores opera sobre la señal combinada. Si la decodificación falla, se solicita una retransmisión (Dahlman *et al.*, 2013).

### 1.5. Gestión de sesión del Sistema de Paquetes Evolucionado (EPS)

La conectividad IP entre un UE y una red de paquetes de datos externa de una red móvil pública terrestre (*Public Land Mobile Network*, PLMN) se conoce como servicio de conectividad PDN. Cada conexión PDN está compuesta por una o más portadoras que transportan los datos sobre el EPS como se muestra en la figura 1–5. Se establece una portadora EPS cuando el UE se conecta a una PDN y permanece establecida todo el tiempo que dure la conexión para brindar al UE conectividad IP siempre encendido. La portadora EPS es conocida como la portadora por defecto. Cualquier portadora

adicional que se establezca para la misma conexión PDN se conoce como portadora dedicada (Poikselkä *et al.*, 2012).

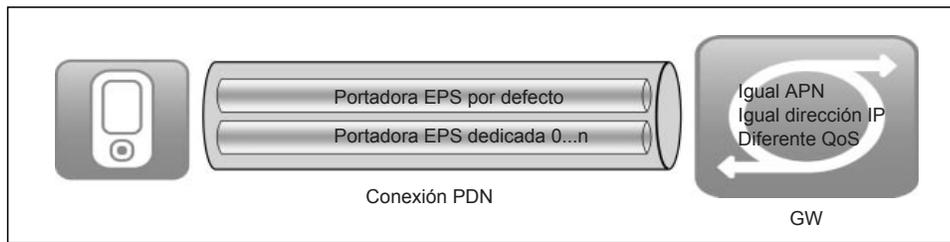


Figura 1–5: Conexión de red de paquetes de datos incluyendo portadoras EPS (Poikselkä *et al.*, 2012).

El UE enruta paquetes del UL a diferentes portadoras EPS basándose en filtros para paquetes del UL en las plantillas de flujo de tráfico (*Traffic Flow Template*, TFT) asignadas por la P-GW. La P-GW enruta paquetes del DL a diferentes portadoras EPS basándose en filtros para paquetes del DL en las plantillas de flujo de tráfico. Para cada portadora hay un túnel GTP-U entre la eNodeB y la S-GW, y entre la S-GW y P-GW (Poikselkä *et al.*, 2012).

Cuando se ofrecen servicios vía múltiples nombres de punto de acceso (*Access Point Name*, APN), se crean una o más portadoras por defecto adicionales con nuevas direcciones IP. La figura 1–6 muestra un ejemplo donde el UE tiene dos conexiones PDN simultáneas, una para la IMS y otra para Internet (Poikselkä *et al.*, 2012).

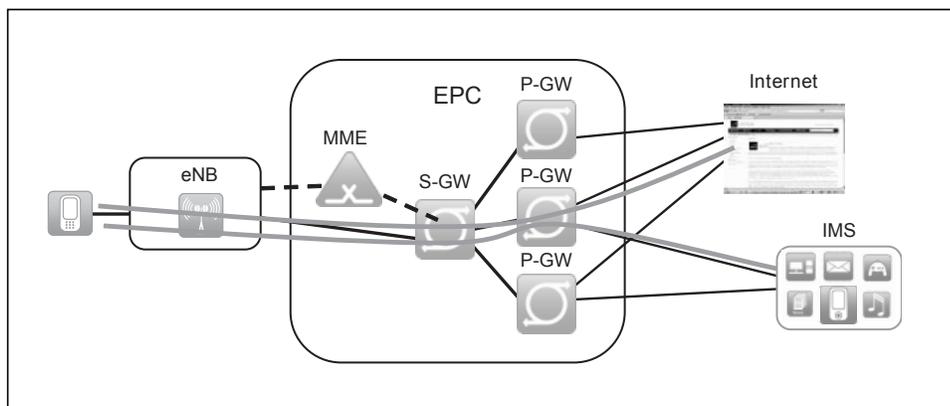


Figura 1–6: Conexiones PDN con IMS y APNs de Internet (Poikselkä *et al.*, 2012).

## 1.6. Calidad de servicio (QoS)

El EPS provee calidad de servicio a nivel de portadora, lo que significa que todo el tráfico mapeado a la misma portadora EPS recibe el mismo tratamiento de transferencia de paquetes a nivel de portadora. Para brindar diferente tratamiento de transferencia de

paquetes a nivel de portadora se requiere portadoras EPS separadas. La portadora EPS identifica excepcionalmente flujos de tráfico que reciben un tratamiento de calidad de servicio común entre el UE y una P-GW. La TFT de la portadora EPS es el conjunto de filtros de todos los paquetes asociados con esa portadora EPS (Poikselkä *et al.*, 2012).

Para propósitos de diferenciación de QoS se crea una portadora dedicada que puede ser de razón de bit garantizada (*Guaranteed Bit Rate*, GBR) o no garantizada. La(s) dirección(es) IP y APN asignadas para la portadora por defecto son utilizadas por las portadoras dedicadas dentro de la misma conexión PDN. La creación de una portadora dedicada es iniciada por la red, pero puede ser provocada por solicitudes de recursos por el UE (Poikselkä *et al.*, 2012).

La diferencia entre portadoras por defecto y dedicadas es transparente para la eNodeB. Una portadora EPS es referida como portadora GBR si los recursos de red dedicados a un valor GBR asociado con la portadora EPS son asignados permanentemente en el establecimiento/modificación de la portadora. De otra forma, una portadora EPS es referida como portadora no GBR (Poikselkä *et al.*, 2012).

Para cada portadora EPS (por defecto y dedicada), el soporte de QoS se basa en los siguientes parámetros (Poikselkä *et al.*, 2012):

- Identificador de clase de QoS (*QoS Class Identifier*, QCI): Escalar que se utiliza como referencia para acceder a parámetros específicos de nodos que controlan el tratamiento de transferencia de paquetes a nivel de portadora.
- Prioridad de asignación y retención (*Allocation and Retention Priority*, ARP): Contiene información acerca del nivel de prioridad (escalar), capacidad de derecho de prioridad (bandera) y vulnerabilidad de derecho de prioridad (bandera). El propósito fundamental de ARP es decidir si una solicitud de establecimiento/modificación de portadora puede ser aceptada o necesita ser rechazada debido a limitaciones de recursos.
- GBR: Denota la razón de bit que se espera sea provista por una portadora EPS.
- Máxima razón de bit (*Maximum Bit Rate*, MBR): Limita la razón de bit que se espera sea provista por una portadora EPS.

### 1.6.1. Capacidad de voz

A diferencia del tráfico de cola total (como descarga de archivos), el cual típicamente tolera la demora y no requiere una razón de bit garantizada, las aplicaciones de tiempo real como VoIP tienen restricciones de demora y de variación de la demora (*jitter*) muy estrictas. Es importante establecer los requerimientos de capacidad del sistema para

estos servicios, lo cual es un desafío en particular en sistemas basados totalmente en paquetes como LTE, el que depende de planificación adaptativa (Sesia *et al.*, 2009).

El requerimiento de capacidad del sistema se define como el número de usuarios de VoIP satisfechos, dados un modelo de tráfico particular y restricciones de demora. Los detalles del modelo de tráfico utilizado para evaluar LTE pueden ser encontrados en (Orange *et al.*, n.d.). Aquí, un usuario de VoIP es considerado que está en desconexión (no satisfecho) si más del 2% de los paquetes de VoIP no arriban satisfactoriamente al receptor de radio dentro de 50 ms y por lo tanto son descartados. Se asume una demora extremo a extremo total (de terminal móvil a terminal móvil) por debajo de 200 ms. La capacidad del sistema para VoIP puede entonces ser definida como el número de usuarios presentes por celda cuando más del 95% de los usuarios se encuentran satisfechos (Sesia *et al.*, 2009).

### 1.6.2. Algoritmo de estimación de calidad de voz para redes que soportan VoIP

Para medir la calidad de voz percibida por el usuario, existen dos tipos de métodos utilizados: los métodos subjetivos y objetivos. Según los métodos subjetivos, la calidad de la voz se establece a través de la opinión del usuario. La calidad de audio puede ser evaluada directamente (*Absolute Category Rating*, ACR), o en forma comparativa contra un audio de referencia (*Degradation Category Rating*, DCR). El MOS (*Mean Opinion Score*) es el promedio de los ACR medidos entre un gran número de usuarios. La metodología de evaluación subjetiva más ampliamente usada es la del MOS estandarizada en la recomendación de la Unión Internacional de Telecomunicaciones (*International Telecommunication Union*, ITU) ITU-T P.800 (Joskowicz and Bovino, 2007). Los métodos subjetivos representan la medida más fiable y respetada para medir la calidad percibida de la voz, pero resultan ser costosos y difíciles de implementar de forma permanente, en este sentido los métodos objetivos resultan ser los más apropiados (Martínez, 2014). Existen dos tipos de métodos de medición objetiva de calidad de voz propuestos por la ITU: los que utilizan métodos intrusivos, que basan su estimación en la comparación de señales; y los métodos no intrusivos, que basan su estimación en los parámetros funcionales de red (Veettil, 2014). El principal inconveniente de los métodos objetivos es que estos no dan como resultado una representación directa de la calidad percibida por el usuario, por tanto se hace necesaria su calibración en base a los resultados obtenidos de los métodos subjetivos, fundamentalmente del MOS (Martínez, 2014).

### 1.6.2.1. Métodos de medición no intrusivos

Los métodos no intrusivos, también conocidos como pasivos son desarrollados para ejecutar mediciones de tráfico de aplicaciones en tiempo real. A diferencia de los métodos intrusivos, estas mediciones son ejecutadas sin conocer la señal original. Los métodos tradicionales no intrusivos realizan su predicción directamente de diferentes parámetros de deterioro de la red IP, por ejemplo *jitter*, retardo y pérdida de paquetes (De Rango *et al.*, 2006). La figura 1–7 muestra un esquema de esta técnica.

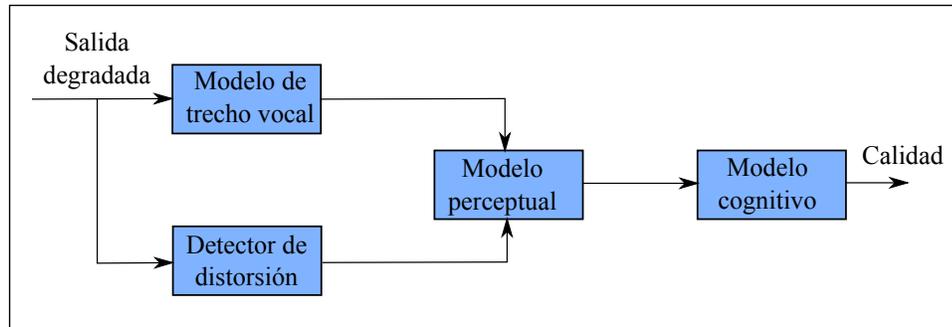


Figura 1–7: Técnica no intrusiva (Martínez, 2014).

Los métodos no intrusivos estiman la calidad percibida utilizando directamente los parámetros funcionales de la red, en donde estos y la calidad percibida por el usuario están directamente relacionados y forman junto a otras medidas, la calidad de servicio entregada al usuario (Martínez, 2014).

### 1.6.2.2. Modelo E

El método de medición objetiva más popular es el denominado Modelo E perteneciente a los métodos no intrusivos. Modelo E o *E-Model*, es una abreviatura de "European Telecommunications Standards Institute (ETSI) Computation Model", el cual fue desarrollado por un grupo de la ETSI seleccionado por la UIT. El modelo E se diferencia de otros métodos debido a que él representa a la misma vez una herramienta de simulación de la red (De Rango *et al.*, 2006). La figura 1–8 muestra un esquema funcional del Modelo E.

Este modelo funciona en base a asumir que cada tipo de degradación de la calidad está asociado con cierto tipo de factor de daño. El resultado primario del modelo es una cuantificación escalar de la calidad de audio que se estima percibirá un usuario. Una característica fundamental de este modelo es la utilización de factores de degradación de la transmisión que reflejen los efectos de los modernos dispositivos de procesamiento de señales. El modelo E calcula, en base a varios parámetros medibles de la red, el

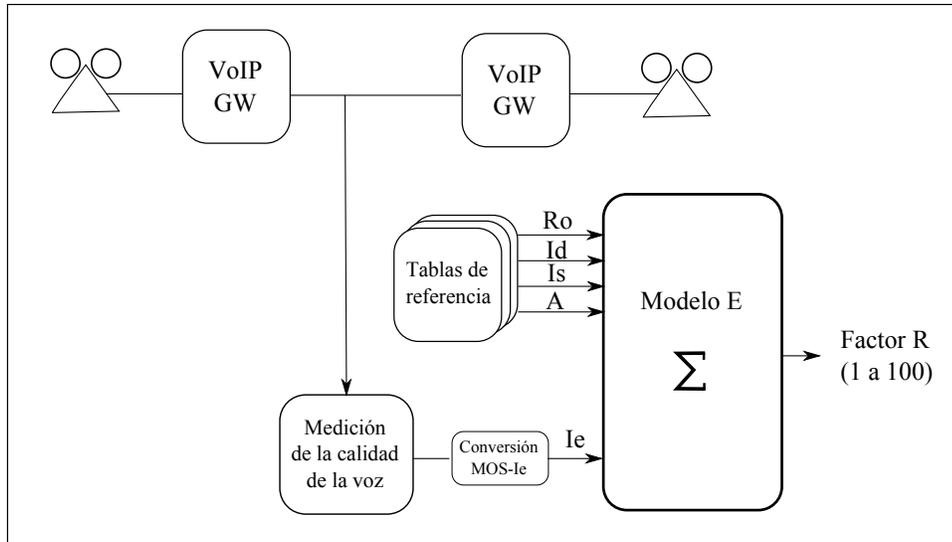


Figura 1-8: Esquema funcional del Modelo E (DeñRango *et al.*, 2006).

parámetro  $R$  (*Rating Factor*) que puede relacionarse con una escala de MOS como muestra la Tabla 1-1 (DeñRango *et al.*, 2006).

Tabla 1-1: Relación del parámetro  $R$  con el MOS

Satisfacción de los usuarios	Parámetro $R$	Valor MOS
Muy satisfechos	90	4.3
Satisfechos	80	4.0
Algunos usuarios no satisfechos	70	3.6
Muchos usuarios no satisfechos	60	3.1
Prácticamente todos los usuarios no satisfechos	50	2.6

Nota: Los valores de  $R$  y MOS corresponden a los límites inferiores.

$R$  se obtiene según la expresión 1.1:

$$R = (R_0 - I_s) - I_d - I_{e-eff} + A \tag{1.1}$$

El modelo E calcula los sumandos  $R_0$ ,  $I_s$ ,  $I_d$ ,  $I_{e-eff}$  a partir de los parámetros de transmisión que se observan en el modelo de referencia de la figura 1-9.  $R_0$  (efecto del ruido o relación señal a ruido) e  $I_s$  (relacionado con el volumen de la conexión y con la cuantificación) son intrínsecos a la señal de voz en la entrada de la red y no dependen de ésta, son irrelevantes en el caso de análisis de VoIP.  $I_d$  modela las degradaciones producidas por los retardos y el eco, mientras que  $I_{e-eff}$  representa las degradaciones producidas por los codecs y por las pérdidas de paquetes de distribución aleatoria.  $A$  representa el factor de ventaja que significa que el usuario aceptaría una degradación en la calidad a cambio de facilidad de acceso (Joskowicz and Bovino, 2007).

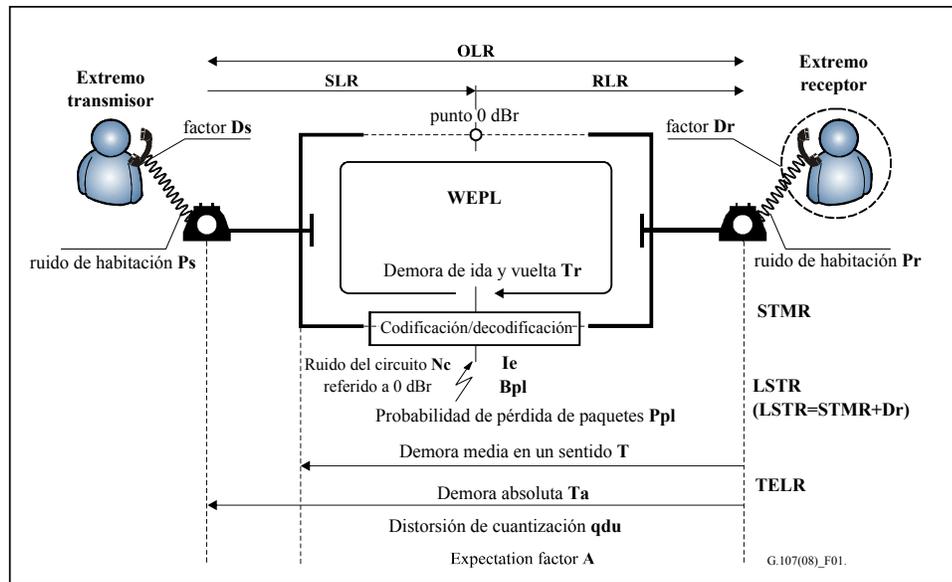


Figura 1–9: Factores para el cálculo de  $R$  (Dahlman *et al.*, 2013).

