UNIVERSIDAD DE MÁLAGA

Escuela Técnica Superior de Ingeniería de Telecomunicación

Programa de Doctorado en Ingeniería de Telecomunicación

Tesis Doctoral

OPTIMIZACIÓN DE LA CALIDAD DE EXPERIENCIA EN REDES CELULARES MEDIANTE EL AJUSTE DEL TRASPASO ENTRE CELDAS

Autora:

MARÍA LUISA MARÍ ALTOZANO

Directores:

SALVADOR LUNA RAMÍREZ

MATÍAS TORIL GENOVÉS

Málaga, 2021

AUTORIZACIÓN PARA LECTURA DE LA TESIS DOCTORAL

Por la presente, Dr. D. Matías Toril Genovés y Dr. D. Salvador Luna Ramírez, profesores doctores del Departamento de Ingeniería de Comunicaciones de la Universidad de Málaga, certifican que la doctoranda María Luisa Marí Altozano, Ingeniera de Telecomunicación, ha realizado en el Departamento de Ingeniería de Comunicaciones de la Universidad de Málaga, bajo su dirección, el trabajo de investigación correspondiente a su TESIS DOCTORAL titulada

Optimización de la calidad de experiencia en redes celulares mediante el ajuste del traspaso entre celdas

En dicho trabajo se han propuesto aportaciones originales dentro de la optimización de la calidad de experiencia en redes LTE mediante el ajuste parámetros de traspaso. Los resultados expuestos han dado lugar a las siguientes publicaciones en revistas y aportaciones a congresos que no han sido utilizadas en tesis anteriores.

1. M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, and C. Gijón, "A QoE-Driven Traffic Steering Algorithm for LTE Networks ", IEEE Transactions on Vehicular Technology, vol. 68, no. 11, pp. 11271-11282, Nov. 2019.
2. M. L. Marí-Altozano, M. Toril, S. Luna-Ramírez and C. Gijón, "A Self-Tuning Algorithm for Optimal QoE-Driven Traffic Steering in LTE," IEEE Access, vol. 8, pp. 156707-156717, Sep. 2020.
3. M. L. Marí-Altozano, S. Mwanje, S. Luna-Ramírez, M. Toril, H. Sanneck and C. Gijón, "A service-centric Q-learning algorithm for mobility robustness optimization in LTE ", aceptado en IEEE Transactions on Network and Service Management.
4. M.L. Marí-Altozano, S. Luna-Ramírez, M. Toril, "Limitaciones del equilibrio de carga para la mejora de la calidad de experiencia en redes LTE", XXXII Simposio de la Unión Científica Internacional de Radio (URSI) 2017, Cartagena(España), Sep. 2017.
5. M.L. Marí-Altozano, S. Luna-Ramírez, M. Toril, C. Gijón, "Algoritmo de control de carga basado en calidad de experiencia en redes LTE", XXXIII Simposio de la Unión Científica Internacional de Radio (URSI) 2018, Granada(España), Sep. 2018.
6. M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, "Load balance performance analysis with a quality of experience perspective in LTE networks", CA15104 IRACON, Graz(Austria), Sep. 2018.

7. M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, "A QoE-driven Traffic Steering algorithm for LTE networks", CA15104 IRACON, Dublin(Ireland), Jan. 2019.
8. M.L. Marí-Altozano, S. Luna-Ramírez, M. Toril, C. Gijón, "Optimización de la calidad de experiencia en redes LTE mediante el reparto de tráfico", XXXIV Simposio de la Unión Científica Internacional de Radio (URSI) 2019, Sevilla(España), Sep. 2019.
9. M.L. Marí-Altozano, S. Mwanje, S. Luna-Ramírez, M. Toril, C. Gijón, "Una visión basada en QoE para algoritmo MRO en redes LTE", XXXV Simposio de la Unión Científica Internacional de Radio (URSI) 2020, Málaga(España), Sep. 2020.

Por todo ello, consideran que esta Tesis es apta para su presentación al Tribunal que ha de juzgarla. Y para que conste a efectos de lo establecido, AUTORIZAN la presentación de esta Tesis en la Universidad de Málaga.

En Málaga, a 24 de abril de 2021

Fdo.: Dr. D. Matías Toril Genovés

Fdo.: Dr. D. Salvador Luna Ramírez

UNIVERSIDAD DE MÁLAGA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DE TELECOMUNICACIÓN

Reunido el tribunal examinador en el día de la fecha, constituido por:

Presidente Dr. D. _____

Vocal Dr. D. _____

Secretario Dr. D. _____

para juzgar la Tesis Doctoral titulada *"Optimización de la calidad de experiencia en redes celulares mediante el ajuste de parámetros de traspaso entre celdas"*, realizada por Dª. María Luisa Marí Altozano y dirigida por los Dres. D. Salvador Luna Ramírez y D. Matías Toril Genovés, acordó por _____ otorgar la calificación de _____
y, para que conste, se extiende firmada por los componentes del tribunal la presente diligencia.

Málaga, a _____ de_____ del _____

El presidente: El secretario: El vocal:

Fdo.: _____ Fdo.: _____ Fdo.: _____

A mis padres y mi hermano, gracias por vuestro apoyo incondicional.

A David, gracias por comprenderme siempre.

A Salvador y Matías, gracias por darme esta oportunidad.

A mis compañeros de laboratorio: Luis, Carolina, Juanlu, Pablo y Juanfran, gracias por esos momentos, tan necesarios, de desconexión.

DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD DE LA TESIS PRESENTADA PARA OBTENER EL TÍTULO DE DOCTOR

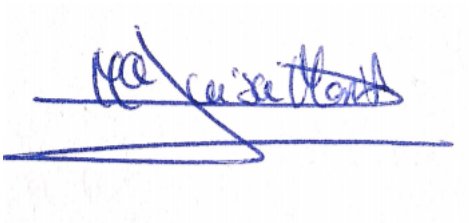
Dña María Luisa Marí Altozano, estudiante del programa de doctorado en Ingeniería de Telecomunicación de la Universidad de Málaga, autora de la tesis, presentada para la obtención del título de doctor por la Universidad de Málaga, titulada: *"Optimización de la calidad de experiencia en redes celulares mediante el ajuste del traspaso entre celdas"*, realizada bajo la tutorización de Salvador Luna Ramírez y dirección de Salvador Luna Ramírez y Matías Toril Genovés.

DECLARO QUE:

La tesis presentada es una obra original que no infringe los derechos de propiedad intelectual ni los derechos de propiedad industrial u otros, conforme al ordenamiento jurídico vigente (Real Decreto Legislativo 1/1996, de 12 de abril, por el que se aprueba el texto refundido de la Ley de Propiedad Intelectual, regularizando, aclarando y armonizando las disposiciones legales vigentes sobre la materia), modificado por la Ley 2/2019, de 1 de marzo.

Igualmente asumo, ante a la Universidad de Málaga y ante cualquier otra instancia, la responsabilidad que pudiera derivarse en caso de plagio de contenidos en la tesis presentada, conforme al ordenamiento jurídico vigente.

En Málaga, a 24 de abril de 2021



Fdo.: María Luisa Marí Altozano

Índice general

Índice de figuras

Índice de tablas

Resumen

En los últimos años, la gestión de las redes móviles se ha convertido en una tarea muy compleja por, entre otras razones, la diversidad de los servicios en movilidad. Para afrontar este problema, se ha extendido el uso de técnicas de automatización para la gestión de redes, dando lugar a las redes autoorganizadas (*Self Organizing Networks*, SON). En paralelo, las crecientes expectativas de los usuarios han provocado que los operadores cambien sus procesos de gestión, actualmente enfocados en el rendimiento de la red, a una aproximación más moderna centrada en la opinión del usuario. Ante este cambio de paradigma, los procedimientos clásicos de optimización de red deben actualizarse para considerar la calidad de experiencia (*Quality of Experience*, QoE) de forma explícita. Uno de estos procedimientos es el reparto de tráfico, cuyo objetivo es distribuir la demanda de recursos entre celdas adyacentes para aliviar los problemas de congestión causados por la distribución irregular de los usuarios. Otro de estos procedimientos clásicos es la optimización del traspaso por movilidad, que persigue mejorar la configuración de los parámetros de traspaso, evitando la caída de llamadas y la carga de señalización innecesaria.

Esta tesis aborda el problema de la regulación automática de parámetros de traspaso en redes *Long Term Evolution* (LTE) con criterios de QoE mediante tres contribuciones principales.

En primer lugar, se propone un algoritmo de reparto de tráfico basado en balance de carga para redes LTE con servicios de naturaleza muy diferente. A diferencia de trabajos previos, en los que el objetivo era equilibrar algún indicador de QoS, en este trabajo se persigue equilibrar la QoE de todas las celdas de la red. Para ello, utilizando lógica difusa, se ajustan los márgenes de traspaso entre celdas vecinas por adyacencia y por servicio en función de medidas de QoE recogidas por el sistema de gestión de la red. Los resultados muestran que el algoritmo de balance de QoE propuesto alivia los problemas de QoE equilibrando la QoE en todo el escenario.

En segundo lugar, se propone un algoritmo de reparto de tráfico entre celdas vecinas en una red LTE mediante el ajuste de los márgenes de traspaso con el objetivo de maximizar la QoE global del

sistema. Alternativamente al algoritmo anterior, que está basado en reglas heurísticas, éste utiliza un algoritmo de ascenso de gradiente para asegurar que cualquier modificación de parámetros siempre provoca una mejora en la QoE global del sistema. Para ello, se estima el impacto de un cambio en el margen de traspaso en la QoE del sistema con un modelo analítico del rendimiento de red que se puede ajustar con estadísticas de la red real. Los resultados indican que el rendimiento del método propuesto es mejor que el de los métodos clásicos de balance de carga y de QoE.