### 1.7. Algoritmos de control de demora

El problema del *jitter* se ha enfrentado mediante diferentes alternativas presentadas desde la década del 80. Los estudios en (Gruber, 1981) y (Montgomery *et al.*, 1983) marcan la génesis de estos trabajos.

El primer paso en la clasificación, define dos grandes grupos, atendiendo a si el *buffer* es fijo o adaptativo. Para la técnica de *buffer* fijo, la demora de extremo a extremo se mantiene constante durante toda la comunicación. Esta solución no es consistente con la variabilidad de la demora en las redes de paquetes. Por el contrario, con la técnica de *buffer* adaptativo, es posible ajustar la demora a las condiciones de la red. Esto evita que se introduzca una demora excesiva cuando las condiciones son favorables y viceversa. Para lograrlo, es necesario monitorear la conversación y adecuar el tamaño del *buffer*.

El segundo paso de la clasificación se aplica a las técnicas adaptativas solamente y tiene que ver con el momento en que se ajusta el *buffer* de reproducción. La variante intra-ráfagas modifica la demora de extremo a extremo en cualquier momento de la conversación. Por el contrario, cuando el ajuste se realiza inter-ráfagas, el cambio del tamaño del *buffer* se produce solo durante los intervalos de silencio. La variante inter-ráfagas puede materializarse siguiendo un modelo tolerante a pérdidas, intolerante a pérdidas o basado en calidad. En los intolerantes a pérdidas, no se tiene en cuenta la posibilidad que aparece con las técnicas de ocultamiento de paquetes perdidos, en lugar de esto se trabaja según el principio de minimizar las pérdidas de paquetes.

Evidentemente, tal modelo acarrea una sobrestimación de la demora requerida, lo que puede llegar a afectar la interactividad de la conversación. En el caso de los algoritmos tolerantes de pérdidas, se trata de alcanzar un compromiso entre la cantidad de paquetes que se pierden y la demora introducida. Por último, los algoritmos basados en calidad, tratan de maximizar la calidad percibida por el usuario.

### 1.8. Realimentación a nivel de radio

Un *buffer* de *jitter* en un receptor de voz sobre LTE (VoLTE) puede ser influenciado por realimentación a nivel de radio (RLF) desde el extremo local y remoto para ajustar preventivamente la demora del *buffer* de *jitter* en anticipación de pérdidas futuras predichas que tienen una alta probabilidad de ocurrencia. Los eventos de radio de la RLF y los escenarios que disparan los ajustes preventivos pueden ser identificados, y sus usos pueden ser expresados en términos de fórmulas matemáticas. En diseños anteriores, el *jitter* instantáneo es derivado de una historia ponderada del flujo de media, y consecuentemente solo los paquetes que ya han llegado se utilizan para calcular el *jitter* instantáneo para ajustar el tamaño del *buffer*. Proveyendo y usando RLF desde los extremos local y remoto, la demora anticipada (para paquetes que aún no han arribado) puede ser utilizada para preventivamente ajustar el *buffer*, minimizando la pérdida de paquetes sin introducir demora innecesaria (Verger *et al.*, 2014).

Un elemento de procesamiento acoplado al radio y al *buffer* de *jitter* en el dispositivo UE puede operar para identificar eventos de radio que afectan el tiempo de futuros paquetes que serán recibidos en el futuro por el radio, calcular una demora proyectada asociada con los eventos de radio, y determinar una demora efectiva de acuerdo a la demora proyectada. La demora efectiva puede entonces ser utilizada para controlar cuando los paquetes entrantes que ya están en cola, son liberados del *buffer* de *jitter* (Verger *et al.*, 2014).

El elemento de procesamiento en el UE también puede operar para estimar dos factores empíricos para cada tipo de evento de radio, y calcular la demora proyectada asociada a cada tipo de evento de radio de acuerdo a los dos factores empíricos estimados para el evento de radio respectivo. Dos factores empíricos pueden incluir un primer factor empírico correspondiente a una demora esperada causada por el tipo de evento de radio respectivo, y un segundo factor empírico correspondiente a la duración del evento de radio respectivo. Al calcular la demora proyectada, el elemento de procesamiento puede aplicar una función de ponderación al segundo factor empírico (Verger *et al.*, 2014).

Los eventos que pueden ser identificados y utilizados para ajustar la razón de reproducción del *buffer* de *jitter* incluyen (Verger *et al.*, 2014):

- *Handovers*: Los *handovers* (HO) inter-eNB con transferencia X2 pueden introducir cortes de voz de entre 80 ms a 100 ms, los cuales son típicamente precedidos por reportes de mediciones desde el UE a la eNB. Basado en los reportes de mediciones, la eNB de servicio puede comenzar la preparación del *handovers*, lo que implica intercambio de señalización entre la eNB de servicio y la eNB objetivo y control de admisión del UE en la eNB objetivo. Luego de que termina la preparación del HO, la eNB de servicio envía un comando de *handover* al UE, y la conexión con la celda de servicio es liberada. Esto marca el comienzo de la duración de la desconexión, a continuación el UE se puede sincronizar con, y acceder a la eNB objetivo usando el Canal de Acceso Aleatorio (*Random Access Channel*, RACH). Una vez que el UE reanuda la transferencia de datos con la eNB objetivo, el período de corte finaliza. La implementación del *buffer* de *jitter* puede utilizar el intervalo de tiempo (entre el envío de un reporte de medición que cumpla con la histéresis del HO, y el comando de HO que marca el comienzo del corte) para prepararse para este corte por venir.
- Continuidad de Llamada de Voz de Radio Único (*Single Radio Voice Call Continuity*, SRVCC): SRVCC es una solución que transfiere llamadas de VoLTE en progreso con una red LTE a redes de voz de generaciones anteriores. Por tanto, SRVCC es muy importante para llenar las brechas que existen en el despliegue de redes LTE. Este procedimiento puede producir un corte promedio de 200 ms. Aunque el ciclo de vida del *buffer* de *jitter* puede expirar antes de culminar este procedimiento, el conocimiento previo de un procedimiento SRVCC próximo puede ayudar al UE a construir suficientes tramas en el *buffer* de *jitter* para enmascarar el corte percibido por el usuario.
- Falla del Enlace de Radio (*Radio Link Failure*, RLF): (T311, N310) Luego de recibir N310 indicaciones consecutivas de fuera de sincronismo desde capas inferiores, se puede inicializar un temporizador T310, y al expirar el conteo, el UE puede intentar un procedimiento de restablecimiento de la conexión. Típicamente, SRVCC se activa antes de este evento. Sin embargo, para operadores que no soportan SRVCC, esto puede resultar en una llamada caída, si no se encuentra otra celda adecuada. Para acomodar desconexiones como esta, el *buffer* de *jitter* puede ser operado para llavear las mediciones de la Señal de Referencia de Potencia Recibida (*Reference Signal Received Power*, RSRP) con bastante antelación antes de incrementar el almacenado.
- Razón de Error de Bloque (*Block Error Rate*, BLER) para el enlace de subida (UL) y de bajada (DL): La BLER tanto para el UL como para el DL puede ser utilizada de una manera similar y en adición a los reportes de mediciones, para adaptar la razón de reproducción del *buffer* de *jitter*.

- Número de retransmisiones del UL: El número de retransmisiones del UL puede desempeñar también un rol fundamental en el *jitter* visto desde el extremo remoto.

Una vez que se han identificado los eventos locales y remotos que probablemente contribuirán a picos en el *jitter*, se pueden estimar un número específico de factores empíricos para cada tipo de evento  $e$ . Como se mencionó anteriormente se pueden estimar dos factores empíricos. Estos son  $D_e$  y  $T_e$ , los que corresponden a la demora adicional esperada causada por el evento y la duración del deterioro, respectivamente. La demora adicional total que se espera del enlace de radio local puede ser definida mediante la siguiente ecuación:

$$D_{radio}(t) = \sum_e D_e W(t - t_e, T_e) \quad (1.2)$$

donde:

$D_e$  : demora asociada al evento  $e$ .

$t - t_e$  : tiempo actual menos el tiempo en el cual el evento ocurrió.

$T_e$  : duración del deterioro.

$W(t, T)$  : función de ponderación de tiempo adecuada.

La demora total óptima del *buffer* de *jitter* puede ser calculada de la manera siguiente:

$$D_{efectiva} = D_{radio}(t) + D_{cola} \quad (1.3)$$

donde  $D_{cola}$  es la demora calculada para todos los paquetes que ya han sido puestos en cola en el *buffer* de *jitter*.

Se hace posible determinar la demora total esperada actual que incorpora todos los eventos de radio previos, y es propiamente ponderada por tiempo de arribo. La función de ponderación puede ser aplicada para asegurar que  $D_{radio}(t)$  refleje el estado actual del radio, y además para asegurar que el efecto de un evento particular no sea contado doble una vez que el paquete es puesto en cola en el *buffer* y la demora haya sido añadida a  $D_{cola}$ . Consecuentemente, la función de ponderación puede tender rápidamente a cero en el tiempo esperado entre cuando el evento ocurre y cuando los paquetes con puestos en cola. Esto asegura que  $D_{radio}(t)$  cumpla el rol de porción de ataque-rápido-descenso-rápido del algoritmo del *buffer* de *jitter*, mientras que  $D_{cola}$  cumple el rol de adaptación lenta a las condiciones promedio del enlace (Verger *et al.*, 2014).

Los beneficios de integrar la información descrita anteriormente en la implementación del *buffer* de *jitter* incluyen la capacidad de hacer el algoritmo usado para  $D_{cola}$  más agresivo (resultando en demoras más cortas) sin incrementar el riesgo de comprometer

el flujo de media debido a deterioros de radio. También permite insertar demora preventivamente en el sistema para compensar el *jitter* inducido por el deterioro de radio por venir, sin esperar que el *buffer* de *jitter* pierda un paquete, y luego adaptarse al evento después del hecho. Finalmente, hace posible que el *buffer* de *jitter* no tenga que adaptarse innecesariamente a una mayor latencia debido a eventos con degradación de condiciones de radio de una única vez donde la degradación es solo momentánea (Verger *et al.*, 2014).

Esta tesis está enfocada solamente en los eventos *handovers*.

### 1.9. Tipos de *handovers* (HO) en LTE

Los *handovers* pueden ser clasificados teniendo en cuenta el sistema objetivo, la frecuencia o el método mediante el cual son llevados a cabo. Los *handovers* intra-LTE incluyen transiciones hacia una frecuencia portadora igual o diferente dentro de un sistema LTE. Estos pueden ser divididos en los siguientes casos (Helenius, 2011):

- *Handovers* intra-eNodeB: se refieren al caso donde las celdas fuente y destino residen en la misma eNodeB. En este caso no se requiere procedimiento X2 para el *handover*.
- *Handovers* inter-eNodeB: representa una situación donde las celdas de origen y objetivo se encuentran en diferentes eNodeBs. Este caso asume que la MME no cambiará como resultado del *handover*. La S-GW puede o no ser reubicada. Se necesita iniciar un proceso de *handover* X2 o S1.
- *Handovers* inter-eNodeB con cambio de MME: el proceso de *handover* basado en X2 no puede manejar una reubicación de la MME, por lo que se debe utilizar el procedimiento basado en S1.

LTE no está limitado solamente a *handovers* intra sistema. Un UE en una red LTE también es capaz de completar un *handover* optimizado hacia otros sistemas. Estos sistemas incluyen UMTS, GSM, y también cdma2000 especificado por 3GPP2. Estos tipos de *handovers* se conocen como inter-Tecnología de Acceso de Radio (inter-RAT) (Helenius, 2011).

Esta tesis se concentra en los casos intra-LTE, intra-frecuencia y basados en X2.

#### 1.9.1. Configuración de mediciones

Para realizar *handovers*, la eNodeB debe brindar al UE los datos de configuración necesarios para hacer las mediciones. Esto se hace utilizando señalización RRC dedicada. La configuración es señalizada al UE luego de que este se ha conectado a la red. También puede ser actualizada por la nueva celda de servicio luego de que ha ocurrido un *handover* satisfactorio. La configuración de medición está compuesta por cinco parámetros listados debajo (3GPP TS 36.331, 2014):

- **Objetos de medición:** representan las fuentes de las mediciones. En caso de un objetivo de medición intra-sistema, un objeto de medición representa una sola frecuencia portadora LTE.
- **Configuraciones de reportes:** dictan cuándo el UE debe enviar un reporte de medición a la eNodeB. La configuración de reporte también incluye información de qué tipo de cantidades y a cuántas celdas reportar.
- **Identidades de mediciones:** se utilizan para conectar un objeto de medición con una configuración de reporte. Se pueden configurar múltiples identidades de mediciones en un solo UE. Esto permite agregar múltiples detonadores de eventos de medición a una sola portadora, así como agregar el mismo detonador a múltiples portadoras. El UE utiliza el número de identidad de medición como referencia cuando envía reportes de mediciones.
- **Configuraciones de cantidades:** definen las cantidades de medición y el filtrado apropiado que el UE debe realizar cuando mide.
- **Espacios de medición:** son los períodos de tiempo que el UE puede utilizar para hacer mediciones. No se planifican transmisiones del UL, ni del DL durante estos tiempos. Los espacios de mediciones no se necesitan en escenarios intra-frecuencia, debido a que el UE ya mide las celdas de la portadora de servicio.

A continuación se muestra la relación existente entre los tres primeros parámetros explicados anteriormente.

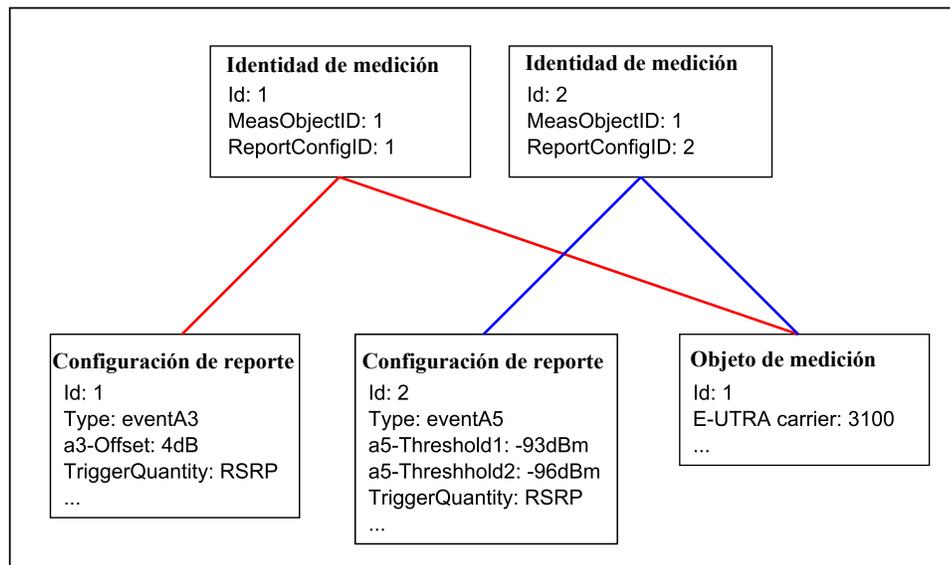


Figura 1–10: Configuración de medición básica utilizando una portadora (Helenius, 2011).

### 1.9.2. *Handover X2*

Para realizar el *handover*, las eNodeBs tienen que intercambiar señalización con las otras. Esto se realiza a través de la interfaz X2, utilizando el protocolo X2-AP. La conexión de señalización requiere que las dos eNodeBs tengan la interfaz X2 configurada. En caso que esta interfaz no exista o esté bloqueada, es posible realizar un *handover* asistido por la MME usando la interfaz S1. En el caso X2, el *handover* se completa puramente dentro la E-UTRAN, sin ninguna participación del núcleo de la red (Helenius, 2011).

En contraste con UMTS, el *handover* en LTE es duro. Esto significa que la interfaz de radio hacia la eNodeB fuente es desmantelada antes de que la nueva conexión con la eNodeB objetivo sea construida. Por tanto, la pérdida de datos durante el tiempo de desconexión es un problema. Para prevenir la pérdida de paquetes, LTE usa transferencia de datos desde la eNodeB fuente hacia la eNodeB objetivo durante el proceso de *handover* (Helenius, 2011).

En la figura 1–11 se muestra una descripción detallada del intercambio de señalización. La explicación de los mensajes es enumerada debajo.

1. El UE envía un Reporte de Medición basado en la configuración de medición establecida por la eNodeB. Este reporte contiene información acerca de las celdas vecinas.
2. La eNodeB analiza el resultado de medición, y decide que el *handover* es necesario. Escoge la mejor celda objetivo del reporte de medición enviado por el UE.
3. La eNodeB envía un mensaje Handover Request utilizando X2-AP a la eNodeB objetivo. El mensaje contiene información necesaria para la realización del HO. Esta información incluye referencias del contexto de señalización del UE en la celda fuente (de S1 y X2), contexto RRC, contexto de Portadoras de Acceso de Radio (*Radio Access Bearer*, RAB) y la identificación de la celda objetivo.
4. La eNodeB objetivo puede realizar control de admisión para chequear si tiene recursos disponibles para el nuevo UE. Considera la información de QoS recibida en el contexto RAB mientras asegura la disponibilidad de recursos.
5. Si el control de admisión acepta el HO, la eNodeB objetivo comienza a preparar la interfaz de radio. También envía un mensaje Handover Request Acknowledge a la eNodeB fuente. Este mensaje contiene un mensaje RRC RRCConnectionReconfiguration dentro de un contenedor transparente para que la eNodeB fuente los envíe al UE. El mensaje contiene parámetros necesarios para que el UE se conecta a la eNodeB objetivo. Incluye además un preámbulo de acceso aleatorio dedicado. Esto significa que la eNodeB objetivo ha reservado recursos de acceso de radio para el UE. De esta

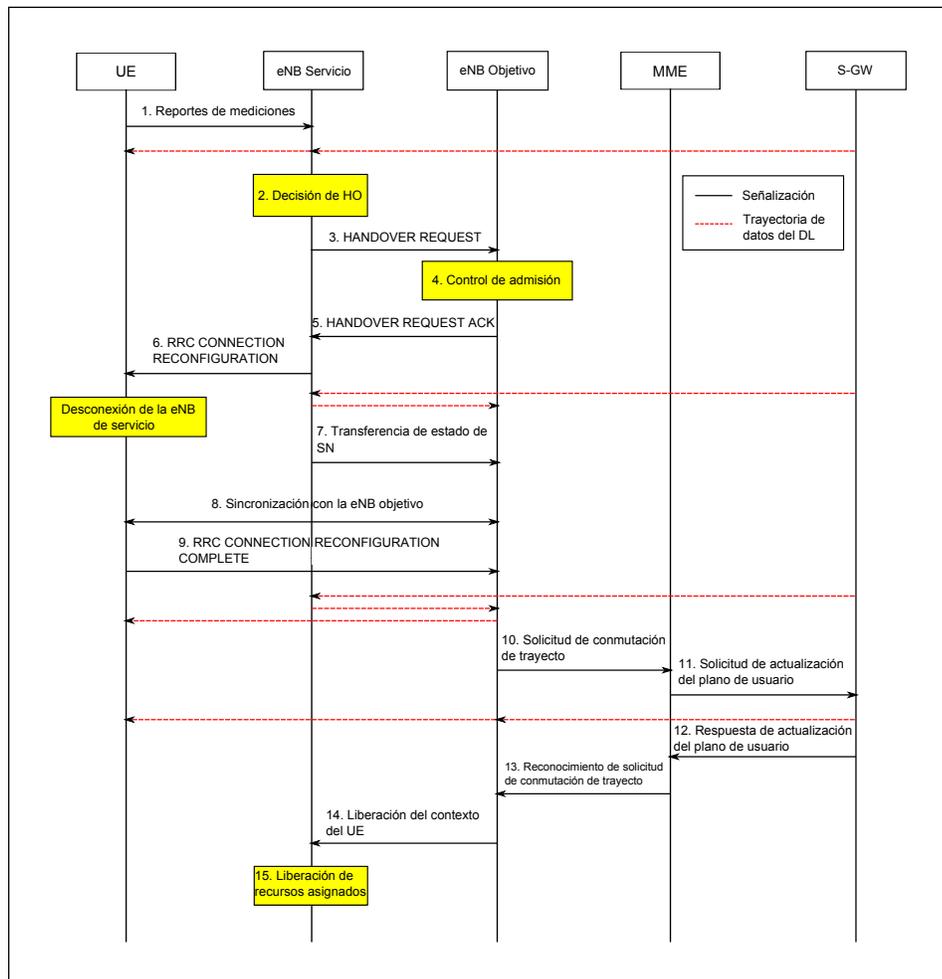


Figura 1–11: Flujo de señalización de handover basado en X2, intra S-GW (Helenius, 2011).

manera el UE no tiene que realizar un procedimiento de acceso aleatorio basado en contención.

6. La eNodeB fuente transfiere el RRCConnectionReconfiguration al UE. Tan pronto como el mensaje se envía, la eNodeB puede comenzar la transferencia de datos del DL a través de X2. Cuando el UE recibe el mensaje, se desconecta de la celda de servicio.
7. La eNodeB de servicio envía un mensaje de Transferencia de Estado del Número de Secuencia (*Sequence Number*, SN) a la objetivo a través de la interfaz X2. Este mensaje se utiliza para transferir los números de secuencia PDCP a la eNodeB objetivo. Para el UL el mensaje incluye el número de secuencia de la primera unidad de datos faltante. Para el DL se anuncia el próximo número de secuencia a ser asignado.
8. El UE usa los parámetros dados para sincronizarse con la celda objetivo. Si ha recibido un preámbulo dedicado de acceso aleatorio, no necesita realizar el acceso aleatorio basado en contención. En esta tesis se asume que se usa el preámbulo dedicado.

9. El UE reconoce a través de X2 a la eNodeB objetivo que el HO ha sido exitoso vía el mensaje RRCConnectionReconfigurationComplete. Al recibir esta confirmación, la eNodeB objetivo comienza a enviar los datos transferidos al UE. Se requiere que la eNodeB objetivo envíe todos los paquetes recibidos a través de la interfaz X2 antes que cualquier posible nuevo paquete recibido desde la S-GW.
10. Después de una confirmación del UE, la eNodeB objetivo envía un mensaje Path Switch Request hacia la MME utilizando S1-AP sobre S1-MME. Esto se hace para notificar a la MME sobre el cambio de localización del UE y para solicitar la conmutación del trayecto del plano de usuario hacia la eNodeB objetivo.
11. Al recibir la petición de conmutación de trayecto, la MME envía a un mensaje User Plane Update hacia la S-GW sobre S11. Cuando la S-GW recibe la petición, intercambia la ruta de datos de la eNodeB originaria a la eNodeB objetivo. Justo después de conmutar la ruta, la S-GW envía un paquete GTP “marcador de final” hacia la eNodeB fuente. Este paquete no contiene datos de usuario. Cuando la eNodeB fuente recibe este paquete, lo debe remitir a la eNodeB objetivo. El marcador de final se usa para señalar el fin de datos transferidos para la eNodeB objetivo. La eNodeB objetivo puede usar esta información en la función de reordenamiento de paquetes.
12. La S-GW envía un mensaje User Plane Update Response a la MME sobre S11 para señalar una conmutación de trayecto satisfactoria.
13. La MME reconoce la solicitud de conmutación de ruta hacia la eNodeB objetivo sobre S1-MME.
14. La eNodeB objetivo notifica a la eNodeB fuente acerca del *handover* satisfactorio sobre X2.
15. Cuando la eNodeB fuente recibe el mensaje UE Context Release, puede eliminar cualquier contexto que todavía mantenga para el UE. El contexto se reserva hasta este punto en caso que el *handover* falle.

### 1.10. Conclusiones parciales

1. La demanda de nuevos servicios móviles y la evolución de la interfaz de radio a LTE han causado la evolución del núcleo de la red. Los estándares de LTE se diseñan como una tecnología móvil basados puramente en IP.
2. Los servicios de VoLTE tienen restricciones de demora, variación de la demora ( *jitter* ) y pérdida de paquetes muy estrictas, estableciéndose 200 ms como valor de demora extremo a extremo máximo antes de que se comience a degradar la calidad, así como un porcentaje de pérdida de paquetes igual al 2%.

3. La anticipación de la ocurrencia de los eventos de radio descritos permite insertar demora preventivamente en el sistema para compensar el *jitter* inducido por el deterioro de radio por venir, sin esperar que el *buffer* de *jitter* pierda un paquete, y luego adaptarse al evento después del hecho.
4. Se elige solamente el *handover* para su estudio en esta tesis.

# CAPÍTULO 2

## MATERIALES Y MÉTODOS

En el presente capítulo se define el *software* utilizado para la generación de trazas de VoLTE, exponiendo los parámetros y modelos de simulación empleados, particularizando en: modelo de movilidad, modelo de desvanecimiento, algoritmo de *handover*, modelo de aplicación, así como el método utilizado para recopilar los datos. Como último punto, se hace una descripción del funcionamiento de los algoritmos que serán retomados en el capítulo siguiente.

### 2.1. Definición del simulador

#### 2.1.1. NS-3

En la investigación se utiliza el simulador de eventos discretos NS-3, este cuenta con un módulo para redes LTE llamado LTE - EPC *Network Simulator* (LENA), que se encuentra actualmente en etapa de desarrollo. EL simulador posee la ventaja de estar implementado en código abierto, permitiendo realizar modificaciones y adaptaciones al código en caso de ser necesarias (Fontán *et al.*, 2012).

NS-3 es un simulador de eventos discretos, escrito completamente en C++, con licencia GNU GPLv2. Tiene un núcleo organizado en librerías, cada una de ellas contiene las clases necesarias para simular un protocolo u objeto particular dentro de un sistema (Fontán *et al.*, 2012).

Para simular una red se crea un script en C++ que define todos los objetos a utilizar, las vinculaciones entre estos y parámetros asociados. También especifica la duración de la simulación y brinda los parámetros de entrada para la generación de números aleatorios (Fontán *et al.*, 2012).

Existen distintos niveles de Interfaz de Programa de Aplicación (*Application Program Interface*, API) en NS-3, se puede programar a bajo nivel con un manejo fino de todos los objetos y parámetros del sistema, como también a más alto nivel con una API conformada por *Helpers*. Estos *Helpers* son objetos que manejan la configuración de otros objetos, ofreciendo de esta forma una interfaz más amigable a la hora de programar (Fontán *et al.*, 2012).

### 2.1.2. LENA

El simulador LENA está actualmente en etapa de desarrollo por el Centre Tecnologic de Telecomunicacions de Catalunya (CTTC) y está basado en el simulador de redes NS-3. El módulo LTE de LENA fue incorporado al release NS3.15 del simulador NS-3, agregando, entre otras capacidades, la posibilidad de poder simular un escenario LTE+EPC (Fontán *et al.*, 2012).

## 2.2. Descripción del escenario

### 2.2.1. Topología

Se utilizó una red compuesta por 7 emplazamientos de eNodeBs, cada uno formado a su vez por tres celdas. Estos emplazamientos están distribuidos en forma de rejillas hexagonales, en una formación 2-3-2 como se muestra en la figura 2-1. Los emplazamientos están distanciados 500 m uno de otro.

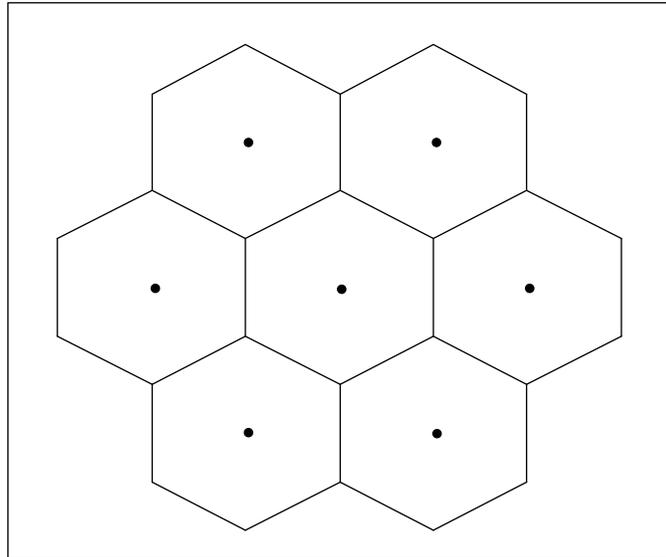


Figura 2-1: Distribución de los emplazamientos de eNodeBs.

El escenario responde a un UE que intercambia información con un cliente remoto (*remote host*, RH). Dado que NS-3 no soporta dos EPC, se define un enlace punto a punto entre la pasarela de datos/servicios y el RH para simular el *core*. Este enlace tiene una demora de 10 ms.

### 2.2.2. Modelo de movilidad

En el proceso de simulación de redes inalámbricas la movilidad es un componente fundamental. Para obtener resultados de simulación precisos es un prerequisite poseer un buen entendimiento de las propiedades fundamentales de la movilidad de los nodos. La caracterización de las propiedades del modelo de movilidad es el punto de partida

para la derivación de resultados analíticos concernientes a redes móviles (Mitsche *et al.*, 2014).

Un ejemplo notable del impacto del modelado de movilidad en el proceso de simulación y análisis de redes inalámbricas, es el modelo de movilidad de vector aleatorio (*Random Waypoint Mobility Model*, RWPM) (Mitsche *et al.*, 2014). En este modelo, un nodo móvil se mueve en un dominio convexo a través de una trayectoria en zigzag, en la que cada segmento de línea recta es llamado tramo. En cada punto de giro el nodo escoge un nuevo destino aleatoriamente y se mueve hacia ese destino a una velocidad constante, la cual se obtiene independientemente de una distribución de velocidad dada en cada punto de desviación. El nodo puede también permanecer estacionario por un tiempo de pausa aleatorio antes de comenzar su movimiento hacia el próximo destino (Hyytiä *et al.*, 2006). En la figura 2-2 se muestra el comportamiento del modelo RWPM.

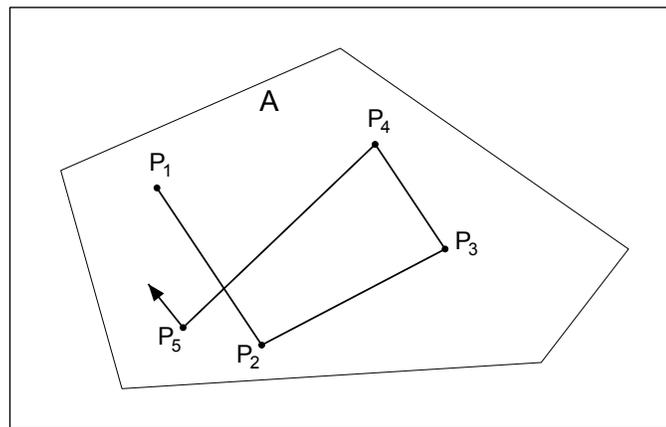


Figura 2-2: Movimiento en zigzag del proceso RWM (Hyytiä *et al.*, 2006).

En las simulaciones desarrolladas durante la investigación se utilizó el modelo de movilidad de vector aleatorio de estado estable (*Steady-State Random Waypoint Mobility Model*, SSRWPM). Este modelo está basado en el RWMM, para el caso cuando la velocidad, la pausa y la posición son variables aleatorias uniformemente distribuidas. La diferencia reside en que los valores iniciales de estos parámetros no provienen de una distribución uniforme, sino de una distribución estacionaria del RWMM. La implementación de este modelo es específica de 2D y está creada para nodos no estacionarios (ns-3 project, 2015c).

La implementación de este modelo en NS-3 cuenta con los siguientes atributos:

- MinSpeed: Valor mínimo de velocidad en m/s.
- MaxSpeed: Valor máximo de velocidad en m/s.
- MinPause: Valor mínimo de pausa en s.
- MaxPause: Valor máximo de pausa en s.

- MinX: Valor X mínimo de la región de recorrido en m.
- MaxX: Valor X máximo de la región de recorrido en m.
- MinY: Valor Y mínimo de la región de recorrido en m.
- MaxY: Valor Y máximo de la región de recorrido en m.
- Z: Valor Z de la región de recorrido (constante) en m.

En las simulaciones realizadas se definió una “caja” cuyas dimensiones son pasadas a los atributos anteriores para definir la región de recorrido dentro de la cual se mueven los UEs. Las dimensiones de esta caja dependen de la distribución y cantidad de emplazamientos utilizados. En el escenario implementado se utiliza una velocidad constante y un movimiento continuo. La tabla 2–1 muestra los valores asignados a cada atributo.

Tabla 2–1: Valores asignados a los atributos del modelo SSRWMM

Parámetro	Valor
MinSpeed	8.3333 m/s
MaxSpeed	8.3333 m/s
MinPause	0
MaxPause	0
MinX	-250 m
MaxX	1250 m
MinY	-250 m
MaxY	1116.03 m
Z	1.5 m

### 2.2.3. Modelo de pérdida de trayectos

En las simulaciones se utilizó el modelo de pérdidas de propagación de Friis. Según (ns-3 project, 2015a), la expresión 2.1 se utiliza para calcular el valor de la potencia recibida:

$$P_R = \frac{P_T G_T G_R \lambda^2}{(4\pi d)^2 L} \quad (2.1)$$

donde:

$P_R$  : Potencia de recepción (W).

$P_T$  : Potencia de transmisión (W).

$G_T$  : Ganancia de la antena transmisora (adimensional).

$G_R$  : Ganancia de la antena receptora (adimensional).

$\lambda$  : Longitud de onda (m).

$d$  : Distancia (m).

$L$  : Pérdidas del sistema (adimensional).