Por último, se propone un algoritmo de MRO basado en QoE en un escenario multiservicio. Mientras que el objetivo de los métodos MRO tradicionales es aumentar la tasa de traspasos realizados con éxito, el objetivo de optimización de este algoritmo es doble: mejorar la QoE de borde de celda y, simultáneamente, mejorar la tasa de traspasos exitosos en la red. Para conseguir dicho objetivo, se ajusta el punto de traspaso por adyacencia según medidas de QoE y medidas de fallo de traspaso. El punto de traspaso está definido por dos parámetros de control traspaso: el margen de traspaso y el temporizador de traspaso (*Time-To-Trigger*). Los resultados muestran que el algoritmo de MRO basado en QoE propuesto mejora la QoE de borde de celda en toda la red al mismo tiempo que mejora el porcentaje de traspasos realizados con éxito, y que su rendimiento es mejor que aquel de los algoritmos de MRO clásicos.

Los métodos propuestos se basan en contadores de rendimiento agregados de celda, y, por tanto, se conciben para ser implementados en el sistema de gestión de red. Para su validación, se han realizado pruebas en una red LTE piloto y en un simulador dinámico de nivel de sistema que modela un escenario macrocelular realista con servicios diferentes.

Abstract

In the last few years, mobile network management has become a really complex task due to the diversity of mobile services. To solve this problem, automatic network management techniques have been extensively used, resulting in the so-called Self-Organizing Networks (SON). In parallel, the raising user expectations have forced operators to shift their management processes, currently focused on network performance, to a more modern approach, focused on end user opinion. To this end, classical network optimization procedures must be updated to explicitly consider the Quality of Experience (QoE). One of these procedures is traffic steering, whose objective is to re-distribute radio resource demand among adjacent cells to alleviate congestion problems caused by uneven user distribution. A related problem is Mobility Robustness Optimization, seeking to improve handover parameters to avoid call drop and unnecessary signaling load.

In this Ph.D. thesis, the problem of QoE-driven automatic handover parameters tuning in Long Term Evolution (LTE) networks is addressed through three main contributions.

Firstly, a novel QoE-driven traffic sharing algorithm based on mobility load balancing is proposed for LTE networks offering services of very different nature. Unlike previous approaches, where the aim was to balance some QoS indicator, the aim here is to equalize the QoE provided by all cells in the network. For this purpose, using Fuzzy Logic, the handover margins between adjacent cells are tuned on a per-adjacency or per-service basis based on QoE measurements collected in the network management system. Results show that the proposed QoE-driven traffic sharing algorithm alleviates QoE problems by equalizing user QoE throughout the scenario.

Secondly, a novel self-tuning algorithm for parameters in a classical mobility load balancing scheme is proposed to steer traffic among adjacent cells in a Long-Term Evolution (LTE) network driven by QoE criteria. Alternatively to the previous algorithm, which is based on heuristic rules, this one takes a gradient ascent approach to ensure that parameter changes always improve the overall system QoE. For this purpose, the impact of parameter changes on system QoE is estimated

with an analytical network performance model that can be adjusted with statistics taken from the real network. Results show that the method outperforms classical load and QoE mobility load balancing schemes.

Finally, a novel QoE-aware MRO algorithm is proposed considering a multi-service scenario. Unlike classical approaches, whose aim is to increase successful handover rates, the optimization aim in this work is two-folded: to improve cell edge QoE while improving successful handover rates in the whole network. For this purpose, the handover trigger point, defined by the pair of HO control parameters HO margin and Time to Trigger, are tuned on a per-adjacency basis according to QoE and HO failure measurements. Results show that the proposed QoE-aware MRO algorithm improves cell edge QoE throughout the network while increasing the percentage of successful handovers compared to traditional approaches.

All the proposed methods are based on aggregated network performance counters and, therefore, are conceived to be implemented in the Operation Support Subsystem (OSS). Method assessment is carried out in a pilot LTE network and a system-level LTE system simulator implementing a realistic macrocellular scenario with different services.

Lista de acrónimos

3GPP	3rd Generation Partnership Project
AI	Artificial Intelligence
ANN	Artificial Neural Network
AWGN	Additive White Gaussian Noise
BC	Best Channel
BLER	Block Error Rate
CCO	Capacity and Coverage Optimization
CQI	Channel Quality Indicator
DNN	Deep Neural Network
DQN	Deep Q-Network
CQI	Channel Quality Indicator
DQN	Deep Q-Network
eMBB	Enhanced Mobile BroadBand
eNB	evolved Node B
EPC	Evolved Packet Core
ETU	Extended Typical Urban

FLC	Fuzzy Logic Controller
FTP	File Transfer Protocol
HARQ	Hybrid Automatic Repeat Request
HO	HandOver
HOM	HandOver Margin
HTTP	HyperText Transfer Protocol
IP	Internet Protocol
KPI	Key Performance Indicator
LTE	Long Term Evolution
MAC	Medium Access Control
MCS	Mobile and Coding Scheme
ML	Machine Learning
MLB	Mobility Load Balance
MLP	Multi-Layer Perceptron
mMTC	Massive Machine Type Communications
MOS	Mean Opinion Score
MRO	Mobility Robustness Optimization
OSS	Operations Support System
PBGT	Power BudGeT
PDSCH	Physical Downlink Shared Channel
PF	Proportional Fair
PIRE	Potencia Isotrópica Radiada Efectiva
PL	PathLoss
PP	Ping-Pong

PRB	Physical Resource Block
QCI	Quality Class Indicator
QL	Q-Learning
QoE	Quality of Experiencie
QoS	Quality of Service
RL	Reinforcement Learning
RLF	Radio Link Failure
ROP	Reporting Output Period
RR	Round Robin
RRM	Radio Resource Management
RSRP	Reference Signal Received Power
RSRQ	Reference Signal Received Quality
RTP	Real Time Protocol
SIR	Signal to Interference Ratio
SINR	Signal to Interference plus Noise Ratio
SNR	Signal to Noise Ratio
SON	Self Organizing Networks
SPI	Service Priority Index
TTI	Time Transmission Interval
TTT	Time To Trigger
URLLC	Ultra Reliable and Low Latency Communications
VoIP	Voice over IP

Capítulo 1

Introducción

El crecimiento exponencial de la demanda de tráfico asociada a los servicios en movilidad es un hecho de un tiempo a esta parte [1]. Además, la proliferación de teléfonos inteligentes y tabletas ha modificado el perfil de los servicios demandados en la red móvil, aumentando la dificultad en su gestión.

Todos estos cambios han provocado que se generalice el uso de técnicas de automatización para gestionar las redes de comunicaciones móviles, agrupadas bajo el término de redes autoorganizadas (*Self Organizing Networks*, SON) [2]. Tradicionalmente, la gestión automatizada de redes celulares se ha hecho en base a indicadores de calidad de servicio (*Quality of Service*, QoS). Sin embargo, las crecientes expectativas de los usuarios han condicionado la forma en que los operadores gestionan hoy sus redes. En un mercado en el que la oferta de servicios es similar en todos los operadores, la opinión del usuario se ha convertido en el principal factor que diferencia a unos operadores de otros. Con esta premisa, los operadores están cambiando sus procesos de gestión a una aproximación más moderna centrada en la calidad de experiencia (*Quality of Experience*, QoE). En este contexto, la QoE se define como la aceptabilidad global de un servicio, tal y como se percibe subjetivamente por el usuario final. Una medida de la QoE ampliamente utilizada es la nota media de opinión (*Mean Opinion Score*, MOS) [3].

Debido a las distintas expectativas de los usuarios de cada servicio, es habitual que servicios diferentes muestren comportamientos completamente distintos con relación a la calidad de experiencia aun cuando las condiciones de red puedan ser idénticas [3].

La perspectiva QoE podría aportar un criterio novedoso, no observado en la bibliografía hasta donde es conocido por los autores. Es posible que las técnicas habituales de gestión de las redes (que no tienen en cuenta dicha perspectiva) muestren limitaciones cuando son los estadísticos QoE, y no los de QoS, los que se usan como figura de mérito final para los algoritmos de optimización. Los operadores deben de hacer uso de toda la información disponible en sus redes con la finalidad de gestionar la QoE, ya que este tipo de información puede garantizar una operación de la red muy eficiente.

Ante el problema de las limitaciones de las técnicas clásicas de gestión de redes celulares en lo que se refiere a la QoE, cabe destacar que la tecnología 5G surge recientemente para hacer frente a los retos que presentan las redes de comunicaciones móviles. Para ello, utiliza diferentes tecnologías de acceso radio que se caracterizan por ser complejas, diversas y dinámicas.

La necesidad por parte de la industria de conseguir una gestión eficiente de las redes ha fomentado la investigación de nuevas técnicas automáticas de gestión de red.

En este contexto, se han propuesto numerosos métodos de optimización automática para mantener la configuración óptima de la red ante los cambios del entorno. Los casos de uso más relevantes son la optimización de la cobertura y la capacidad, la coordinación de interferencia entre celdas, el balance de carga y la optimización del traspaso por movilidad [4]. En esta tesis, el estudio se centra en los dos últimos casos de uso.

El balance de carga busca minimizar los efectos negativos de los cambios de la demanda de tráfico, que provocan que algunos elementos de red estén sobrecargados, mientras otros se encuentran infrautilizados. Aunque estos problemas aparecen en todos los segmentos de la red, es la red de acceso radio la que ha concentrado la mayor atención, por su complejidad.