El modelo de Friis es válido solo para propagación en el espacio libre dentro de la llamada región de campo lejano, la cual puede ser considerada aproximadamente como

la región en la que se cumple  $d > 3\lambda$ . El modelo retornará un valor para  $d < 3\lambda$ , ya que esto (en lugar de un error fatal), es práctico para muchos escenarios de simulación (ns-3 project, 2015a).

#### 2.2.4. Modelo de Desvanecimiento

En los radioenlaces es inevitable que la señal sufra de algunas fluctuaciones, las cuales se denominan desvanecimiento de la señal. Estas son producto de aspectos tales como la precipitación, refracción o propagación multitrayecto y dependen del tipo de fenómeno atmosférico, así como de la frecuencia de operación del enlace.

El módulo LTE incluye un modelo de desvanecimiento basado en trazas. La principal característica de este modelo es el hecho de que la evaluación de desvanecimiento durante el tiempo de simulación está basada en trazas pre-calculadas. Esto se hace para limitar la complejidad computacional del simulador (ns-3 project, 2015b).

En la simulación implementada en la presente tesis se utilizó la traza de desvanecimiento `fading_trace_EVA_60kmph.fad` provista por NS-3, la misma se ajusta a un escenario vehicular según se define en (3GPP TS 36.104, 2014). A continuación se muestra un sumario de la traza en cuestión.

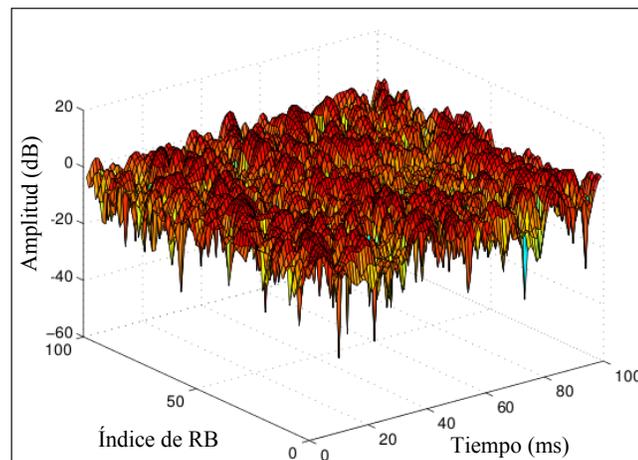


Figura 2-3: Porción de la traza `fading_trace_EVA_60kmph.fad` (ns-3 project, 2015b).

#### 2.2.5. Configuración del protocolo RLC

El protocolo RLC se especifica en (3GPP TS 36.322, 2007), y como se explicó en el capítulo 1, comprende tres tipos diferentes de modos de operación: Modo Transparente (*Transparent Mode*, TM), Modo de No Reconocimiento (*Unacknowledge Mode*, UM) y Modo Reconocido (*Acknowledged Mode*, AM). El simulador LENA incluye un modelo para cada uno de los modos de operación del protocolo RLC (ns-3 project, 2015b).

Se utilizó el modo RLC AM en el diseño presentado en esta tesis, por lo que la parte transmisora de la entidad RLC soporta retransmisiones de unidades de paquetes de datos RLC.

### 2.2.6. Configuración del protocolo RRC

La implementación del protocolo RRC en el proyecto LENA tiene dos vertientes: RRC ideal y RRC real. De acuerdo al modelo RRC ideal, todos los mensajes RRC y elementos de información son transmitidos entre la eNodeB y el UE de una manera ideal, sin consumir recursos de radio y sin errores. Teniendo en cuenta esta cuestión, se eligió el modelo RRC real donde para cada mensaje RRC enviado, se crea una PDU real siguiendo la codificación ASN.1 como se especifica en (3GPP TS 36.331, 2014). Además, las PDUs RRC son enviadas sobre Portadoras de Radio de Señalización (*Signaling Radio Bearers*, SRB) y están sujetas al mismo modelado de transmisión utilizado para comunicaciones de datos, incluyendo planificación, consumo de recursos de radio, errores del canal, demoras, retransmisiones, entre otros (ns-3 project, 2015b).

### 2.2.7. Planificador

En la simulación se utilizó el planificador de justicia proporcional (*Proportional Fair*, PF). En esta estrategia, los recursos compartidos son asignados al usuario con las condiciones de radio relativamente mejores, o sea, en cada instante de tiempo, el usuario  $k$  es seleccionado para transmitir de acuerdo a la expresión 2.2 (Dahlman *et al.*, 2013):

$$k = \arg \left( \max \frac{R_i}{\bar{R}_i} \right) \quad (2.2)$$

donde:

$R_i$  : Razón de datos instantánea para el usuario  $i$ .

$\bar{R}_i$  : Razón de datos promedio para el usuario  $i$ .

El promedio es calculado sobre un cierto período  $T_{PF}$ . Para asegurar un uso eficiente de las variaciones a corto plazo del canal y, al mismo tiempo, limitar las diferencias a largo plazo en la calidad de servicio a un nivel aceptable, se debe establecer la constante de tiempo  $T_{PF}$  mayor a la constante de tiempo para las variaciones a corto plazo. A la vez,  $T_{PF}$  debe ser suficientemente pequeño para que las variaciones de calidad dentro del intervalo  $T_{PF}$  no sean notadas fuertemente por el usuario (Dahlman *et al.*, 2013). En la siguiente figura se muestra un ejemplo de planificación utilizando justicia proporcional, el usuario planificado se muestra con líneas discontinuas.

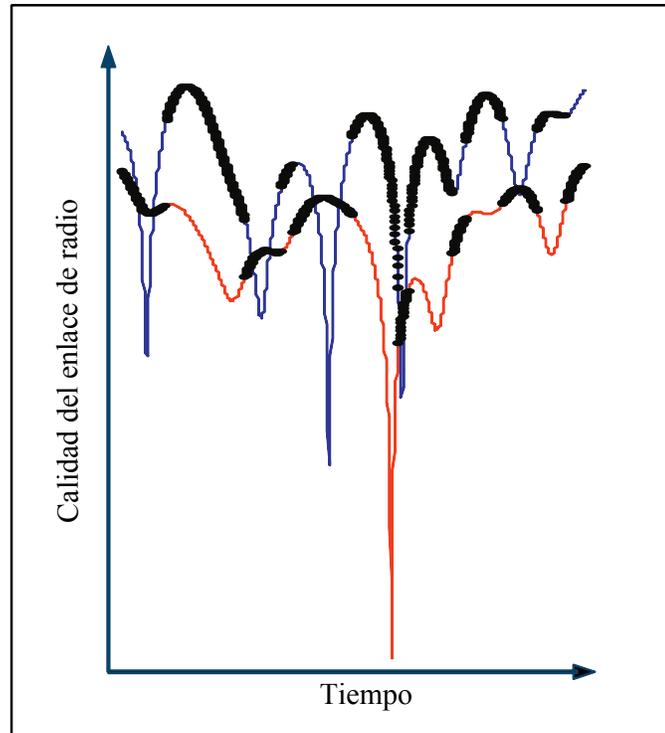


Figura 2–4: Comportamiento de planificación para dos usuarios con calidades promedio de canal diferentes utilizando el planificador de justicia proporcional (Dahlman *et al.*, 2013).

### 2.2.8. Algoritmo de *handover*

En NS-3 existen dos formas de disparar un *handover*: manual y automática. La primera se logra planificando en el código del programa la ejecución del método `SendHandoverRequest`, especificando tiempo de simulación, eNodeB de servicio y eNodeB objetivo. En la opción automática, la entidad RRC de la eNodeB ejecuta el *handover* basado en mediciones del UE y de acuerdo al algoritmo de *handover* seleccionado.

En las simulaciones realizadas se utilizó la forma automática, así como el algoritmo de *handover* `A3RsrpHandoverAlgorithm`, también conocido como algoritmo de *handover* de la mejor celda. Como indica el nombre, la idea detrás del mismo es proveer al UE la mejor Señal de Referencia de Potencia Recibida (*Reference Signal Received Power*, RSRP) (ns-3 project, 2015b).

Para realizar este concepto se escogió el evento A3 (la RSRP de la celda vecina supera a la RSRP de la celda de servicio) ejecutándose el *handover* hacia la mejor celda en el reporte de medición cuando se satisface dicho evento.

Una simulación que utilice este algoritmo es usualmente más vulnerable a *handovers* ping-pong, especialmente cuando se habilita el modelo de desvanecimiento. Para contrarrestar esta situación se incluye en el algoritmo los parámetros histéresis y tiempo de disparo (*Time-to-Trigger*, TTT) definidos en (3GPP TS 36.331, 2014) a la configuración de mediciones (ns-3 project, 2015b).

La histéresis demora el *handover* en relación a la RSRP. El valor es expresado en dB, está en un rango entre 0 y 15 dB, y tiene una precisión de 0.5 dB.

El TTT demora el *handover* en relación al tiempo. 3GPP define 16 valores válidos para el TTT: 0, 40, 64, 80, 100, 128, 160, 256, 320, 480, 512, 640, 1024, 1280, 2560, y 5120; todos expresados en milisegundos.

Para las simulaciones realizadas se estableció un valor de 0.0 dB para la histéresis y de 40 ms para el TTT.

### 2.2.9. Potencia de transmisión. Modelo de antena. Ancho de banda

En la simulación el UE transmite con una potencia de 10 dBm y la eNodeB irradia 46 dBm. Se utilizó el modelo de antena parabólica con un ancho de haz de 70 grados y una atenuación máxima de 20 dB. Para ambos, DL y UL, se asigna un ancho de banda de 5 Mhz.

### 2.2.10. Modelo de aplicación

El códec de voz adaptativo multi-razón (*Adaptive Multirate*, AMR) es uno de los códecs de voz más populares utilizados en LTE (Musable, 2012). Para tráfico de VoLTE, generalmente se emplean códecs AMR con una razón fuente de 12.2 kbps para codificar señales de voz (Jin *et al.*, 2011). Este códec tiene una duración de paquete de 20 ms, dando un tamaño de paquete típico de 31 bytes. Sin embargo, el encabezado normalmente contiene 40 o 60 bytes, que comprenden 12 bytes del protocolo de tiempo real (*Real Time Protocol*, RTP), 8 bytes del protocolo de datagrama de usuario (*User Datagram Protocol*, UDP) y 20 bytes para IPv4 o 40 bytes para IPv6. Tales encabezados son inapropiados a través del cuello de botella de la interfaz aérea (Cox, 2012).

Para resolver este problema, PDCP incluye un protocolo de la IETF conocido como compresión robusta de encabezado (*Robust Header Compression*, ROHC). El principio es que el transmisor envía el encabezado completo en el primer paquete, pero solo envía las diferencias en paquetes subsiguientes. La mayoría del encabezado permanece constante de un paquete al siguiente, por lo que los campos con diferencias son considerados más pequeños. El protocolo puede comprimir los encabezados originales de 40 y 60 bytes a 1 y 3 bytes respectivamente, lo cual reduce grandemente el sobreencabezado (Cox, 2012). En las simulaciones se utilizó la versión IPv4.

Con el objetivo de establecer un tráfico de VoIP, se utilizó la aplicación UDPClientServer, provista por NS-3. Para emular el códec anteriormente descrito se utilizan los siguientes atributos de la aplicación:

- Interval: Tiempo a esperar entre paquetes. Se le asigna un valor de 20 ms.
- PacketSize: Tamaño de los paquetes generados. Se asigna un valor de 20 bytes teniendo en cuenta la compresión de cabecera y que la aplicación por defecto adiciona un encabezado de 12 bytes.

Además de todo lo mencionado anteriormente, se estableció una portadora dedicada por UE, con un valor de QCI igual 1, definido por la clasificación NGBR\_IMS, como se especifica en (3GPP TS 23.203, 2012). Se estableció el tiempo de inicio de la aplicación utilizando una variable aleatoria uniforme, cuyo valor mínimo es 500 ms y valor máximo es 510 ms, para evitar comportamientos de simulación indeseados y dar tiempo suficiente para que se creen todos los objetos definidos y se establezcan las conexiones entre ellos. El tiempo final de la aplicación es 230 s, mientras que el tiempo total de la simulación es de 240 s.

### 2.3. Recolección de las trazas

Los valores de demora se capturaron utilizando la clase FlowMonitor. El módulo FlowMonitor tiene como objetivo proveer un sistema flexible para medir el desempeño de protocolos de red. Este utiliza “sondas”, instaladas en los nodos de red, para rastrear los paquetes intercambiados por los nodos, y medir un conjunto de parámetros. Los paquetes son divididos de acuerdo al flujo que pertenecen, donde cada uno es definido de acuerdo a las características de las sondas, por ejemplo, para IP se define un flujo para paquetes con el mismo protocolo, dirección IP/puerto fuente y dirección IP/puerto de destino (ns-3 project, 2014).

Las estadísticas que son recolectadas para cada flujo pueden ser exportadas en formato XML. Además, el usuario puede acceder a las sondas directamente para solicitar estadísticas específicas de cada flujo (ns-3 project, 2014).

Por defecto, FlowMonitor muestra solamente la demora total de todos los paquetes en un flujo. Debido a la necesidad de conocer la demora de cada paquete individual, se modificó el código fuente de la clase mencionada, en particular el método ReportLastRx, imprimiendo la salida la variable *delay* para cada flujo, cada vez que dicho método es llamado.

Dado que NS-3 no soporta la utilización de dos EPC, se realizaron dos simulaciones: una para el DL y otra para el UL. En estas se asignaron diferentes valores al generador de números aleatorios (*Random Number Generator*, RNG) con el objetivo de garantizar

la no correlación entre ambas corridas, siendo las distribuciones de los eventos (y de *handovers*), por tanto, diferentes.

Con el objetivo de añadir la demora del *core* de redes reales que utilizan servicios de VoIP, se le agregaron a la suma de las demoras del UL y el DL 81.27 ms, a los cuales se les descontaron 20 ms, de los 101.27 ms especificados en (UIT-T G.1050, 2007), procedentes de la demora del enlace punto a punto entre la P-GW y el RH en el UL y el DL.

Como el flujo de media en la simulación es un flujo continuo y los algoritmos de reproducción del *buffer* utilizados ajustan solamente en intervalos de silencio, es necesario agregar silencios a la traza de salida. Esto se hace de acuerdo a (UIT-T P.59, 1993).

Dado que NS-3 no siempre tiene en cuenta la contribución del tiempo de desconexión provocado por el *handover* a la demora obtenida mediante FlowMonitor, se realizó una inserción de 120 ms de demora ante la ocurrencia de este radio evento, según lo propuesto en (Lee and Kim, 2014), a los paquetes que se reciben luego del HO. Esto equivale a una transmisión de 6 paquetes, los cuales son considerados que se transmiten a la misma vez una vez reestablecida la comunicación entre el eNB y el UE con una diferencia de 20 ms entre uno y otro.

## 2.4. Algoritmos de reproducción

Dado que no es posible encontrar un tamaño fijo óptimo para el *buffer* de reproducción en aplicaciones de voz sobre IP, las soluciones se mueven en torno al uso de un *buffer* adaptativo. La mayoría de los algoritmos de *buffer* de reproducción adaptativo descritos en la literatura dependen de anticipar, con la mayor exactitud posible, la demora en la red para calcular el instante de reproducción. Le es imprescindible reaccionar rápidamente a cambios persistentes en el rendimiento de la red (ágil) pero ignorar los transientes en condiciones de ruido (estable). En resumen, deben establecer un compromiso entre la demora y la compensación de la variación de la demora, basando su funcionamiento en el historial del retraso de los paquetes.

Los algoritmos utilizados en esta tesis para el trabajo con el *buffer* de reproducción son seis, cuyas principales características se exponen a continuación.

### 2.4.1. Algoritmos de Ramjee

Aunque se han propuesto muchas formas para el cálculo del instante de reproducción, el trabajo presentado en (Ramjee *et al.*, 1994) es uno de los más explorados. Para este algoritmo el ajuste de parámetros se realiza entre ráfagas de audio. Para el primer

paquete de cada ráfaga la demora de extremo a extremo se obtiene a partir de 2.3.

$$D_0 = d_i + \beta v_i \quad (2.3)$$

Donde  $d_i$  es la demora estimada para el siguiente paquete,  $v_i$  es la variación de la demora y  $\beta$  es el factor de seguridad usado para definir la relación de compromiso entre la latencia de extremo a extremo y la pérdida de paquetes. Su valor fue fijado originalmente a 4.

Para los demás paquetes de esa misma ráfaga la demora de extremo a extremo se obtiene por medio de 2.4, donde  $t_i$  es una marca de tiempo colocada por el transmisor al paquete.

$$D_i = t_i + D_0 \quad (2.4)$$

Para obtener el estimado del retardo, Ramjee propuso cuatro predictores. Después de obtener el estimado de la demora empleando uno de los cuatro algoritmos de Ramjee, la variación de la latencia para cualquiera de ellos se calcula según la expresión 2.5, en la que  $n_i$  es la demora de red experimentada por el paquete  $i$ , y el factor de peso  $\alpha$  es seleccionado de acuerdo al método que se emplee.

$$v_i = \alpha v_{i-1} + (1 - \alpha) |d_i - n_i| \quad (2.5)$$

En el primero de los algoritmos de Ramjee (algoritmo 1) la demora para el paquete  $i$  es calculada usando un mecanismo propuesto por (Jacobson, 1988) para estimar el tiempo RTT (*Round-Trip-Time*) en conexiones TCP. Este algoritmo es básicamente un filtro recursivo lineal que se caracteriza por el factor  $\alpha$ , que pondera las observaciones pasadas o las recientes.

El segundo algoritmo (algoritmo 2) es una pequeña modificación del primero que tiene como objetivo lograr que el mecanismo reaccione más rápidamente cuando la latencia en la red se incrementa. La idea es usar diferentes valores de  $\alpha$ , uno para la tendencia hacia el incremento del retardo, y el otro para la tendencia a la disminución del retardo.

El tercer algoritmo (algoritmo 3) tiene un enfoque un poco diferente. Se fundamenta en utilizar la demora mínima de todos los paquetes recibidos en la ráfaga de audio anterior ( $k-1$ ) como la demora medio.

El cuarto algoritmo (algoritmo 4) incluye la detección de picos de latencia. Un pico constituye un aumento repentino y elevado del retraso de red de extremo a extremo y está seguido de una serie de paquetes que llegan casi simultáneamente, conduciendo a la finalización del pico.

Los primeros algoritmos no se adaptan lo suficientemente rápido en el caso de los picos, tomando mucho tiempo para aumentar sus estimaciones del retardo y para decremen- tarlas cuando el pico se acaba. Sin embargo el algoritmo 4, funciona según dos modos de operación, dependiendo de si se ha detectado un pico. Para cada paquete que llega al receptor, el algoritmo comprueba el modo actual y en caso necesario, cambia su modo. Si un paquete llega con un retardo mayor que el umbral dado (por ejemplo un cierto múltiplo del actual retardo), el algoritmo cambia al modo de pico. La detección de la terminación del pico hace uso de una variable “var” para ajustar la pendiente del pico. Cuando su valor es bastante pequeño, el algoritmo regresa al modo normal.

En la figura 2-5 se muestran los pseudocódigos para los cuatro algoritmos de Ramjee tratados anteriormente.

<p><math>n_i</math> : demora de red para el paquete <math>i</math>.  <math>\alpha = 0,998002</math>.  <math>\beta = 0,75</math>.</p> <p><b>Algoritmo 1</b>  <math>\hat{d}_i = \alpha \hat{d}_i + (1 - \alpha)n_i</math></p> <p><b>Algoritmo 2</b>            if <math>n_i &gt; \hat{d}_i</math>  <math>\hat{d}_i = \beta \hat{d}_i + (1 - \beta)n_i</math>            else  <math>\hat{d}_i = \alpha \hat{d}_i + (1 - \alpha)n_i</math>            end</p> <p><b>Algoritmo 3</b>            Para el primer paquete de una ráfaga  <math>\hat{d}_i = n_i</math>            Para todo paquete recibido            if <math>n_i &lt; \hat{d}_i</math>  <math>\hat{d}_i = n_i</math>            end</p>	<p><math>n_i</math> : demora de red para el paquete <math>i</math>.  <math>\alpha = 0,785</math>.</p> <p><b>Algoritmo 4</b>            if <i>normal_mode</i>            if <math> (n_i - n_{i-1})  &gt; 2\hat{d}_i + 800</math>  <math>var = 0</math>  <math>normal\_mode = false</math>            end  <math>\hat{d}_i = \alpha \hat{d}_i + (1 - \alpha)n_i</math>            else  <math>var = \frac{var}{2} + \frac{ 2n_i - n_{i-1} - n_{i-2} }{8}</math>            if <math>var \leq 63</math>  <math>normal\_mode = true</math>            else  <math>\hat{d}_i = \hat{d}_i + n_i - n_{i-1}</math>            end            end</p>
-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Figura 2-5: Pseudocódigos para los algoritmos 1, 2, 3 y 4 de Ramjee (López *et al.*, n.d.).

### 2.4.2. Algoritmo 5

El algoritmo 5, presentado en (López *et al.*, n.d.), adapta dinámicamente  $\beta$  para lograr una mejor calidad. El valor de  $\beta$  es calculado para el primer paquete de una ráfaga como:

$$\beta = \min \left( \max \left( \frac{a}{\hat{d}_i}, 1 \right), 10 \right) \quad (2.6)$$

donde  $a$  es una constante.

La idea es reducir los valores de  $\beta$  con el incremento de la demora estimada de la red. La estimación de la demora de la red es la misma que la utilizada en el algoritmo 2. Los valores del coeficiente son acotados entre 1 y 10 (López *et al.*, n.d.).

En la figura 2–6 se muestra el pseudocódigo del algoritmo 5.

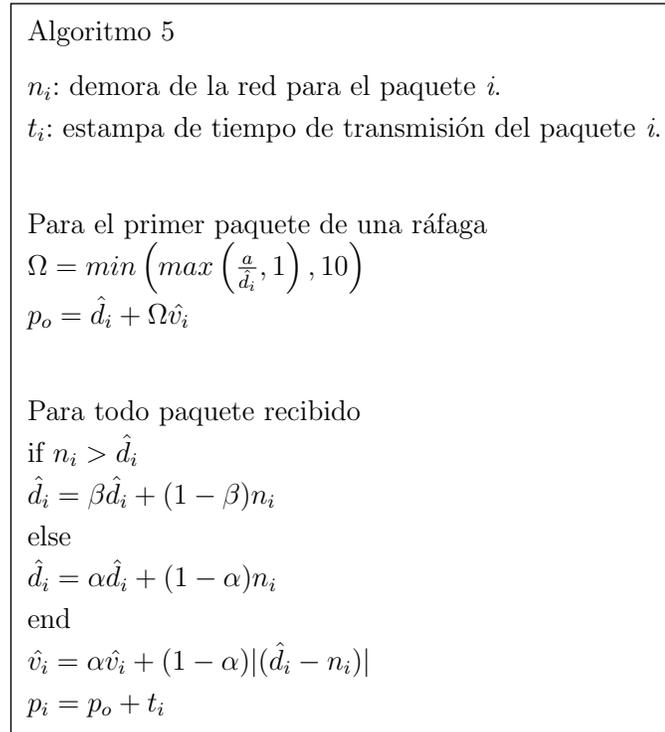


Figura 2–6: Pseudocódigo del algoritmo 5 (López *et al.*, n.d.).

### 2.4.3. Algoritmo eEM

El algoritmo eEM opera al igual que los algoritmos anteriormente descritos entre ráfagas, o sea, el *buffer* es ajustado durante los períodos de silencio maximizando el Factor R esperado para la siguiente ráfaga (Lobina, n.d.).

La figura 2–7 muestra los bloques principales involucrados en este algoritmo: las estadísticas relevantes para las pérdidas y la demora son predichas por medio de los paquetes anteriormente enviados, basado en esta información, el ajuste del *buffer* de reproducción se realiza de tal manera que se maximice la calidad conversacional esperada durante unidades conversacionales futuras (Lobina, n.d.).

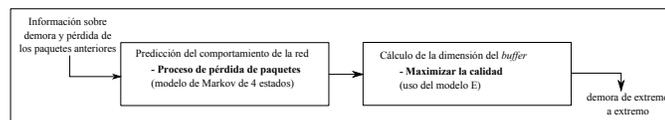


Figura 2–7: Muestra de la estrategia de reproducción del algoritmo 6 (Lobina, n.d.).

## 2.5. Conclusiones parciales

1. Con el simulador de eventos discretos NS-3 se puede simular una red LTE con un alto grado de detalle. Mediante el proyecto LENA se puede modelar tanto la interfaz de

radio como el EPC. Esto permite una correcta evaluación de los protocolos, interfaces y señalización de interés para el trabajo.

2. Dado que el código fuente de NS-3 es completamente accesible, se puede modificar para obtener la demora de cada paquete en el flujo de media, lo cual permite la generación de trazas de VoLTE.
3. Es necesario la inserción de períodos de silencios a las trazas de voz debido a que los seis algoritmos tratados en la investigación ajustan el tamaño del *buffer* de *de-jitter* entre ráfagas.
4. Teniendo en cuenta que los algoritmos de reproducción investigados presentan características diferentes, no se puede establecer la supremacía de uno sobre otro, a menos que la evaluación se realice empleando la misma traza.

## CAPÍTULO 3

# RESULTADOS Y DISCUSIÓN

En este capítulo se presenta una traza de VoLTE obtenida mediante el simulador de eventos discretos NS-3, así como la resultante de la inserción de intervalos de silencio. En la traza se observa la ocurrencia de *handovers*, radio eventos propios y fundamentales en la movilidad de la red LTE. Estos radio eventos introducen picos de demora asociados a la interrupción del flujo de datos durante su etapa de ejecución debido a que son *handovers* duros.

Las variaciones de la demora normalmente son tratadas a través del *buffer* de reproducción, implementando determinados algoritmos de predicción para estimar el retardo que se debe aplicar en cada ráfaga. Bajo este principio, se realiza en primer lugar un análisis del desempeño de seis algoritmos de reproducción (que no tienen en cuenta la futura ocurrencia de los HO) en términos de calidad empleando la traza de VoLTE obtenida en la simulación. Posteriormente, se procede a analizar nuevamente el comportamiento de dichos algoritmos, teniendo en cuenta la detección de picos que provee la anticipación a la ocurrencia de *handovers*. Finalmente, se ejecuta un análisis comparativo de la calidad percibida por el usuario entre ambos casos.

La evaluación se realiza usando el Modelo E a partir de los parámetros de demora y pérdida de paquetes. En el trabajo se calculó el factor  $R$  a partir de los parámetros promedios para el tiempo de conversación. Sin embargo, este valor promedio no arroja claridad en cuanto al comportamiento durante toda la conversación. Por esta razón, se calculó además el factor  $R$  “instantáneo” dividiendo el tiempo de conversación en segmentos de 10 s (correspondientes a 500 paquetes de 20 ms cada uno) y se computó el factor  $R$  a partir de los parámetros medios de cada segmento. El valor del factor  $R$  se hace coincidir con un valor de MOS según la correspondencia descrita en el capítulo 1, de modo que se obtienen valores de MOS instantáneos.

### 3.1. Trazas de VoLTE

La traza que se obtuvo como resultado de la implementación del escenario creado para LTE en el simulador NS-3 se muestra en la figura 3–1, luego de sumar las demoras de

los enlaces de bajada y subida y adicionar la demora del *core*. La traza contiene 38 *handovers*, distribuidos 19 en el UL y 19 en el DL.

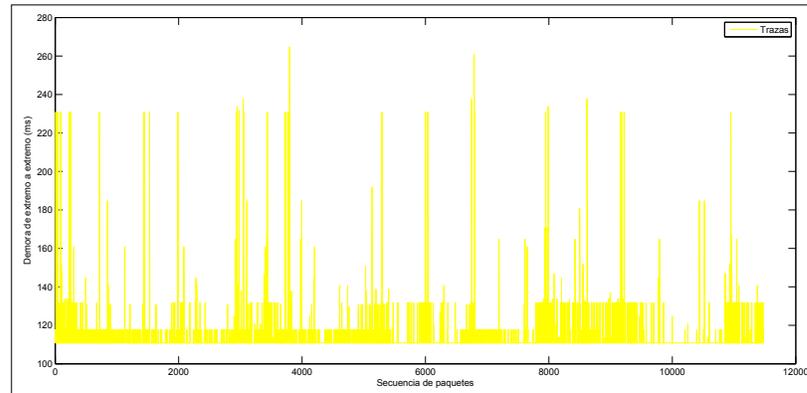


Figura 3–1: Traza de VoLTE obtenida mediante simulación.

Como se puede apreciar, la demora de extremo a extremo no rebasa normalmente los 200 ms, por lo que la calidad de servicio se estima que sea muy buena. Este patrón se rompe cuando ocurren los *handovers* ya que, sumando solamente la demora del *handover* (120 ms) en un sentido (DL o UL), y el *core* (101.27 ms), se superan los 200 ms de demora extremo a extremo, por lo que se compromete la calidad de servicio.

### 3.2. Adición de silencios

Los *buffer* de reproducción pueden adaptar su demora de dos maneras, la primera es intra-ráfaga, donde se modifica la demora de extremo a extremo durante la conversación, ya sea dentro del período de actividad o durante el de silencio y no necesariamente al inicio de la ráfaga. Por otro lado se encuentra la vertiente inter-ráfaga, donde el cambio del tamaño del *buffer* se produce solo durante los intervalos de silencio. Cualquier variación en la demora de reproducción es introducida por la expansión o compresión artificial de los períodos de silencio al comienzo de la ráfaga.

Los algoritmos para el *buffer* de reproducción investigados ajustan la demora de acuerdo a la variante inter-ráfaga, por lo que resulta necesario añadir a la traza de manera artificial intervalos de silencio propios de una conversación. Esta inserción se realizó de acuerdo a la recomendación de la UIT (UIT-T P.59, 1993), obteniéndose como resultado, la pérdida de algunos *handovers* al coincidir con la posición de algunos intervalos de silencios. En la figura 3–2 se muestra la traza anteriormente mostrada con los intervalos de silencios insertados.

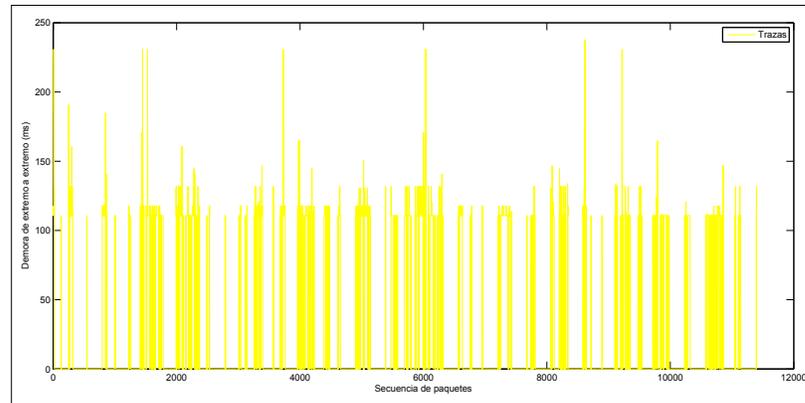


Figura 3-2: Traza con períodos de silencio insertados.

### 3.3. Evaluación de los algoritmos de reproducción sin detección de picos

En la figura 3-3 se expone la actuación de los seis algoritmos mencionados en el capítulo anterior sobre la traza de voz comentada con anterioridad. Estos algoritmos realizan una estimación de la demora basándose en la latencia de los paquetes recibidos, aplicando criterios sobre el peso de las demoras más recientes o más antiguas en dependencia de las particularidades de cada uno.

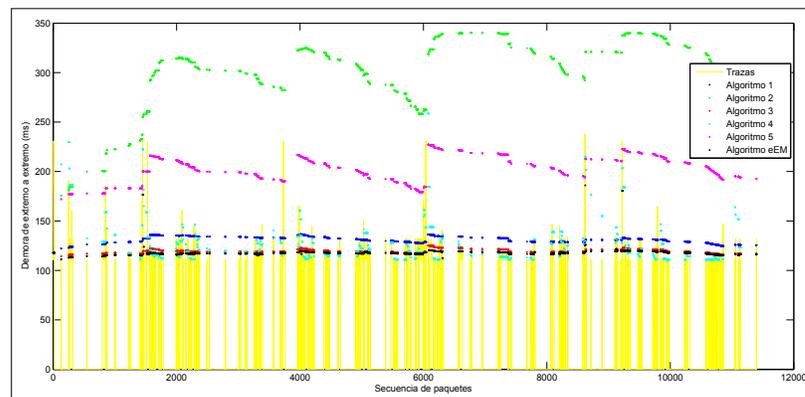


Figura 3-3: Comportamiento de los 6 algoritmos de reproducción evaluados sin utilizar detección de picos.

En las figuras 3-4 y 3-5 se muestran los comportamientos del algoritmo 1 de Ramjee. El desempeño del algoritmo 3 de Ramjee se expone en las figuras 3-6 y 3-7, mientras que se puede observar el funcionamiento del algoritmo eEM en las figuras 3-8 y 3-9.

Tras el análisis del comportamiento de los algoritmos representados en las figuras de la 3-4 a la 3-9, se comprobó que los tres algoritmos tienen un comportamiento lento

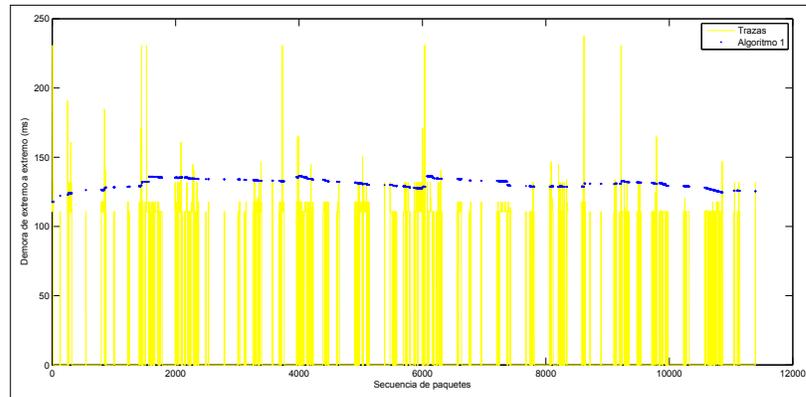


Figura 3-4: Comportamiento del algoritmo 1 de Ramjee.