Existen múltiples técnicas para equilibrar el tráfico en una red móvil, entre las que destaca, por su eficacia, la modificación del área de servicio de las celdas. En la literatura, se han propuesto diferentes maneras de cambiar el área de una celda. Un primer grupo de técnicas ajusta parámetros físicos de la estación base, como la potencia de transmisión de pilotos [5] o el ángulo de orientación de la antena [6] [7] [8]. En la práctica, estas técnicas rara vez se utilizan para el balance de tráfico, porque pueden crear huecos de cobertura. Como alternativa, un segundo grupo de técnicas modifica los parámetros de procesos de gestión de recursos radio, como la reselección de celda [9] o el traspaso [10]. Esta última suele ser la opción preferida, ya que el cambio de los parámetros de reselección de celda sólo tiene impacto en usuarios en estado desconectado y no contribuye a redistribuir la carga de celda ya que no hay conexiones activas. Por este motivo, la mayoría de los algoritmos de reparto de tráfico para redes celulares utilizan los márgenes de traspaso [11] [12].

Para encontrar el valor óptimo de los parámetros, el problema del ajuste de los márgenes puede formularse como un problema clásico de optimización [13]. Sin embargo, los operadores suelen resolver el problema con reglas heurísticas, ya que los datos y medidas necesarios para construir el modelo analítico del problema raramente están disponibles. Dependiendo de la rapidez del proceso de reasignación de tráfico, el indicador de rendimiento que se equilibra es la carga media de celda [11] [12], o la tasa de bloqueo de llamadas [13]. Como se muestra en [14], esta última opción se comporta mejor cuando los problemas de congestión son persistentes, por su mayor estabilidad. Además, no requiere inversión en equipamiento, ya que puede realizarse de forma centralizada desde el sistema de gestión de red.

Para la definición de las acciones de control, pueden utilizarse controladores de tipo proporcional (*Proportional Integrative Derivative*, PID) [15] o basados en lógica difusa [14]. Igualmente, los controladores pueden adaptarse de forma dinámica mediante técnicas de aprendizaje autónomo (p. ej., Q-learning [16]). La eficacia de estos métodos de balance de tráfico se ha demostrado en diferentes entornos (p. ej., macrocelulares [14], multiportadora [17], femtocelulares [18], red heterogénea [19]...) y tecnologías radio (p. ej., GSM [14], UMTS [20], LTE [15], LTE-A [17]). Sin embargo, diversas pruebas de campo han puesto de manifiesto algunas limitaciones originadas por el desplazamiento de usuarios hacia celdas distintas de la mejor servidora [21].

Por su parte, la optimización del traspaso por movilidad busca regular los parámetros de traspaso para evitar llamadas caídas y eliminar traspasos innecesarios. Tradicionalmente, este proceso se ha venido haciendo manualmente, tras una ardua labor previa de análisis, que no garantiza buenos resultados. Para resolver esta limitación, se introduce el caso de uso de la optimización del traspaso por movilidad (*Mobility Robustness Optimization*, MRO) en el conjunto de técnicas SON. Los experimentos demuestran que se puede mejorar dinámicamente el rendimiento del traspaso en la red detectando las pérdidas de conexión causadas por los traspasos realizados demasiado pronto o demasiado tarde [22].

En la literatura se pueden encontrar muchos trabajos que abordan el problema de la correcta configuración de los parámetros de traspaso. En [23] y [24] los autores proponen algoritmos de optimización conjunta del margen de traspaso (*HandOver Margin*, HOM) y el *Time-To-Trigger* (TTT). Sin embargo, estos algoritmos de optimización consideran un único valor de HOM y TTT por celda, haciendo imposible adaptarlos de manera individual por cada adyacencia concreta. El método adaptativo propuesto en [25] se modifica el parámetro *Cell Individual Offset* (CIO) por adyacencia en función de tipo de fallo de traspaso. Pero no ajusta otros parámetros, como el margen de traspaso y TTT, que tienen un fuerte impacto en el rendimiento del traspaso [26] [27].

Para mejorar el rendimiento del traspaso por movilidad se han usado distintas técnicas.Un

primer grupo de técnicas comprende la modificación de los parámetros de control del traspaso por movilidad, como el HOM, TTT [23] [24] o CIO [25]. Dentro del primer grupo de técnicas de optimización del traspaso por movilidad, y para conseguir una forma más flexible de modificar los parámetros de traspaso también se han propuesto algoritmos de aprendizaje autónomo [28] [29]. Un segundo grupo de técnicas usadas para optimizar del traspaso por movilidad son los algoritmos de decisión de traspaso [30] [31]. También se han utilizado los protocolos de enrutado móvil para reenrutar las llamadas de datos o de voz hacia un nuevo camino [32].

Al igual que los métodos de equilibrio de trafico, la eficacia de los métodos de optimización del traspaso por movilidad ha sido probada a lo largo de las diferentes tecnologías de acceso radio (GSM [33], UMTS [34], LTE [28], LTE-A [35] 5G [29]) y en distintos entornos de redes celulares (p. ej., macrocelulares [28] y redes heterogéneas [26], con una sola portadora [29], y en sistemas multiportadora [36] ...).

Tanto para el caso de uso de reparto de carga como para la optimización del traspaso por movilidad, no se han encontrado trabajos previos que propongan algoritmos de ajuste de parámetros de una red móvil considerando de forma explícita la QoE. Solo en [37] [38] se considera la modificación de parámetros de un planificador dinámico de recursos para repriorizar servicios con criterios de QoE. En esta tesis, el control y la optimización de la QoE comprenden las acciones para mejorar la eficiencia operacional de la red al mismo tiempo que se cumplen los objetivos de QoE. Considerando de forma explícita la QoE, se consigue mejorar la percepción del servicio por parte del usuario final aun cuando no se produzca una mejora significativa del rendimiento de la red. Los algoritmos propuestos basan su funcionamiento en la información disponible en el sistema de gestión de la red (*Operational Support System*, OSS). Dicha información proviene principalmente de contadores de eventos agregados por celda y hora, y de trazas de conexión, que registran los eventos de usuarios concretos de la red. Por ello, se han concebido para ser integrados en las herramientas de optimización de red que forman parte de la OSS de una red celular.

1.1. Objetivos del trabajo

El objetivo general de esta tesis es desarrollar algoritmos de optimización de la calidad de experiencia en redes LTE mediante la modificación de los parámetros de traspaso.

Para alcanzar ese objetivo general, es necesario:

1. Investigar métodos para estimar la QoE a nivel de usuario, celda y servicio a partir de la

información centralizada en el sistema de gestión de red.

2. Desarrollar algoritmos de reparto de tráfico que ajusten los márgenes de traspaso de forma heurística para equilibrar la QoE entre celdas vecinas de forma reactiva.

3. Desarrollar algoritmos de reparto de tráfico que modifiquen los márgenes de traspaso para maximizar la QoE global del sistema con criterios de optimalidad.

4. Desarrollar algoritmos de optimización del traspaso por movilidad que ajusten los márgenes y temporizadores del traspaso considerando de forma explícita la QoE de los usuarios de borde de celda.

Los algoritmos propuestos combinarán diversas técnicas del ámbito de la inteligencia artificial. Por un lado, los algoritmos de balance de tráfico implementarán las reglas de ajuste mediante controladores de lógica difusa. Por otro lado, los algoritmos de optimización del traspaso por movilidad emplearán técnicas de aprendizaje por refuerzo, implementadas con redes neuronales.

Los beneficios previstos de los métodos desarrollados en esta tesis son:

- Para los abonados, una mejor experiencia de uso, evitando los problemas derivados de la falta de recursos en servicios con requisitos muy diferentes.
- Para los operadores, una mejora del servicio prestado a los usuarios sin necesidad de desplegar nuevos recursos, fruto de la adaptabilidad de los métodos de optimización de parámetros para cada uno de los servicios.

1.2. Metodología de trabajo

Para cumplir los objetivos anteriormente marcados, se establece el siguiente plan de trabajo:

1. *Formulación del problema.* Estudio detallado del problema que se pretende abordar y revisión del estado de la técnica sobre QoE, SON y aprendizaje autónomo.

2. *Actualización de la herramienta de simulación.* El simulador dinámico de red LTE a nivel de sistema disponible debe modificarse para poder desarrollar y validar en él los algoritmos de balance y optimización de la QoE. Las modificaciones necesarias son las ya explicadas en el objetivo específico 1, es decir, aquellas necesarias para poder estimar la QoE por usuario,

celda y servicio a partir de la información disponible en el sistema centralizado de gestión de red.

3. *Desarrollo y validación del algoritmo de reparto de tráfico para equilibrar la QoE entre celdas vecinas mediante el ajuste de márgenes de traspaso.* Durante las pruebas, se comparará el algoritmo propuesto con las técnicas clásicas de balance de carga, basadas en indicadores de calidad de servicio.

4. *Desarrollo y validación del algoritmo de reparto de tráfico para optimizar la QoE global del sistema mediante el ajuste de márgenes de traspaso.* Utilizando el conocimiento adquirido durante el desarrollo del algoritmo de balance de QoE, se propone un nuevo algoritmo cuyo objetivo sea optimizar la QoE global de la red. Durante las pruebas, se comparará el algoritmo propuesto con las técnicas clásicas de balance de carga y con la técnica de balance de QoE.

5. *Desarrollo y validación del algoritmo de optimización del traspaso por movilidad basado en QoE mediante la modificación de márgenes y temporizadores de traspaso.* Durante las pruebas, se comparará el algoritmo propuesto con técnicas clásicas de optimización del traspaso por movilidad que también utilizan Q-learning.

1.3. Estructura de la memoria

Tras este primer capítulo, en esta memoria se distinguen dos partes claramente diferenciadas, dedicadas a las técnicas de reparto de tráfico y las estrategias de optimización del traspaso por movilidad. Ambos problemas se tratan, respectivamente, en los Capítulos 2 y 3, que comparten la misma estructura por claridad.

Previamente, en el Capítulo 2, se presentan los conceptos necesarios para contextualizar este trabajo. Se introduce la problemática de la demanda de tráfico en redes celulares y se presentan los modelos de tráfico utilizados en este trabajo. Al mismo tiempo, se repasan brevemente las técnicas de reparto de carga, inteligencia artificial, gestión de la QoE y monitorización del rendimiento en redes celulares. El Capítulo 3 se centra en las técnicas de reparto de tráfico en LTE. En primer lugar, se realiza un análisis preliminar del estado de las técnicas de reparto de tráfico en LTE, formulando analíticamente el problema a resolver. Seguidamente se presentan los dos algoritmos de reparto de tráfico propuestos en esta tesis. A continuación, se describen los experimentos realizados para validar los algoritmos y las conclusiones obtenidas de los resultados.

En el Capítulo 4 se presentan las técnicas de optimización del traspaso por movilidad. Como en el capítulo anterior, se comienza con un análisis detallado del estado de la técnica. En segundo lugar,

se formula el problema, identificando las variables de decisión y las principales cifras de mérito. En tercer lugar, se describe el algoritmo de optimización del traspaso por movilidad propuesto en esta tesis. A continuación, se muestran los experimentos realizados para validar la técnica de inteligencia artificial propuesta para optimizar la calidad de experiencia. Por último, se presentan las conclusiones de los resultados obtenidos.

En el Capítulo 5 se presentan las conclusiones generales y líneas futuras de extensión del trabajo realizado en esta tesis. Además, se presentan las publicaciones que avalan este trabajo y describen las aportaciones principales de esta tesis. Finalmente, en el Anexo A se describe la herramienta de simulación utilizada en esta tesis, actualizada para alcanzar los objetivos planteados. Para terminar, en el Anexo B se presenta un amplio resumen del trabajo en inglés.

Capítulo 2

Gestión de tráfico en redes móviles

En este capítulo, se presentan los conceptos necesarios para contextualizar esta tesis. En primer lugar, se introduce la problemática de la diversidad de servicios en redes celulares, presentando los modelos de tráfico utilizados en este trabajo. A continuación, se repasan brevemente las técnicas de reparto de carga que se usaban en las redes 2G y 3G, ampliadas posteriormente con las redes autoorganizadas en 4G. Seguidamente, se analizan las técnicas de inteligencia artificial más comunes en la gestión de redes celulares, por su especial relevancia en este trabajo. A continuación, se describen algunas técnicas para la gestión de la QoE en redes celulares, analizando sus principales características. Para finalizar el capítulo, se presentan las herramientas más utilizadas para la verificación del rendimiento de redes celulares, algunas de las cuales se emplean en este trabajo.

2.1. Demanda de tráfico en redes celulares

Como se explicó en la introducción, en las redes celulares se han sucedido muchos cambios en los últimos años: un crecimiento exponencial de la demanda de tráfico asociada a los servicios en movilidad, un cambio de los patrones de tráfico soportados y un aumento de las expectativas de los usuarios. Estos cambios han exigido la intervención de los operadores para asegurar un rendimiento eficiente de sus redes. Tradicionalmente, los operadores de redes móviles se han esforzado por aumentar la tasa de transmisión de datos de usuario. Con los nuevos dispositivos, se ha extendido el uso de aplicaciones que requieren conexión permanente (por ejemplo, redes sociales), lo que supone un aumento de la carga de señalización en la red. Lejos de reducirse, la tendencia de crecimiento

de la demanda de tráfico continuará en los próximos años con el despliegue de los sistemas 5G, que además, traerá consigo la introducción de nuevos casos de uso de servicios móviles [1]. Los nuevos sistemas 5G se conciben como un conjunto de técnicas que comprenden la densificación de red de acceso, la inclusión de nuevos tipos de nodos, la ubicación flexible del espectro o la compartición de infraestructuras [39]. Este nuevo entramado de tecnologías y servicios hace que la gestión de las redes sea mucho más compleja, lo que ya se identifica como uno de los principales problemas de las nuevas tecnologías radio [40]. Estos cambios requieren entender mejor el funcionamiento conjunto de terminales, aplicaciones y redes.

Para desarrollar mecanismos de gestión de red, es imprescindible conocer las características del tráfico generado por los servicios que se ofrecen en dichas redes. Cada servicio tiene características y requisitos muy diferentes. Algunos tienen altos requisitos de caudal, otros imponen severas restricciones de retardo, mientras que otros necesitan una fiabilidad muy alta. En este trabajo, en el que la principal herramienta de validación es un simulador de nivel de sistema, es indispensable disponer de modelos de tráfico precisos y realistas para que las soluciones alcanzadas por los algoritmos propuestos estén probadas en situaciones próximas a la realidad.

2.1.1. Modelos de tráfico

La Tabla 2.1 muestra las características principales de los cuatro servicios considerados en este trabajo: voz sobre el protocolo de internet (VoIP), transmisión progresiva de vídeo (VIDEO), servicio de descarga de ficheros mediante el protocolo de transferencia de ficheros (FTP) y servicio de descarga de páginas web (WEB) [41].

VoIP es un servicio de tasa de bit garantizada (*Guaranteed bit rate*, GBR) con bajo caudal de datos. El modelo del servicio VoIP está diseñado para generar 20 bytes de voz cada 10 ms, con una tasa de bit de 16 kbps. Por el contrario, los servicios VIDEO, FTP y WEB son servicios de tasa de bit no garantizada (*Non-Guaranteed Bit Rate*, non-GBR). El modelo del servicio de vídeo, inspirado en [42], se corresponde con la transmisión de vídeo en vivo de calidad fija (720 p) y tasa variable. Para modelar este servicio de vídeo, se implementa un modelo sencillo de *buffer* de reproducción en el lado del cliente. En la transmisión de vídeo en vivo, la generación del contenido y la petición de reproducción tienen lugar al mismo tiempo (a diferencia de lo que ocurre con el vídeo bajo demanda, en el que todo el contenido está disponible al inicio de la sesión). Por lo tanto, el servidor de vídeo comienza a enviar tramas de vídeo al cliente conforme se generan, y éstas se guardan en el búfer de cliente hasta que se alcanza un mínimo de contenido de vídeo (3 segundos, en este trabajo). Este comportamiento se modela como un retardo inicial de reproducción fijo (*initial*

Tabla 2.1: Parámetros de los modelos de tráfico.

Servicio	Características principales
VoIP	Tasa de codificación de voz 16 kbps
	Tiempo de sesión con distribución exponencial (duración media 60 s).
	Llamada caída tras 1 s sin recursos.
VIDEO	Codificador H.264/MPEG-4 AVC
	Vídeo de calidad fija con tasa variable (*Variable bit rate*, VBR)
	Resolución 720p a 25 tramas por segundo.
	Duración de la secuencia de vídeo con distribución uniforme entre 0 y 540 s.
	Tamaño de fotograma tomado de una traza real (medio 9.2 MB).
	Conexión caída cuando el tiempo de congelación de la imagen duplica al de la secuencia.
FTP	Tamaño de fichero con distribución lognormal (media 20 MB) [41].
WEB	Tamaño de página web con distribución lognormal (media 20 MB).
	No. páginas por sesión con distribución lognormal (media 4).
	Tiempo de lectura con distribución exponencial (media 107 s) [41].

buffering time, L_{ti}, de 3 segundos). Durante la reproducción, si el búfer del cliente se vacía, el vídeo se congela (evento denominado *stalling*) y el reproductor espera hasta que el búfer se llena de nuevo por completo. La duración de la secuencia de vídeo sigue una distribución uniforme entre 0 y 540 segundos. Obviamente, aquellos vídeos con duración inferior a 3 segundos no experimentan interrupción. El tamaño de cada fotograma de la secuencia de vídeo se toma de una traza de vídeo H.264 real [43]. Además, se simula un modelo de vídeo caído en el que la conexión finaliza si el tiempo de sesión supera el doble de la duración del contenido de la secuencia de vídeo. Los otros dos servicios de datos, FTP y WEB, son servicios de entrega de mejor esfuerzo *(best-effort)*. FTP es un servicio de descarga de ficheros y WEB consiste en la descarga de varias páginas web con distintos tamaños entre las que existe un tiempo aleatorio de lectura.

En todos los servicios, la llegada de llamadas se modela con una distribución de Poisson [44]. El valor de las tasas de llegada de cada servicio se determina según la mezcla de servicios observada en una red real.

2.2. Reparto de carga en redes 2G/3G

El problema de la gestión del tráfico en las redes actuales no es nuevo. En las antiguas redes 2G y 3G, el tráfico también se distribuía de forma irregular tanto en el espacio tiempo como en el tiempo, lo que complicaba mucho el dimensionado de las mismas. El resultado era una situación en

la que, mientras algunas celdas de la red estaban permanentemente congestionadas, otras estaban infrautilizadas. Por ello, la finalidad de las técnicas de reparto de carga en las redes 2G y 3G es aliviar los problemas de congestión causados por una demanda de tráfico irregular redistribuyendo a los usuarios entre celdas vecinas. Tal redistribución se consigue cambiando el área de servicio de las celdas mediante la modificación de parámetros de las estaciones base, como por ejemplo, la potencia de transmisión [5], el *offset* de reselección de celda [9], el ángulo de elevación de la antena [6] o los márgenes de traspaso [10]. Esta última es la opción preferida, denominada balance de carga por movilidad (*Mobility Load Balance*, MLB). Los algoritmos MLB se pueden clasificar en algoritmos estáticos o dinámicos [45].

Los algoritmos estáticos pueden hacer uso de enfoques analíticos para asegurar un rendimiento óptimo de forma proactiva a largo plazo [14].