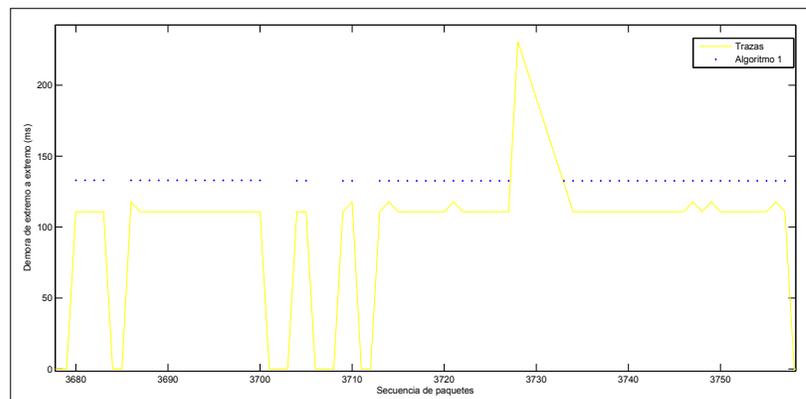


Figura 3-5: Amplificación de la figura 3-4.

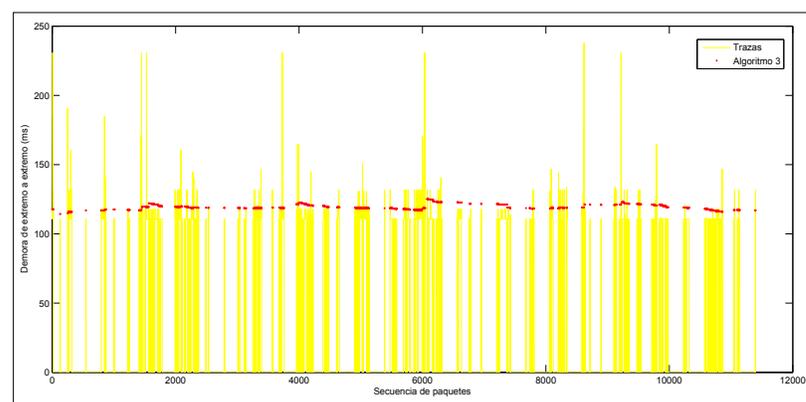


Figura 3-6: Comportamiento del algoritmo 3 de Ramjee.

y estable. Estos son algoritmos muy ajustados a la hora de establecer el tamaño del

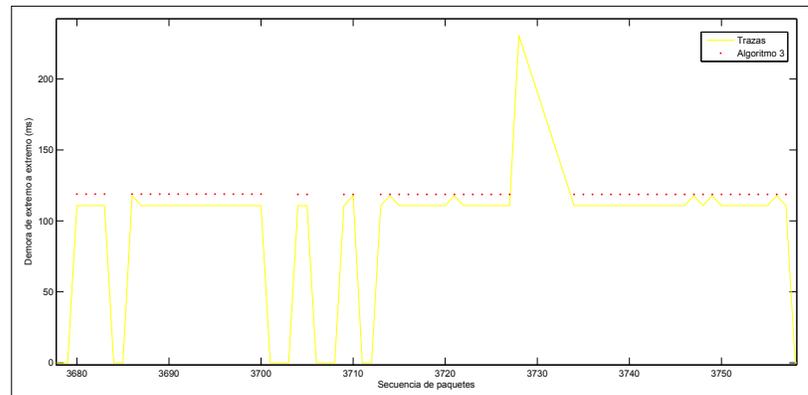


Figura 3-7: Amplificación de la figura 3-6.

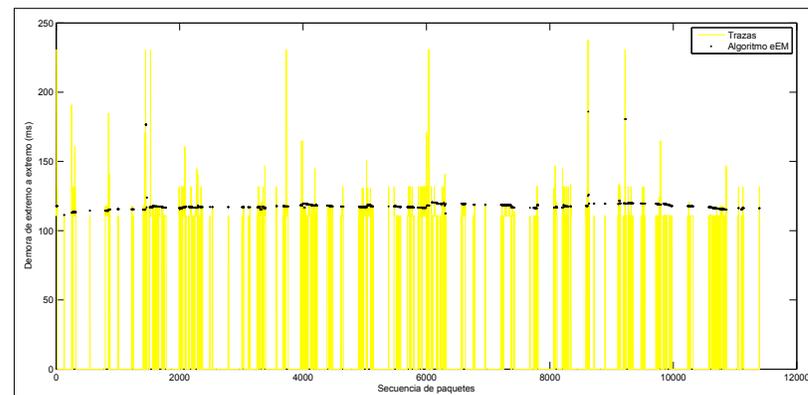


Figura 3-8: Comportamiento del algoritmo eEM.

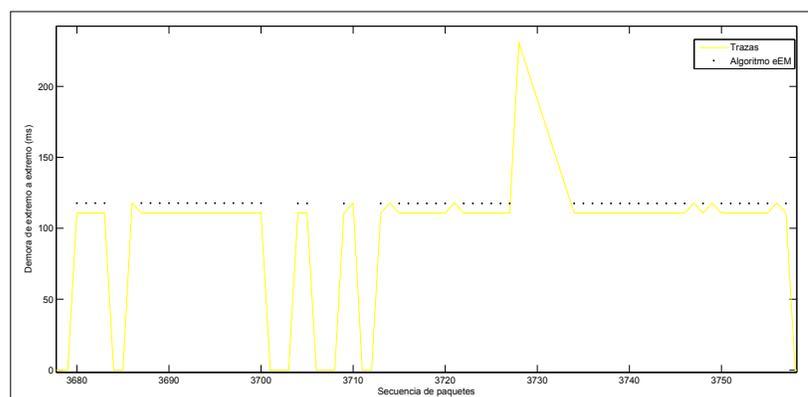


Figura 3-9: Amplificación de la figura 3-8.

*buffer*, tienen la idea de incorporar la menor cantidad de demora posible. Para el caso

del algoritmo eEM, se comprueba que es el más estricto a la hora de ajustar el tamaño del *buffer*, permaneciendo muy cercano a los valores de la demora real de los paquetes cuando no ocurren grandes variaciones. Ante picos muy elevados este algoritmo reacciona rápidamente en algunas ocasiones, con la particularidad de que retorna a valores inferiores muy rápidamente, lo cual indica un comportamiento poco conveniente ante picos de retardo esporádicos y de corta duración. Se manifiesta su calidad de conferir mucho peso a observaciones anteriores de la demora. Por su parte, los algoritmos 1 y 3 se diferencian en que el 1 realiza un ajuste menos estricto del tamaño del *buffer* que el 3, pero ambos poseen la cualidad de no responder con agilidad ante los aumentos de demora. Estas características hacen que ante fluctuaciones o picos en la latencia mayores, como puede ser la ocurrencia de un *handover*, no sean efectivos en el tratamiento y pierdan paquetes, provocando un impacto negativo en la calidad percibida.

En la figura 3–10 se refleja el desempeño del algoritmo 2 de Ramjee.

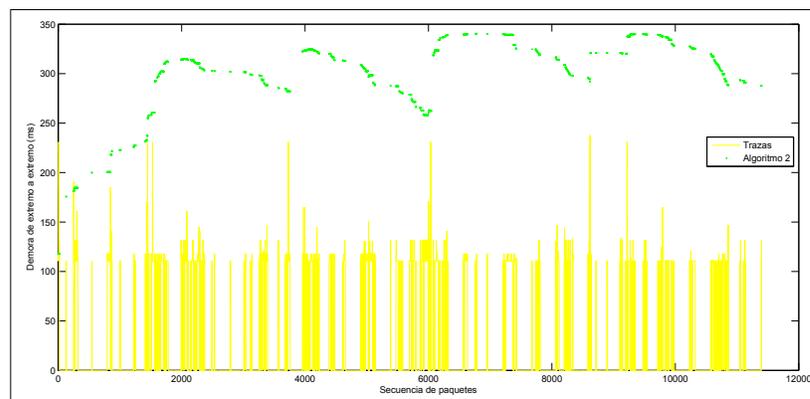


Figura 3–10: Comportamiento del algoritmo 2 de Ramjee.

El algoritmo 2 de Ramjee no desecha paquetes incluso cuando ocurren los *handovers*, sin embargo, la mayoría del tiempo hace una sobreestimación de la demora, introduciendo valores de la misma en ocasiones superiores a los 300 ms, lo cual provoca una caída en la calidad de servicio en la aplicación de VoIP. Este comportamiento viene dado por las características propias del mismo. Este algoritmo usa un filtro AR con características de estabilidad siempre que sea posible, pero si se detecta incrementos significativos de la demora (presencia de un pico), conmuta a un filtro AR ágil que le permite seguir el pico. Una vez que ha terminado, regresa al filtro AR estable, por lo que se comienza a sobreestimar durante la latencia hasta que se logra descender del pico. Esta sobreestimación se refleja en una caída del MOS.

El comportamiento del algoritmo 4 de Ramjee para la traza de VoLTE es mostrado en las figura 3–11 y 3–12.

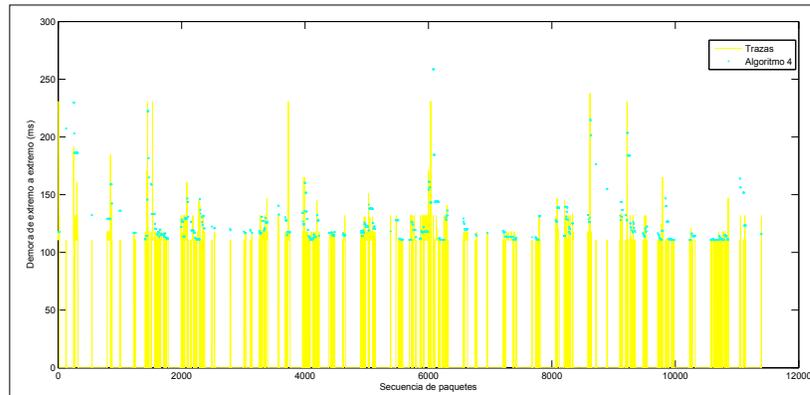


Figura 3–11: Comportamiento del algoritmo 4 de Ramjee.

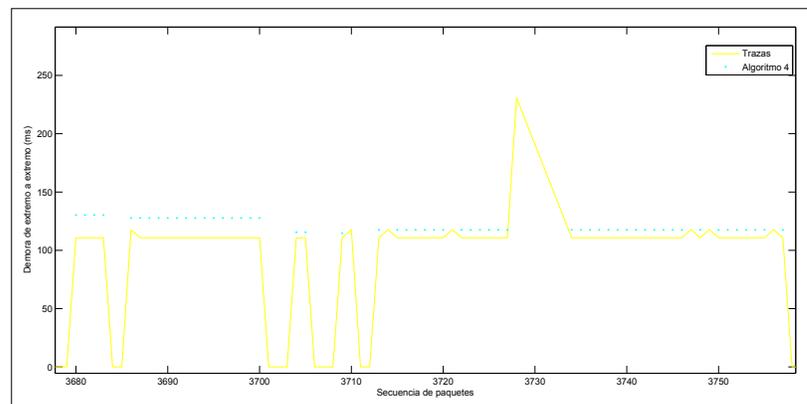


Figura 3–12: Amplificación de la figura 3–11.

Este algoritmo le confiere gran peso a la hora de estimar el tamaño del *buffer* a los valores recientes de retardo, siendo capaz de responder a cambios repentinos de una forma aceptable. Si bien tiene un desempeño ágil e inestable, tampoco es capaz de seguir los picos de *handover*.

Por último, se muestra el comportamiento del algoritmo 5 en las figuras 3–13 y 3–14.

Del análisis del comportamiento se concluyó que este algoritmo es estable y capaz de responder con rapidez a los picos de *handovers*, sin embargo, la precisión del ajuste es limitada en el sentido de que el mismo no supera los 200 ms, lo cual es insuficiente para evitar las pérdidas de paquetes provocadas por los HO. Luego de que ocurre un pico

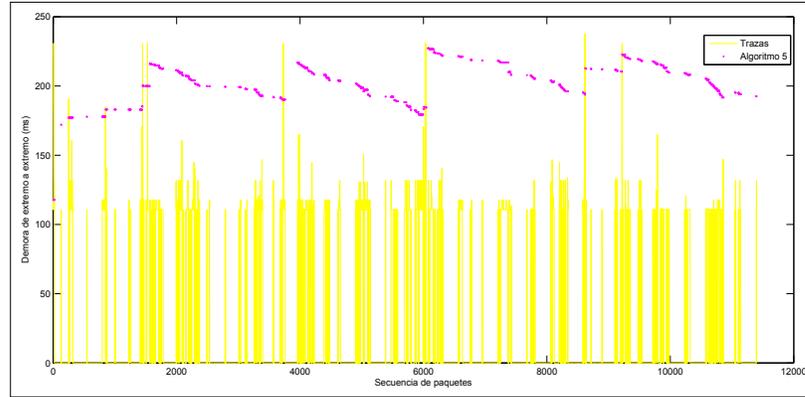


Figura 3–13: Comportamiento del algoritmo 5.

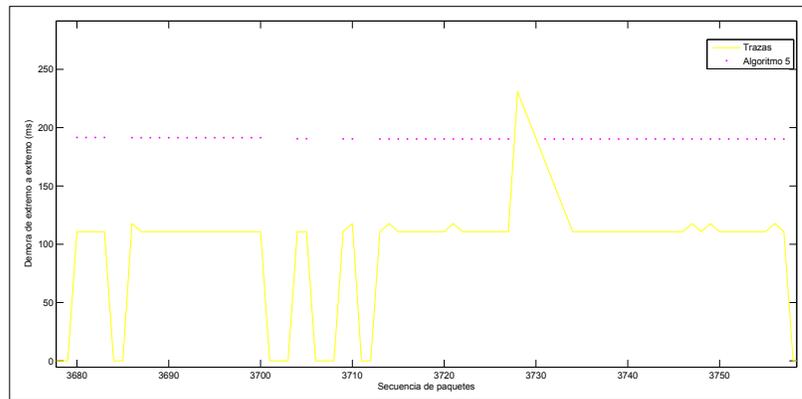


Figura 3–14: Amplificación de la figura 3–13.

de demora relacionado con la presencia del *handover* su recuperación es lenta pero en menor medida con respecto al 2.

Los algoritmos anteriores demostraron que incluso para aquellos que poseen un comportamiento ágil en el tratamiento de picos de la demora, siempre van a existir pérdidas de paquetes asociadas con la ocurrencia de *handovers*, que a la postre repercuten negativamente en la calidad percibida por el usuario. Obviamente este comportamiento se debe a que a los algoritmos les resulta imposible, debido a la naturaleza de su funcionamiento, estimar con antelación la ocurrencia de un pico de demora tan elevado e inesperado como el provocado por un HO.

A continuación se muestra el MOS instantáneo que fue obtenido para los seis algoritmos de *buffer* de reproducción, representando 23 muestras de MOS medidas cada una en intervalos de 10 s sobre segmentos de 500 paquetes de 20 ms de duración.

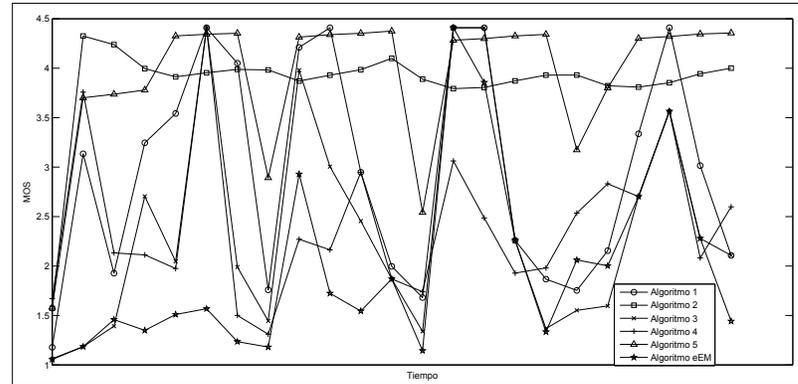


Figura 3–15: MOS instantáneo para los 6 algoritmos evaluados.

Al principio de la conversación los predictores comienzan subestimando la demora, debido a que este parámetro y la variación de la demora se fijan inicialmente en cero, lo que hace que el *buffer* pierda muchos paquetes. Por tanto, los valores del MOS se afectan hasta que se alcanza el período de estabilidad, esta es la causa de que todos los algoritmos inician con valores modestos de MOS.