Por el contrario, los algoritmos dinámicos confían en esquemas reactivos sencillos, pero tienen el inconveniente de que son propensos a inestabilidades. Los primeros algoritmos MLB diseñados para LTE (*Long Term Evolution*) estaban basados en controladores proporcionales sencillos controlados por el desequilibrio de carga entre celdas adyacentes [12] [11]. Como se muestra en [21], estos algoritmos pueden degradar de forma considerable el rendimiento de la red debido a la interferencia causada por la ajustada reutilización de frecuencias en LTE. Para solventarlo, los algoritmos más sofisticados usan controladores de lógica difusa adaptables con técnicas de aprendizaje autónomo [16], combinan MLB con el control remoto de la elevación del ángulo eléctrico [6] o replanifican la potencia de las estaciones base [18].

2.3. Redes autoorganizadas

Ante el cambio de paradigma en lo que se refiere a la cantidad y patrón de tráfico demandado en las redes celulares, se ha generalizado el uso de las herramientas SON. Esta generalización de herramientas SON ha permitido a los operadores celulares manejar el creciente tamaño y complejidad de sus redes. Los beneficios obtenidos se resumen a continuación [46]:

1. Reducción de costes de despliegue y costes operacionales, que, en otro caso, se verían incrementados por el elevado número de indicadores y parámetros que han de ser monitorizados y configurados.
2. Menores tiempos de despliegue en entornos de operación multitecnología y multifabricante.
3. Actualizaciones de red mucho más sencillas y rápidas.

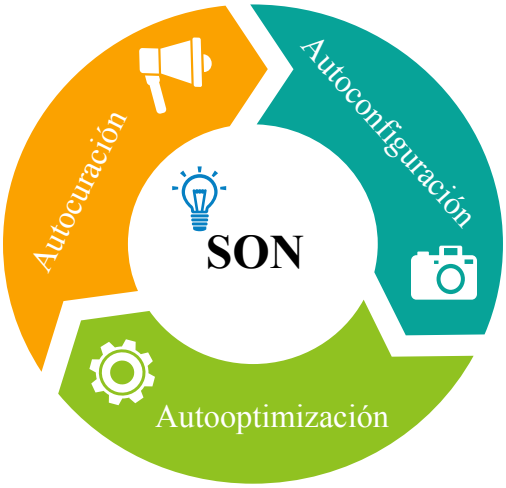


Figura 2.1: Casos de uso SON.

4. Liberación a los ingenieros radio de la realización de tareas manuales que han de repetirse en el tiempo y en el espacio.

Para su estudio, las técnicas SON suelen dividirse en tres grandes grupos de aplicaciones (Figura 2.1): la planificación automática (o autoconfiguración), la resolución automática de problemas (o autocuración) y la optimización automática de parámetros (o autoajuste). A continuación, se reflejan ejemplos de casos de uso de cada una de esas aplicaciones.

Autoconfiguración

El caso de uso de autoconfiguración es el proceso por el cual se configuran los nuevos elementos de red (como, por ejemplo, estaciones base) utilizando procedimientos de instalación automáticos (*plug&play*) que suministran la configuración básica necesaria para la operatividad del sistema. El proceso de autoconfiguración tiene lugar en el estado pre-operacional de la red, que comienza cuando se alimenta el nuevo elemento del sistema y se le provee de conectividad con la red troncal, finalizando cuando se enciende el transmisor de radiofrecuencia. Ejemplos de casos de uso son la selección de nuevos emplazamientos [47], la asignación de frecuencias [48], códigos de aleatorización [49] o identificadores de celda [50], la estructuración jerárquica de la red [51] [52] [53], la asignación de licencias [54] o la actualización de programas [4].

Autocuración

El caso de uso de autocuración es una parte importante de la gestión automática de las redes celulares, ya que, aunque la red no es capaz de recuperar una celda dañada físicamente, se puede identificar qué elemento de red no funciona correctamente. Algunos de los procesos que la autocuración puede incluir son la correlación y priorización de alarmas [55] [56], la detección de problemas (p.ej., celda dormida) [57], el diagnóstico de causas [58] o la compensación de problemas [59].

Autooptimización

El objetivo de la autooptimización es encontrar la configuración de los parámetros de red más eficaz en un determinado momento. Como las condiciones de red cambian con el tiempo, el sistema ha de reconfigurarse repetidamente. Como se muestra en la Figura 2.2, la optimización de redes celulares es un proceso continuo en lazo cerrado que incluye la evaluación periódica del rendimiento, la optimización de los parámetros de la red y la descarga de esos parámetros optimizados en la red. Las decisiones de optimización pueden tomarse por personas expertas o por sistemas automáticos. El proceso de optimización necesita tener datos de entrada, que pueden recogerse de diferentes fuentes de información. Entre estas fuentes de información, se pueden mencionar las medidas de rendimiento, las alarmas o las trazas de conexión, todas ellas disponibles en el sistema de gestión de red (OSS). Como complemento, pueden utilizarse pruebas de campo con terminales de medida específicos que recogen de rendimiento geolocalizada. Estos datos de entrada, junto con la configuración actual de la red, permiten a los algoritmos de optimización calcular los valores más apropiados para los parámetros de la red, que serán implementados para cambiar el comportamiento de la red y mejorar su rendimiento. Este lazo de optimización tiene como objetivo ajustar los parámetros de red para conseguir una cierta cifra de mérito, expresada en términos de cobertura, calidad de señal o capacidad o una combinación de todas ellas, teniendo en cuenta que cualquier cambio de valores de los parámetros implica un compromiso implícito entre dichas variables.

Los algoritmos de optimización son propietarios de cada fabricante, aunque el operador puede regular su comportamiento a través de parámetros internos. No obstante, las medidas e interfaces necesarios deben estandarizarse para permitir la interacción entre algoritmos en escenarios donde coexisten varios proveedores de equipamiento. Para facilitar la integración en estos entornos multiproveedor, los operadores suelen desarrollar capas de adaptación para traducir y armonizar las medidas de rendimiento de cada proveedor que permitan utilizar una plataforma de optimización común [4].

Los casos de uso de autooptimización son muy numerosos. Entre ellos destacan la optimización

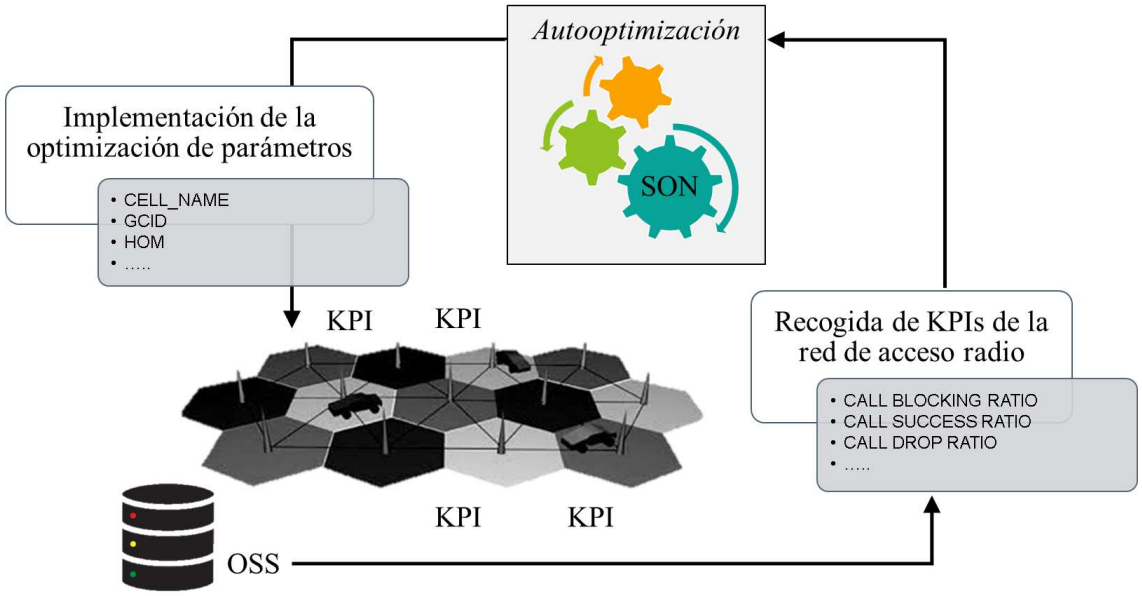


Figura 2.2: Autooptimización en lazo cerrado de la red de acceso radio.

del compromiso entre cobertura y capacidad (*Capacity and Coverage Optimization*, CCO) [60], la optimización de la adaptación del enlace [61], la optimización del planificador de recursos [38], el balance de carga por movilidad (MLB) [15], el alivio de congestión [62] y la optimización del traspaso por movilidad (MRO) [28].

2.4. Inteligencia artificial en la gestión de redes celulares

Aunque se han propuesto muchas técnicas para mejorar el rendimiento de las redes celulares, la aplicación de métodos avanzados para manejar la complejidad de estas redes se ha convertido en un tema candente de investigación. Como ya se ha explicado, las redes celulares son extraordinariamente grandes, dinámicas, heterogéneas, y complejas. Desde hace tiempo, la comunidad investigadora ha dedicado grandes esfuerzos en desplegar nuevos métodos de Inteligencia Artificial (*Artificial Intelligence*, AI) para la gestión automática y la optimización de las redes celulares. Los métodos de AI incluyen técnicas multidisciplinares de aprendizaje automático (*Machine Learning*, ML), algoritmos bioinspirados, redes neuronales difusas, etcétera. Estas técnicas vienen aplicándose con éxito para optimizar sistemas y redes informáticas en escenarios diversos [63] [2] [64]. Estas técnicas suelen aprovechar el conocimiento previo del problema y el aprendizaje recursivo basado en la retroalimentación y en las interacciones locales para mejorar la eficacia de las técnicas conven-

cionales a la hora de encontrar soluciones de gran rendimiento [63] [2]. Por ello, la integración de las técnicas de AI en el diseño de redes inteligentes se ha convertido en una tendencia prometedora para la resolución eficiente de problemas en las redes celulares (por ejemplo, planificación radio, gestión dinámica de recursos radio, regulación de parámetros, etc.).

2.4.1. Clasificación de las técnicas de inteligencia artificial

En este apartado, se clasifican las principales técnicas de inteligencia artificial para la gestión de las redes celulares, indicando algunos trabajos en los que se aplican.

Sistemas de lógica difusa

Se trata de sistemas de control basados en reglas donde el estado actual del sistema se define mediante conjuntos difusos. Estos sistemas constan de tres fases: la fase de fuzzificación, la de inferencia y la de defuzzificación. En la fase de fuzzificación, se analizan matemáticamente entradas numéricas en términos de variables lingüísticas mediante las funciones de pertenencia de entrada. Seguidamente, en la fase de inferencia, se define un conjunto de reglas que relacionan la entrada con la salida en términos lingüísticos. Este tipo de lógica permite que se puedan cumplir varias reglas al mismo tiempo con distintos grados de cumplimiento. Por último, en la fase de defuzzificación, se obtiene el valor de la salida como resultado de agregar el valor de salida de todas las reglas [65]. De esta forma, la lógica difusa es capaz de definir reglas deduciendo conocimiento a partir de información imprecisa, incierta o poco fiable. En el contexto de un sistema de control, la lógica difusa puede ayudar de decidir qué acción tomar basándose en un conjunto de reglas a partir de la información de recursos físicos y de distintos sistemas con condiciones y características variables como son las redes celulares. La principal ventaja que ofrecen este tipo de controladores es que se expresan en términos lingüísticos, lo que permite convertir de forma sencilla la experiencia de los operadores en reglas condicionales del tipo «SI (condición) –ENTONCES (acción)» [66] [14] [67].

La lógica difusa se ha venido utilizando en la optimización de redes celulares desde hace mucho tiempo. Como ejemplo, en [14] se propone un algoritmo de reparto de tráfico basado en lógica difusa que optimiza conjuntamente los márgenes de traspaso y restricciones de nivel de señal basándose en estadísticas de rendimiento de una red de acceso radio 2G.

Igualmente, se han empleado técnicas de lógica difusa para la optimización de redes LTE en el contexto de las redes autoorganizadas. En [18] se combina el reparto de tráfico con la reconfiguración de potencia de celda mediante controladores de lógica difusa. De igual forma, en [68] se propone

un algoritmo basado en lógica difusa para optimizar los parámetros de traspaso con el objetivo de maximizar la tasa de traspasos realizados con éxito.

Redes neuronales artificiales

Las redes neuronales artificiales (ANNs) son un tipo de modelo de aprendizaje estadístico inspirado en las redes biológicas de neuronas de los seres humanos [69]. Las ANNs se presentan generalmente como sistemas de neuronas interconectadas mediante pesos numéricos, que pueden ajustarse para hacer redes adaptativas capaces de aprender [70]. Esta adaptación asegura su capacidad de aprender en entornos no supervisados, especialmente valiosa para el contexto de las redes autoorganizas, donde es preciso estimar funciones que dependen de varias condiciones de entrada desconocidas [71].

Las redes neuronales artificiales son especialmente útiles en el caso de uso de la autooptimización por su rápida convergencia, lo que les permite satisfacer las necesidades de muchas aplicaciones prácticas en entornos dinámicos y cambiantes. Por ejemplo, en [72], la regla convencional de histéresis en el traspaso se sustituye por una red neuronal artificial que realiza reconocimiento de patrones de potencia de señal recibida y decide si efectuar el traspaso o no con la esperanza de reducir la probabilidad de fallo del mismo. En [73], se usa una ANN para construir un modelo de rendimiento de red relacionando el tráfico cursado, la potencia de señal, la calidad de señal y las tasas de llamadas caídas/bloqueadas, que permite a los operadores dirigir sus acciones. De forma paralela, el estudio presentado en [74] propone la optimización del compromiso entre cobertura y capacidad regulando la inclinación de la antena y su potencia mediante un método basado en el aprendizaje automático con una red neuronal difusa. Las ANNs también se pueden usar para abordar el problema del traspaso vertical (cambio de red de acceso radio) para mejorar la QoS de usuario y el rendimiento en redes heterogéneas. Así, en [75], se propone un algoritmo de ajuste adaptativo basado en un modelo de red neuronal que determina el valor óptimo de distintos parámetros definidos a nivel de usuario.

Dependiendo del número de capas de neuronas de las ANN, las redes neuronales artificiales se clasifican en poco profundas o profundas. Las redes neuronales artificiales profundas (*Deep Neural Networks*, DNN) tienen la ventaja de que ofrecen mejores aproximaciones de la función en cuestión, pero también el inconveniente de que tienen altos requisitos en lo que se refiere a coste computacional, tamaño de los juegos de datos para entrenarlas adecuadamente y, además, pueden presentar problemas de sobreajuste [76].

Aprendizaje automático

Las técnicas de aprendizaje automático son técnicas cuyo objetivo es que los ordenadores aprendan sin ser programados específicamente para ello. Es habitual emplear dichas técnicas en la gestión automática de las redes móviles [77]. Con estas técnicas, los algoritmos tradicionales de ajuste reactivo, basados en la comparación de umbrales mediante reglas heurísticas sencillas, se pueden sustituir por algoritmos inteligentes capaces de encontrar automáticamente la mejor política de ajuste y cambiar sus parámetros de forma proactiva para adaptarse rápidamente a entornos cambiantes [78].

Entre las numerosas técnicas de ML, un caso particular es el aprendizaje por refuerzo (*Reinforcement Learning*, RL) [79]. El aprendizaje por refuerzo, inspirado en la psicología conductual, se centra en la forma en que agentes informáticos han de tomar acciones en un entorno para maximizar las recompensas acumuladas. RL tiene como objetivo encontrar una política que maximice las recompensas observadas a lo largo del tiempo. La técnica de RL más utilizada es *Q-Learning* (QL), ya que no requiere un modelo del entorno. Su objetivo es encontrar un valor Q (entendido como un valor de calidad) óptimo para cada par estado-acción.

El aprendizaje por refuerzo se ha usado ampliamente en la optimización de redes móviles. En [80], se propone un método basado en QL para modificar adaptativamente la potencia de transmisión de nuevas femtoceldas en una red heterogénea para mejorar el rendimiento de los usuarios y disminuir el consumo. En [28], se propone un algoritmo QL para optimizar el rendimiento del traspaso reduciendo los fallos por realizarlo demasiado pronto/tarde, así como la tasa de traspasos innecesarios (ping-pong) que aumentan la carga de señalización de la red. Del mismo modo, en [81], se emplea QL para modificar adaptativamente los parámetros de traspaso con el objetivo de reducir los traspasos innecesarios y la tasa de llamadas caídas.

También existen enfoques más sofisticados que combinan RL y DNN (técnica también conocida como aprendizaje por refuerzo profundo). Esta estrategia se ha utilizado para construir algoritmos avanzados de balance de carga en redes celulares capaces de encontrar la política óptima en sistemas complejos aprovechando la capacidad de generalización de las redes neuronales artificiales [82] [83]. Del mismo modo, en [16] se presenta un algoritmo de QL para adaptar un controlador de lógica difusa diseñado para ajustar los parámetros de traspaso con el objetivo de equilibrar la carga entre celdas, seleccionando adecuadamente los consecuentes de las reglas en la máquina de inferencia.

2.5. Gestión de la experiencia de usuario

Las principales tareas en la gestión de la QoE son: a) la caracterización y el modelado de la QoE, b) la medida y monitorización de la QoE, c) la planificación de la QoE, y d) la optimización y el control de la QoE.

Caracterización y modelado de la QoE

En la caracterización de la calidad experiencia, el primer paso es identificar los factores que influyen en la percepción del servicio. El proyecto europeo Qualinet, que hace una clasificación exhaustiva de estos factores, los divide en tres grupos: humanos (p. ej., edad, género o educación), de sistema (p. ej., medidas de red, dispositivo o aplicación), y de contexto (p. ej., hora del día, ubicación o movilidad del usuario [84]).

Tradicionalmente, la QoE se ha evaluado mediante encuestas de satisfacción llevadas a cabo entre los usuarios para conocer su grado de satisfacción utilizando la escala MOS (*Mean Opinion Score*). Sin embargo, este método es muy costoso en tiempo y precio, y además no es agradable para el usuario. Además, este método no se puede usar para tomar decisiones que puedan mejorar la QoE instantáneamente. Por ello, recientemente se han propuesto nuevos métodos que permiten estimar la QoE dependiendo de determinados indicadores de rendimiento asociados a los servicios. Una posible solución es integrar analizadores de QoE dentro de los terminales móviles que sean capaces de ir informando de medidas subjetivas de QoE a un servidor central [85]. Esto simplifica significativamente el proceso de evaluación de QoE. Otras soluciones se centran en incluir nuevos elementos de red (por ejemplo, analizadores de red, inspectores profundos de paquetes, etc.) que pueden capturar el tráfico de un servicio concreto y analizar su rendimiento [86].

No obstante, cualquiera de estas soluciones que persigue estimar la QoE a partir de medidas de tráfico implican algún tipo de traducción a un valor de QoE. Una posible solución para llevar a cabo este proceso es aplicar una función de utilidad asociada a cada servicio en particular para traducir el nivel objetivo de QoS a un nivel de subjetivo de QoE (en escala MOS). Muchos trabajos de investigación se centran en esta solución, y este trabajo también lo hace.