Es necesario destacar que, independientemente del desempeño de los algoritmos para la traza analizada, en todos los casos existieron caídas de la calidad correspondiente a la ocurrencia de los *handovers*. En la figura 3–16 se encuentra señalada una de las caídas correspondiente a la muestra 8 de MOS instantáneo. Esta muestra es aplicada a los paquetes del 3501 al 4000, destacados en la figura en el intervalo encerrado entre líneas rojas. Los valores de esta caída se muestran en la tabla 3–1 para las muestras 7, 8 y 9 de todos los algoritmos.

Tabla 3–1: MOS instantáneo de las muestras 7, 8 y 9 de todos los algoritmos

Algoritmo	MOS instantáneo muestra 7	MOS instantáneo muestra 8	MOS instantáneo muestra 9
1	4.0508	1.7595	4.2109
2	3.9879	3.9825	3.8702
3	1.9900	1.4453	3.9794
4	1.5011	1.3112	2.2706
5	4.3528	2.8926	4.3146
6	1.2358	1.1811	2.9291

Estas caídas se deben a la pérdida de paquetes provocada por la imposibilidad de ajuste de los algoritmos ante la ocurrencia de un *handover*. En el caso del algoritmo 2 no se manifiesta una caída debido a que este sobrestima la demora del *buffer* de reproducción

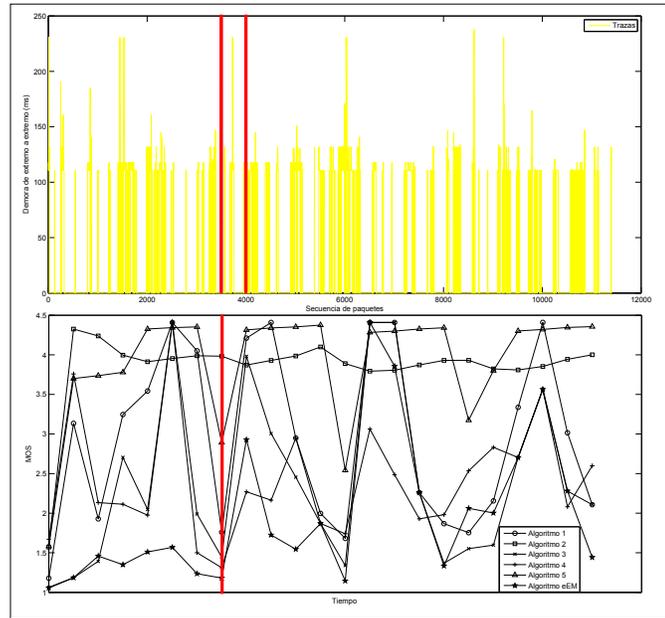


Figura 3-16: MOS instantáneo para los 6 algoritmos evaluados.

e impide que ante la ocurrencia de picos los paquetes no se pierdan, por supuesto esta sobrestimación influye negativamente en la calidad percibida.

Por lo tanto, se respalda la idea del impacto negativo que posee la ocurrencia del radio evento *handover* en la calidad de la aplicación de voz y la incapacidad de los algoritmos del *buffer* de reproducción para tratarlo con eficiencia.

En la tabla 3-2 se exponen los valores MOS promedio correspondientes a cada algoritmo. Aunque este valor no provee de una caracterización pormenorizada de la calidad, sí resulta útil a la hora de tener un criterio sobre el comportamiento general del desempeño de los algoritmos con la traza en cuestión.

Tabla 3-2: Valores MOS promedio correspondientes a los seis algoritmos

Algoritmo	MOS promedio
Y1	2.9659
Y2	3.8478
Y3	2.3970
Y4	2.4190
Y5	3.9213
Y6	1.9856

Del análisis de la tabla anterior se concluyó que el mejor desempeño, utilizando el criterio del MOS promedio, corresponde al algoritmo 5, esta conclusión se respalda en

el comportamiento del MOS instantáneo que mantiene sus valores, con excepción de los casos de *handover*, en valores de calidad elevados.

### 3.4. Evaluación de los algoritmos de reproducción con detección de picos

Para implementar la antelación, se realizó un análisis de la señal de referencia de potencia recibida de la celda de servicio y la celda candidata mediante los reportes periódicos de mediciones enviados por el UE a la eNodeB. Tras el análisis se observó una tendencia al incremento antes de los HO, de la diferencia de la RSRP de ambas celdas, hasta que se cumple el criterio de histéresis y TTT y se ejecuta el evento. Por lo tanto, utilizando esta tendencia, se puede anticipar la ocurrencia de un *handover*. Con este conocimiento, los algoritmos pueden saber con antelación cuándo deben aumentar el tamaño del *buffer* para enfrentar el HO. En la figura 3–17 se muestra un ejemplo de la tendencia mencionada.

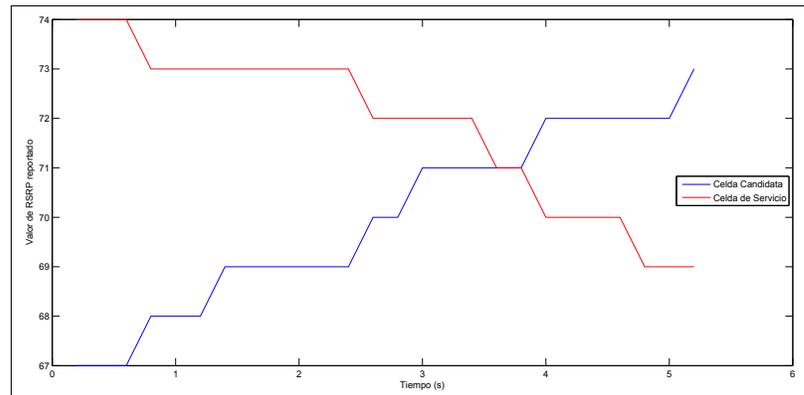


Figura 3–17: Valor reportado de RSRP para celda de servicio y objetivo antes de un *handover*.

Haciendo un estudio de las simulaciones realizadas y de la caracterización del radio evento, se determinó adecuado utilizar el valor de 250 ms como tamaño máximo que el *buffer* debe adquirir cuando ocurre un HO, ya que este es un valor que impide que se pierdan paquetes sin exagerar la demora del *buffer* de reproducción. En la figura 3–18 se muestra el desempeño de los algoritmos de reproducción teniendo en cuenta el conocimiento previo de la ocurrencia de los *handovers*.

Dado que el comportamiento de los algoritmos cuando se utiliza detección de picos es igual a cuando no se utiliza la misma (excepto cuando ocurren *handovers*), se muestra en las figuras de la 3–19 a la 3–24 solamente el desempeño en la vecindad de un HO.

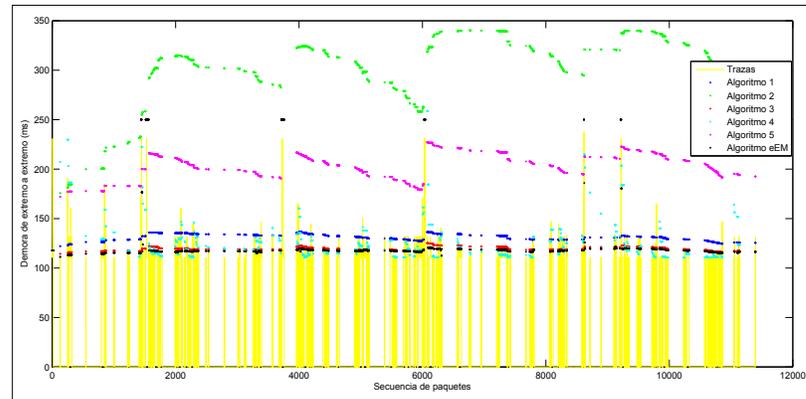


Figura 3–18: Desempeño de los algoritmos de reproducción utilizando detección de picos.

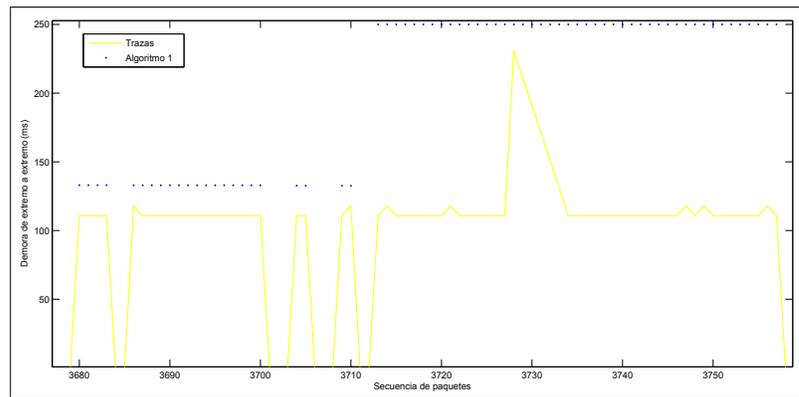


Figura 3–19: Desempeño del algoritmo 1 de Ramjee empleando detección de picos.

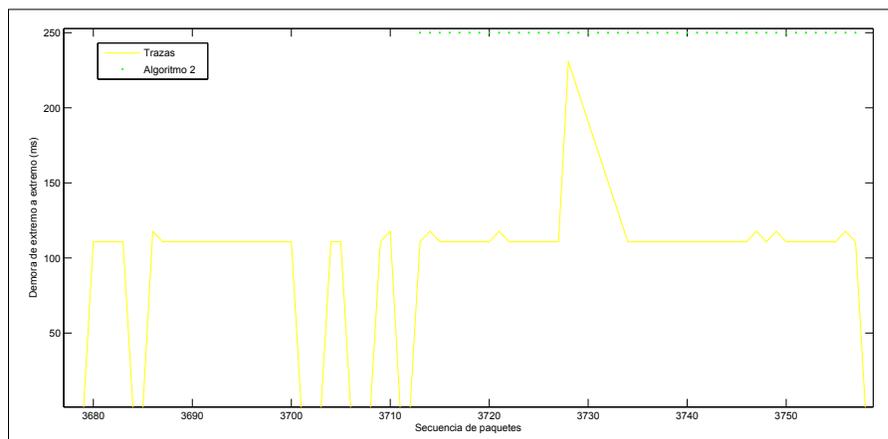


Figura 3–20: Desempeño del algoritmo 2 de Ramjee empleando detección de picos.

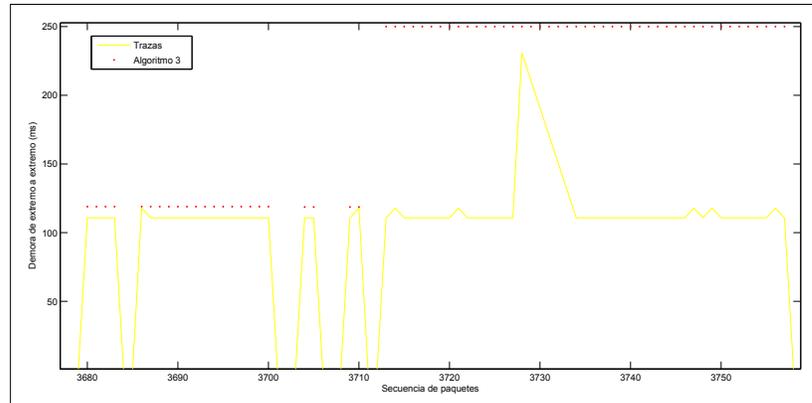


Figura 3–21: Desempeño del algoritmo 3 de Ramjee empleando detección de picos.

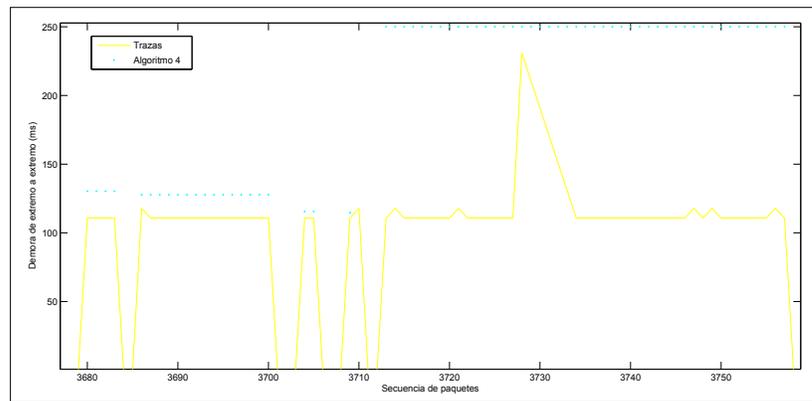


Figura 3–22: Desempeño del algoritmo 4 de Ramjee empleando detección de picos.

Como se puede apreciar, todos los algoritmos fueron capaces de aumentar la demora del *buffer* hasta el valor del pico del *handover*, por lo que no ocurrieron pérdidas de paquetes y la calidad durante estos eventos se ve afectada en un valor mínimo.

### 3.5. Comparación del desempeño de los algoritmos: detección de picos vs. no detección de picos

Con el objetivo de cuantificar la calidad percibida por el usuario se calculó el MOS “instantáneo” a intervalos de 10 s que corresponden a 500 paquetes. Para valorar el impacto de la detección de picos de *handover* se procedió a realizar una comparación entre el desempeño de estos algoritmos con la implementación de la detección de picos y sin ella. En la figura 3–25 se observan simultáneamente las gráficas de MOS instantáneo para ambos procedimientos.

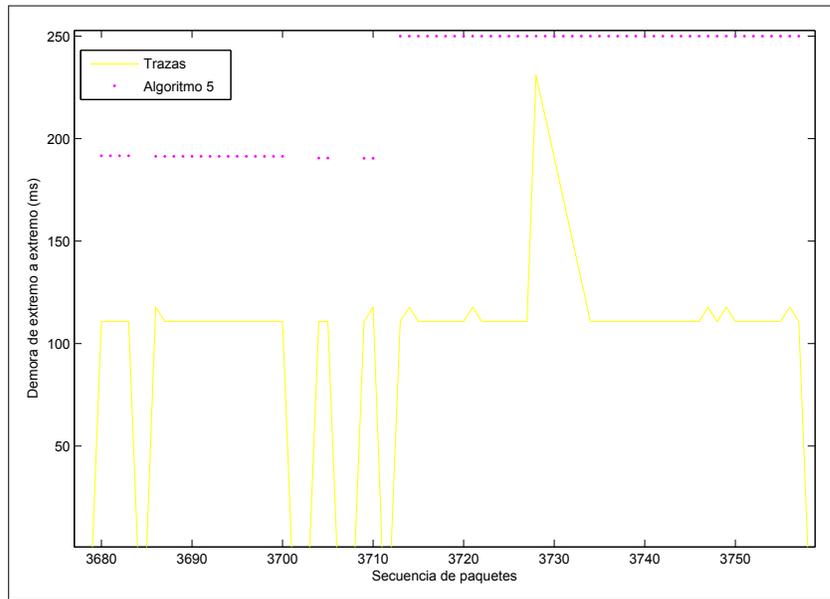


Figura 3–23: Desempeño del algoritmo 5 empleando detección de picos.

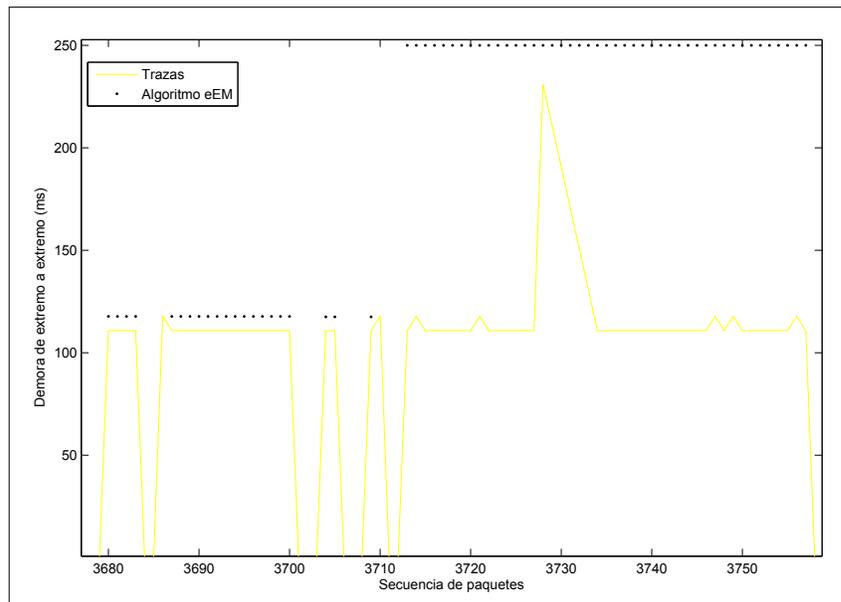


Figura 3–24: Desempeño del algoritmo eEM empleando detección de picos.

En la figura 3–25 se destacan las muestras 8 y 10 correspondientes a los paquetes del 3501 al 4000 para la muestra 8 y los paquetes del 4501 al 5000 para la muestra 10. Ambos intervalos se encuentran señalados en la figura, la muestra 8 con líneas rojas y la muestra 10 con líneas azules.