Una vez identificados los indicadores de QoS que influyen en la percepción del servicio, el siguiente paso es diseñar un modelo matemático sencillo que relacione dichos parámetros de QoS con la QoE final. Si se desea una caracterización más precisa, pueden utilizarse métodos de evaluación subjetiva u objetiva de la QoE. Los primeros se basan en la realización de experimentos con usuarios reales, analizando el impacto sobre la MOS de los distintos parámetros de QoS. Por el contrario, los

métodos objetivos sólo necesitan las medidas extraídas de la red, lo que hace posible su automatización. La utilidad de cada método depende de la situación. En el contexto de las comunicaciones móviles, en [84] se clasifican los modelos de QoE en función de los factores considerados (sistema, contexto o humanos), el tipo de servicio (multimedia, vídeo, datos,...) y el ámbito de aplicación (p.ej., monitorización del servicio, optimización del servicio o planificación y optimización de red). Para servicios móviles, los modelos de QoE deben considerar: a) las fluctuaciones de la QoS como consecuencia de la variabilidad del canal radio, b) la movilidad del usuario y el cambio de tecnología, c) la señalización, d) las especificaciones del terminal, y e) la variabilidad del contexto [87]. Los modelos clásicos de QoE en entornos móviles (p. ej., voz [88], *videostreaming* [89] o web [90]) sólo consideran algunos de estos aspectos, pues no es posible recabar el resto de información. Además, los continuos avances en este tipo de servicios exigen revisar los modelos tradicionales de QoE.

Medida y monitorización de la QoE

La medida y monitorización de la QoE trata de encontrar métodos de cuantificación de la QoE de usuario a partir de esos indicadores globales de rendimiento de la red utilizando los modelos de QoE presentados en el apartado anterior. Estos métodos difieren en el lugar y el mecanismo de recogida de datos [91]. Según el origen de recogida de los indicadores de rendimiento, estos métodos de medida y monitorización de la QoE se clasifican en medidas en la red y medidas en el cliente. Las medidas en la red se basan en la recogida de indicadores de rendimiento de la red, a partir de los que es difícil deducir la experiencia del usuario final. Como alternativa, las medidas en el lado del cliente aportan información muy precisa desde la perspectiva del usuario. Por ello, el 3GGP ha estandarizado mecanismos de reporte periódico de la QoE para los usuarios de servicios de *streaming* RTP (*Real Time Protocol*) o HTTP (*HyperText Transfer Protocol*), descarga progresiva y telefonía multimedia [92] [93]. Sin embargo, este tipo de medidas plantean problemas de privacidad y confianza, dependiendo en último término de los proveedores del servicio y los desarrolladores de aplicaciones. Además, aun disponiendo de información precisa de la QoE de usuario, no aportan información sobre las causas de los deterioros de la QoE. Según el mecanismo de recogida de datos, los métodos de medida y monitorización de la calidad de experiencia se clasifican en métodos de monitorización activa y pasiva. Los métodos activos introducen paquetes artificiales de prueba mediante aplicaciones específicas llamadas *user terminal agents*, que permiten medir la QoE en diferentes segmentos de la red. Estos métodos intrusivos provocan un aumento del tráfico que impide utilizarlos durante las horas de mayor demanda, justamente el periodo en el que más interesa la monitorización. Por su parte, los métodos pasivos extraen información del rendimiento de los servicios a partir de contadores de rendimiento del sistema o sondas en las interfaces entre equipos. De la información que aportan los contadores del sistema, sólo pueden

extraerse tendencias generales del comportamiento de los servicios, pues no es posible diferenciar entre usuarios del mismo servicio, ni entre servicios que comparten el mismo QCI (*Quality Class Indicator*). Por el contrario, las sondas sí permiten segregar cada conexión mediante técnicas de inspección profunda de paquetes (*Deep Packet Inspection*, DPI) [86].

Planificación de la QoE

La gestión tradicional de la capacidad se basa en encontrar límites que especifiquen el máximo nivel de utilización que se puede tolerar asegurando la calidad de experiencia mínima requerida. Por ejemplo, en redes de voz clásicas, dichos límites se calculan mediante el uso de la fórmula de Erlang B/C dependiendo de la tasa de bloqueo de llamadas máxima tolerable. No obstante, en las redes móviles de banda ancha, no existe esta relación tan sencilla entre calidad de servicio y capacidad de red. Aunque existen modelos de planificación que aseguran la igualdad en los recursos asignados a los usuarios y métodos que permiten aproximar los valores máximos permitidos de los principales parámetros de QoE para los servicios más extendidos (p. ej., voz [88] o vídeo [94]), ninguno sirve para estimar la capacidad de cada uno de los segmentos de red (y, en particular, de la red radio) en un escenario multiservicio. La monitorización de la utilización de recursos ha de ser complementada con la monitorización de la calidad de experiencia extremo a extremo, dando así una visión completa de la QoE en referencia a la topología de la red y el tiempo. Todo ello permite construir una correlación fiable entre el grado de utilización y la calidad de servicio, que sirve como base para una planificación económica de la capacidad. Ésto es especialmente importante para servicios en tiempo real, como VoIP o transmisión de vídeo. Existen herramientas pensadas para ser implementadas en la red que permiten identificar el nivel mínimo de recursos necesario en la interfaz radio para conseguir la QoE deseada y minimizar los gastos asociados a la planificación de la red [3]. Estas herramientas en el dispositivo móvil presentan importantes inconvenientes, como la escalabilidad y la dependencia del fabricante. Por ello, los métodos del lado del cliente deben considerarse solo como un complemento de los métodos basados en las medidas de red.

Optimización y control

La optimización y control de la QoE en redes celulares comprende las acciones para mejorar la eficiencia operacional de la red al mismo tiempo que se cumplen los objetivos de QoE. La mayoría de trabajos publicados en este tema proponen algoritmos de planificación dinámica de recursos (*scheduling*) que utilizan de forma explícita criterios o indicadores de QoE para priorizar ciertas conexiones con el objetivo de equilibrar [37] o maximizar [38] algún indicador agregado de QoE. Estos estudios incluyen servicios como la voz sobre IP, la transmisión de vídeo, la transferencia de

ficheros, y la navegación web, en tecnología LTE. Cuando se inició este trabajo, no se encontraron algoritmos de ajuste de parámetros de una red móvil que tuvieran en cuenta de forma explícita la QoE para LTE, salvo los mencionados anteriormente de repriorización de servicios en el planificador [37] [38]. Aprovechando dicha escasez de contribuciones, este trabajo se encuadra en este apartado de optimización de la QoE en redes celulares.

2.5.1. Modelos de QoE

En este trabajo, la QoE se mide usando la escala MOS, que va desde el valor 1 (mala experiencia) hasta el 5 (experiencia excelente). Si no se dispone de encuestas de satisfacción, la QoE puede estimarse de forma objetiva partiendo de medidas básicas de QoS de nivel de red (p. ej., caudal de datos, tasa de pérdidas de paquetes, retardo medio de paquete...) o indicadores de rendimiento del servicio (*Service KPIs*, S-KPI, como p.ej., el tiempo medio de descarga de archivo o el tiempo inicial de retardo en la reproducción de un vídeo). Para ello, las medidas de QoS o S-KPI se recogen por sesión y se traducen a cifras de QoE mediante funciones de utilidad [95]. Una función de utilidad describe la relación entre indicadores objetivos de calidad/rendimiento de servicio y valores subjetivos de calidad de experiencia para cada servicio. Dichas funciones de utilidad proporcionan una estimación de la QoE de usuario, aunque no tienen en cuenta factores de contexto, como pueden ser la ubicación geográfica o la hora del día. Por tanto, los operadores de red que no tienen en cuenta medidas explícitas de QoE pueden estimar la QoE de usuario procesando medidas pasivas de QoS o S-KPI. En este trabajo, se usan los modelos de QoE descritos en los siguientes párrafos.

Para el servicio de VoIP, la QoE de usuario se estima como [96]:

$$QoE^{(VoIP)} = 1 + 0.035R + R(R-60)(100-R)7 \cdot 10^{-6}\,, \tag{2.1}$$

donde $QoE^{(VoIP)}$ es el valor en la escala MOS para una conexión de VoIP, y R es un parámetro que representa la calidad de la conexión, con valores entre 0 (mínimo) y 93 (máximo), que solamente dependen del retardo experimentado por los paquetes de VoIP (retardo boca-oreja). En concreto, R se calcula como el percentil del 90 % de los retardos sufridos por los paquetes de voz en el enlace descendente en milisegundos ($R = 0$ para los retardos más altos, y $R = 100$ para los más bajos). Se ha de tener en cuenta que $\max(QoE^{(VoIP)}) = 4.054$ (cuando $R = 93$); así, el valor de MOS nunca alcanza el valor 5, haciendo patente que, incluso con el mejor rendimiento de red posible, algunos individuos pueden no calificar su experiencia como excelente. De la misma forma, la QoE toma su valor mínimo (es decir, $QoE^{(VoIP)} = 1$) cuando la conexión se cae. La Figura 2.3 muestra

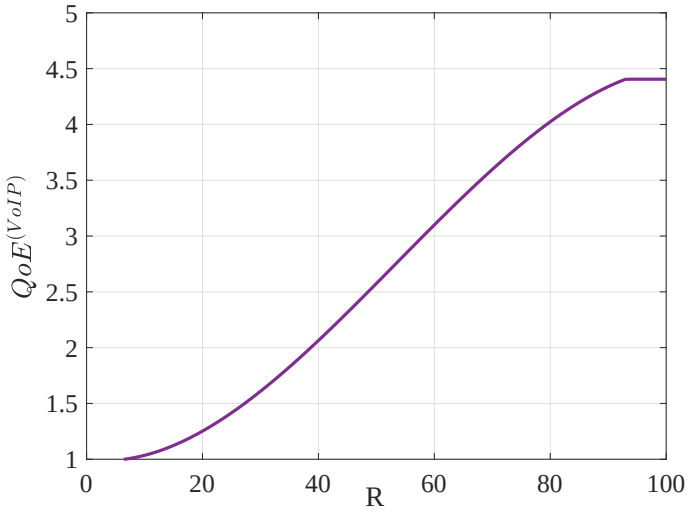


Figura 2.3: Influencia del parámetro R en la calidad de experiencia del servicio VoIP.

la dependencia de la QoE del servicio VoIP con el parámetro R.