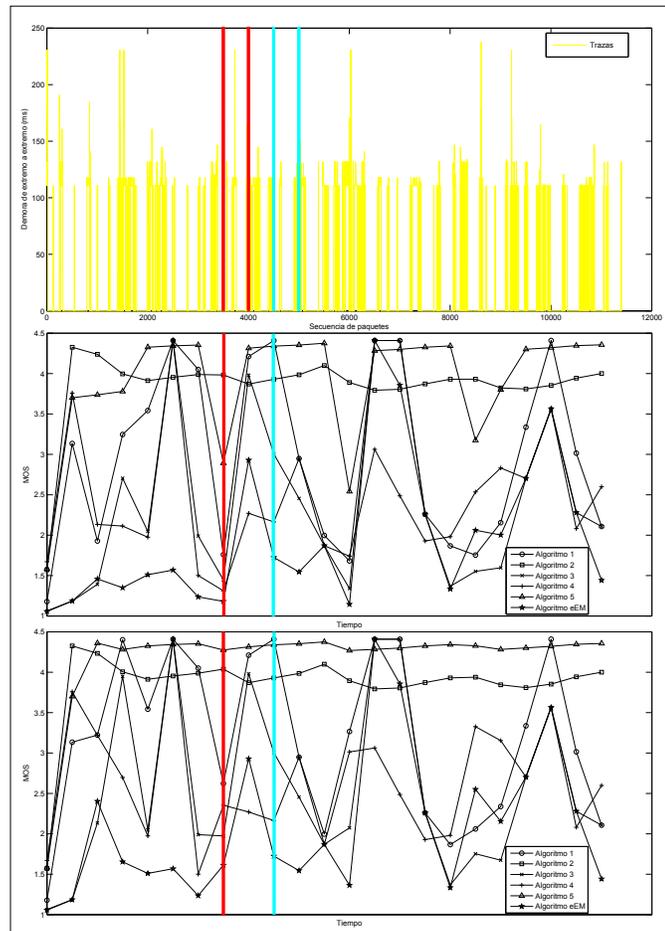


Figura 3–25: Comparación del valor instantáneo de la octava y décima muestra del MOS.

En la tabla 3–3 se muestra la comparación de las muestras MOS instantáneo 8 y 10 de los algoritmos cuando utilizan detección de picos y cuando no anticipan la ocurrencia de los *handovers*, así como el MOS promedio.

Del análisis de la información se constató que la calidad percibida de la aplicación de voz aumentó para todos los algoritmos cuando se aplica la detección de picos ante la ocurrencia de un *handover*, lo cual se aprecia en la muestra 8. Sin embargo, para el caso de la muestra 10 en cuyo intervalo no ocurren *handovers*, se comprobó que no ocurren variaciones en el MOS instantáneo para los dos casos analizados. Debido a esto, se comprueba que el aumento del valor del MOS promedio cuando se emplea detección de picos, se debe únicamente a la calidad de anticipación ante un evento de *handover* y a su correspondiente implementación en los algoritmos de *buffer*.

El algoritmo 5, igual que en el caso sin detección, es el que mejor comportamiento posee en el tratamiento de la demora.

Tabla 3–3: Calidad instantánea (muestra 8 y 10) y calidad promedio de los seis algoritmos

Algoritmo	MOS instantáneo muestra 8		MOS instantáneo muestra 10		MOS promedio	
	sin detección	con detección	sin detección	con detección	sin detección	con detección
	Y1	1.7595	2.6228	4.4089	4.4089	2.9659
Y2	3.9825	4.0385	3.9289	3.9289	3.8478	3.8520
Y3	1.4453	1.9741	3.0047	3.0047	2.3970	2.5504
Y4	1.3112	2.3537	2.1637	2.1637	2.4190	2.6407
Y5	2.8926	4.2754	4.3394	4.3394	3.9213	4.1765
Y6	1.1810	1.6121	1.7246	1.7246	1.9856	2.0960

En la figura 3–26 se muestran los por cientos que representan la clasificación de la calidad de percepción de los usuarios para el algoritmo 5 cuando este funciona de la manera tradicional y cuando se implementa la detección de picos de *handover*, constatándose la mejora al introducir la detección de picos con respecto a cuándo no se implementa este mecanismo. Del total del tiempo de conversación se establece que: el tiempo en el que el usuario se encuentra muy satisfecho crece de un 30 % cuando no se implementa la detección a un 35 % cuando sí se aplica, el tiempo en que se encuentra satisfecho aumenta de un 35 % a un 57 %, el tiempo en el que se está algo insatisfecho se reduce de un 17 % a un 4 % (sin utilizar anticipación). La porción de la conversación en que se encuentra muy insatisfecho se mantiene en un 4 %, no existiendo para el caso en el que se usa detección el período no recomendado.

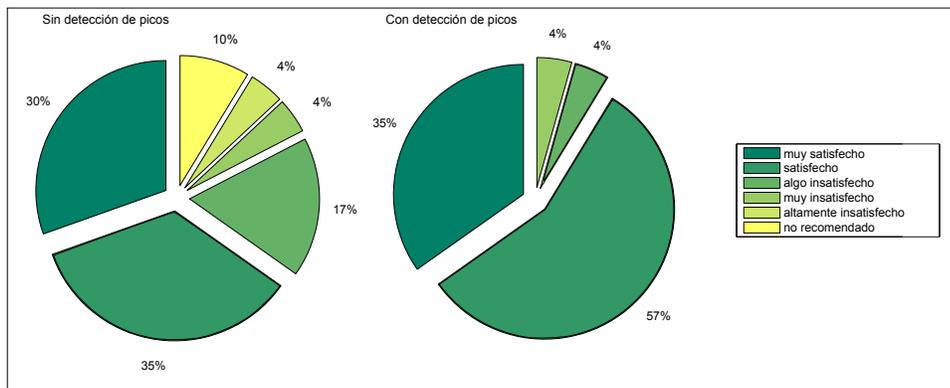


Figura 3–26: Comparación de la calidad percibida por el usuario para el algoritmo 5.

### 3.6. Conclusiones parciales

1. Los algoritmos de reproducción, al no utilizar detección de picos no son capaces de incrementar el tamaño del *buffer* hasta el valor de demora producido por los *handovers*, lo que provoca la pérdida de estos paquetes.
2. Ningún algoritmo mantiene valores de calidad satisfactorios durante la ocurrencia de *handovers* sin tener en cuenta la detección de picos.
3. Se puede utilizar una tendencia al incremento antes de los HO, de la diferencia de la RSRP de las celdas de servicio y candidata, para lograr que los algoritmos sean capaces de anticipar la ocurrencia de un *handover*.
4. Independientemente del tipo de algoritmo de *buffer* que se utilice, la calidad percibida por el usuario (MOS) en todos los casos aumenta cuando se utiliza el conocimiento previo de la ocurrencia de los *handovers*.

## CONCLUSIONES

Luego de realizar un análisis de los elementos investigados y de los resultados obtenidos, se arriba a las siguientes conclusiones:

1. La implementación de servicios de voz sobre redes IP, incluida la solución de VoLTE, enfrenta restricciones fuertes de demora, *jitter* y pérdida de paquetes, provocadas por la gran variabilidad que es característica de estas redes, siendo la transmisión de voz una aplicación de tiempo real, donde se generan los paquetes de audio a una razón constante. A partir de 200 ms de demora extremo a extremo, y para un número de paquetes perdidos superior al 2%, la calidad percibida por el usuario comienza a afectarse.
2. Se puede utilizar el conocimiento previo de la ocurrencia de los eventos de radio siguientes: *handovers*, continuidad de llamada de voz de radio único (SRVCC), falla del enlace de radio (RLF), razón de error de bloque (BLER) para el enlace de subida (UL) y de bajada (DL) y número de retransmisiones del UL, para ajustar preventivamente la demora del *buffer* de *de-jitter* en anticipación de pérdidas futuras predichas que tienen una alta probabilidad de ocurrencia.
3. La utilización del simulador de eventos discretos NS-3 permite generar trazas de voz sobre una red LTE, modelándose el núcleo de paquetes evolucionado y la interfaz de radio, así como los protocolos y conexiones de estos.
4. Cuando los algoritmos de reproducción funcionan de la manera tradicional (sin tener en cuenta el conocimiento previo de la ocurrencia de un *handover*), son incapaces de reaccionar a tiempo cuando ocurre un HO, por lo que no pueden aumentar el tamaño del *buffer* hasta un valor que absorba el pico de demora causado por el evento, lo que conlleva a la pérdida de los paquetes de este intervalo.
5. La calidad de servicio percibida por el usuario, evidenciada por muestras del MOS instantáneo y valores del MOS promedio, se incrementa para todos los algoritmos al utilizar realimentación a nivel de radio. El algoritmo 5 cuando no se utiliza detección de picos y en el caso contrario, presenta el mejor desempeño. Del total del tiempo de conversación se establece que: el tiempo en el que el usuario se encuentra muy satisfecho crece de un 30 % cuando no se implementa la detección a un 35 % cuando sí se aplica, el tiempo en que se encuentra satisfecho aumenta de un 35 % a un 57 %, el tiempo en el que se está algo insatisfecho se reduce de un 17 % a un 4 % (sin utilizar

anticipación). La porción de la conversación en que se encuentra muy insatisfecho se mantiene en un 4 %, no existiendo para el caso en el que se usa detección el período no recomendado.

## RECOMENDACIONES

Para complementar este trabajo y ampliar el área de investigación del mismo se recomienda:

1. Realizar un estudio de los eventos de radio no tratados en la tesis para determinar en qué medida afectan la demora y/o pérdida de paquetes y cómo se puede anticipar su ocurrencia.
2. Modificar el código del simulador NS-3 para que el cálculo de la demora siempre incluya el tiempo de interrupción causado por los *handovers*.
3. Desarrollar experimentos similares al realizado en la investigación para comprobar la mejora alcanzada utilizando la anticipación a los otros radio eventos.

## REFERENCIAS BIBLIOGRÁFICAS

- 3GPP TS 23.203 (2012). Technical Specification Group Services and System Aspects Policy and charging control architecture (Release 11).
- 3GPP TS 36.104 (2014). Evolved Universal Terrestrial Radio Access (E-UTRA): Base Station (BS) radio transmission and reception (Release 12).
- 3GPP TS 36.322 (2007). Evolved Universal Terrestrial Radio Access (E-UTRA) Radio Link Control (RLC) protocol specification (Release 8).
- 3GPP TS 36.331 (2014). Evolved Universal Terrestrial Radio Access (E-UTRA); Radio Resource Control (RRC) Protocol specification (Release 12).
- Agusti Comes, Ramon, Francisco Bernardo Alvarez, Fernando-jose Casadevall Palacio, Ramon Antonio Ferrus Ferre, Jorge Perez Romero and Jose Oriol Sallent Roig (2010). Lte: nuevas tendencias en comunicaciones móviles.
- Ahmad, Aftab (2005). *Wireless and mobile data networks*. John Wiley & Sons.
- Alfredsson, Stefan, Anna Brunstrom and Mikael Sternad (2008). Impact of 4g wireless link configurations on voip network performance. In: *Wireless Communication Systems. 2008. ISWCS'08. IEEE International Symposium on*. IEEE. pp. 708–712.
- Ati, Hurtado and Christian Geovanny (2011). Estudio de factibilidad para la implementación de LTE (Long Term Evolution) en el Ecuador. PhD thesis. QUITO/EPN/2011.
- Cox, Christopher (2012). *An introduction to LTE: LTE, LTE-advanced, SAE and 4G mobile communications*. John Wiley & Sons.
- Dahlman, Erik, Stefan Parkvall and Johan Skold (2013). *4G: LTE/LTE-advanced for mobile broadband*. Academic press.
- De Rango, Floriano, Mauro Tropea, Peppino Fazio and Salvatore Marano (2006). Overview on voip: Subjective and objective measurement methods. *International Journal of Computer Science and Network Security* **6**(1), 140–153.
- Farooq, KHAN (2009). Lte for 4g mobile broadband.
- Fontán, Natalia Pignataro-Fernando, Gustavo Bounous, Leticia Silva and Martín Rodríguez (2012). Modelado y planificación de redes lte.
- Gruber, John G (1981). Delay related issues in integrated voice and data networks. *Communications, IEEE Transactions on* **29**(6), 786–800.
- Helenius, Atte (2011). Performance of handover in long term evolution. *School of Electrical Engineering*.

- Hyttiä, Esa, Pasi Lassila and Jorma Virtamo (2006). Spatial node distribution of the random waypoint mobility model with applications. *Mobile Computing, IEEE Transactions on* **5**(6), 680–694.
- Iñiguez Chavarría, José Bruno et al. (2014). Lte handover performance evaluation based on power budget handover algorithm.
- Jacobson, Van (1988). Congestion avoidance and control. In: *ACM SIGCOMM computer communication review*. Vol. 18. ACM. pp. 314–329.
- Jin, Hu, Changgi Cho, Nah-Oak Song and Dan Keun Sung (2011). Optimal rate selection for persistent scheduling with harq in time-correlated nakagami-m fading channels. *Wireless Communications, IEEE Transactions on* **10**(2), 637–647.
- Joskowicz, José and Rafael Sotelo Bovino (2007). Medida de la calidad de voz en redes ip. *Memoria de Trabajos de Difusión Científica y Técnica* (5), 12–23.
- Lee, Hyun-Jin and Jae-Hyun Kim (2014). A handover mechanism considering the abstracted r-factor. *Vehicular Technology, IEEE Transactions on* **63**(6), 2687–2696.
- Lobina, Mirko Luca (n.d.). Voip streaming over packet-based networks.
- López, Carlos A Rodríguez, Vitalio Alfonso Reguera, Evelio MG Fernández and Walter Godoy Jr (n.d.). Classical playout buffer algorithm revisited.
- MAHMUD, SHAHREAR (2013). QOS PERFORMANCE ANALYSIS: DESIGN AND DEVELOPMENT OF VOICE AND VIDEO MOBILITY OVER LONG TERM EVALUATION (LTE) MODEL. PhD thesis. Blekinge Institute of Technology.
- Martínez, Leonardo Francisco Panes (2014). Gráficos de control multivariables aplicados a la estimación de la calidad percibida de la voz sobre ip. *Revista Ingeniería Industrial*.
- Mitsche, Dieter, Giovanni Resta and Paolo Santi (2014). The random waypoint mobility model with uniform node spatial distribution. *Wireless networks* **20**(5), 1053–1066.
- Montgomery, Warren et al. (1983). Techniques for packet voice synchronization. *Selected Areas in Communications, IEEE Journal on* **1**(6), 1022–1028.
- Musable, Richard (2012). Hadilarijani and glasgow, evaluation of new scheduling scheme for voip with mobility in 3g lte. In: *The Fifth International Conference on Communication Theory, Reliability, and Quality of Service*.
- ns-3 project (2014). NS-3 Network Simulator Model Library Release ns-3-dev.
- ns-3 project (2015a). Friis Propagation Loss Model. Technical report. Available: <https://www.nsnam.org/doxygen>.
- ns-3 project (2015b). LTE Design. Technical report. Available: <https://www.nsnam.org/docs/models/html/lte-design.html>.
- ns-3 project (2015c). Steady State Random Waypoint Mobility Model. Technical report. Available: <https://www.nsnam.org/doxygen>.

- Orange, China Mobile, NTT KPN and Sprint DoCoMo (n.d.). T-mobile, vodafone, and telecom italia, “r1-070674: Lte physical layer framework for performance verification; 3gpp tsg ran wg1,”. Technical report. Tech. Rep.[Online]. Available: [http://www.3gpp.org/ftp/tsg\\_ran/WG1\\_RL1/TSGR1\\_48/Docs/R1-070674.zip](http://www.3gpp.org/ftp/tsg_ran/WG1_RL1/TSGR1_48/Docs/R1-070674.zip).
- Poikselkä, Miikka, Harri Holma, Jukka Hongisto, Juha Kallio and Antti Toskala (2012). *Voice over LTE (VoLTE)*. John Wiley & Sons.
- Ramjee, Ramachandran, Jim Kurose, Don Towsley and Henning Schulzrinne (1994). Adaptive playout mechanisms for packetized audio applications in wide-area networks. In: *INFOCOM'94. Networking for Global Communications., 13th Proceedings IEEE*. IEEE. pp. 680–688.
- RFC 4960 (2007). Stream Control Transmission Protocol (SCTP).
- Rumney, Moray et al. (2013). *LTE and the evolution to 4G wireless: Design and measurement challenges*. John Wiley & Sons.
- Sesia, Stefania, Issam Toufik and Matthew Baker (2009). *LTE: the UMTS long term evolution*. Wiley Online Library.
- UIT-T G.1050 (2007). Series G: Transmission Systems and Media, Digital Systems and Networks.
- UIT-T P.59 (1993). Calidad de transmisión telefónica. Aparatos para mediciones objetivas. Habla conversacional artificial.
- Veettil, Vinayak Thotton (2014). Voice over lte study and test strategy definition. PhD thesis. Arizona State University.
- Verger, Luciano M, Madhusudan Chaudhary, Onur E Tackin, Oren M Elrad, Srinivasan Vasudevan, Thanigaivelu Elangovan and Tiange Shao (2014). Enhancing jitter buffer performance through radio level feedback. US Patent App. 14/293,494.