La función de utilidad del servicio VIDEO se define como [37]:

$$QoE^{(VIDEO)} = 4.23 - 0.0672L_{ti} - 0.742L_{fr} - 0.106L_{tr} \ , \tag{2.2}$$

donde $QoE^{(VIDEO)}$ es el valor de MOS estimado para la conexión de vídeo, L_{ti} denota el tiempo inicial de llenado del búfer (en segundos), L_{fr} es la frecuencia media de *stalling* (s^{-1}) (es decir, número de veces por segundo que, a lo largo de la conexión, el reproductor de vídeo queda en pausa debido a que el búfer del cliente está vacío), y L_{tr} es la duración media de *stalling* (en segundos) en el enlace descendente. Por simplicidad, en este trabajo se asume que el terminal reporta los valores de L_{ti}, L_{fr} y L_{tr}. En su defecto, la red debería estimar esos indicadores del servicio a partir de medidas del caudal de datos del enlace descendente. De la fórmula, se deduce que el valor máximo de QoE para una conexión de vídeo está limitada superiormente a 4.23. Como ocurre en el servicio VoIP, $QoE^{(VIDEO)} = 1$ si la conexión se cae.

Para el servicio FTP, la función de utilidad es [97]

$$QoE^{(FTP)} = \text{máx}(1, \text{mín}(5, 6.5 \cdot T - 0.54)) \ , \tag{2.3}$$

donde T denota el *throughput* medio de usuario en el enlace descendente en Mbps.

Finalmente, la función de utilidad para el servicio WEB es [97]

$$QoE^{(WEB)} = 5 - \frac{578}{1 + (\frac{T+541.1}{45.98})^2} \, , \tag{2.4}$$

donde T es el throughput medio de usuario en el enlace descendente en kbps. Es importante resaltar que $\max(QoE^{(WEB)}) = 5$. Para los usuarios web no se ha considerado ningún modelo de conexión caída, por lo que dichas conexiones alcanzan valores bajos de MOS cuando T es próximo a cero (es decir, $QoE^{(WEB)} = 1$ cuando $T \simeq 0$ Mbps).

Cabe mencionar que los modelos de QoE descritos no dependen de los parámetros de los modelos de tráfico (por ejemplo, la duración de la secuencia de vídeo, el tamaño del fichero FTP o el número de páginas web).

La satisfacción de los usuarios depende del servicio concreto, de forma que la QoE para usuarios de distintos servicios puede ser muy diferente incluso recibiendo la misma cantidad de recursos radio. Como ejemplo, la Figura 2.4 muestra los distintos valores de QoE estimados para el mismo caudal de datos de usuario con la función de utilidad correspondiente para los servicios FTP y WEB. Se aprecia cómo para valores de T bajos, el servicio WEB experimenta valores de QoE más altos que los del servicio FTP, mientras que, para valores altos de T, el servicio FTP alcanza valores de QoE más altos que los del servicio WEB. Este comportamiento distinto de la QoE con el caudal de datos de usuario para FTP y WEB hace patente la diversidad de los dos servicios, poniendo de manifiesto que, aun siendo dos servicios de datos de entrega de mejor esfuerzo, para experimentar la misma QoE, tienen distintos requisitos de T.

2.6. Herramientas para la verificación del rendimiento de redes celulares

La gran complejidad de las redes celulares actuales implica la necesidad de disponer de herramientas que permitan evaluar de forma fiable todos los mecanismos involucrados en la gestión de red. Estas herramientas permiten, entre otras cosas, verificar con antelación el funcionamiento correcto de los algoritmos de optimización por medio de modelos de rendimiento. De esta forma, se evitan efectos indeseados que degraden el rendimiento de las redes en funcionamiento cuando

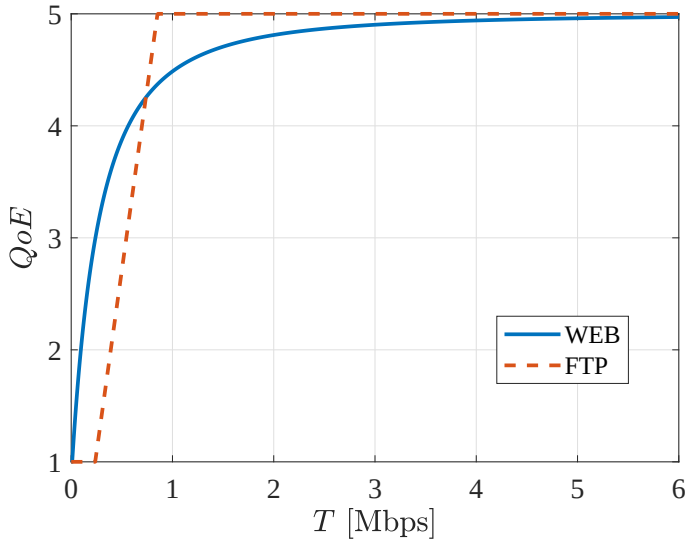


Figura 2.4: Influencia del caudal de datos en la calidad de experiencia de diferentes servicios.

los algoritmos se integren en la red. En la fase de planificación de red, suelen utilizarse primero modelos matemáticos sencillos que permitan evaluar el desempeño de la red desde un punto de vista analítico. La literatura sobre modelos matemáticos como herramienta de validación en redes de comunicaciones móviles es muy amplia. Por su cercanía con el trabajo desarrollado en esta tesis, se puede citar [98], donde se usa un modelo analítico basado en teoría de colas para diseñar algoritmos óptimos de asignación de recursos para aplicaciones móviles de tiempo real. Del mismo modo, en [99], se usa teoría de colas en la planificación de la capacidad y la evaluación del rendimiento de un sistema de comunicaciones móviles. También en [100] se propone un modelo matemático para encontrar la configuración óptima de los parámetros de traspaso en redes LTE con el objetivo de mejorar la eficiencia y fiabilidad de dichos traspasos.

Cuando no se dispone de un modelo analítico del sistema, se emplean simuladores de nivel de red. En esta categoría, se distinguen dos tipos de simuladores: estáticos y dinámicos. Los simuladores estáticos reflejan el rendimiento de una red celular en un instante de tiempo determinado, asemejándose a una fotografía del estado de red. Al prescindir de la variable temporal, los modelos que implementan son más sencillos y sus resultados suelen ser más fáciles de entender. Estas propiedades los hacen especialmente eficientes para la planificación radio, donde deben evaluarse grandes espacios de soluciones. Por el contrario, los simuladores dinámicos reflejan el comportamiento de la red como una sucesión de estados que guardan cierta correlación temporal entre sí [101]. Solo así es posible evaluar procedimientos cuyo resultado depende de estados anteriores, manteniendo

una relación de causalidad. Esta propiedad es imprescindible a la hora de evaluar algoritmos de gestión de recursos radio, como el traspaso, donde debe simularse la movilidad de los usuarios, la correlación temporal de la señal y la evolución de los recursos libres en el sistema.

En el proceso de validación de algoritmos de gestión de red, el último paso tras el modelado analítico y/o la simulación es la prueba de campo. Estas pruebas suelen restringirse a un entorno controlado para reducir riesgos y facilitar la extracción de conclusiones. Aun así, los operadores de red se muestran reticentes a cambiar la configuración por defecto de los equipos de red. A pesar de ello, existen trabajos en los que se han evaluado algoritmos de optimización de parámetros en la red real. Como ejemplo relacionado con esta tesis, en [21] se comprueba en una red LTE real cómo el reparto de tráfico a través de un algoritmo MLB que modifica los márgenes de traspaso puede degradar significativamente el rendimiento del sistema. De igual forma, en [102], se realizan pruebas de campo en una red LTE real para estudiar el rendimiento de un algoritmo diseñado para reducir la cantidad de traspasos ping-pong. Todas las contribuciones de este trabajo han sido validadas y verificadas mediante exhaustivas campañas de simulación. De forma complementaria, se han realizado algunas pruebas de campo en una red piloto para demostrar algunos principios básicos de funcionamiento de los métodos propuestos.

Capítulo 3

Reparto de tráfico en LTE basado en QoE

En este capítulo, se estudia el reparto de tráfico entre celdas vecinas con criterios QoE en un escenario macrocelular de un sistema LTE. Para ello, el capítulo se divide en cinco secciones. La sección 3.1 revisa el estado de la técnica de reparto de tráfico en redes celulares, prestando especial atención a los métodos de balance de carga que tienen en cuenta la QoE. La sección 3.2 formula el problema del reparto de tráfico mediante el cambio del margen de traspaso. Dicha formulación permite identificar las principales limitaciones del reparto de tráfico en LTE cuando el objetivo que se persigue es equilibrar u optimizar la QoE. Tras presentar el problema, en las secciones 3.3 y 3.5 se describen dos algoritmos de reparto de tráfico para redes de LTE macrocelulares, que constituyen las dos primeras contribuciones principales de esta tesis. Dentro de dicho apartado, en primer lugar se definen los algoritmos de ajuste, posteriormente se describen los experimentos realizados para validar su rendimiento, y, por último, se presentan algunas consideraciones de implementación. En la sección 3.6, se exponen las conclusiones derivadas de esta primera contribución.

3.1. Revisión del estado de la técnica

Las técnicas de reparto de tráfico mencionadas en el apartado 2.2 están basadas en indicadores simples de QoS, como, por ejemplo, la carga media de celda o la tasa de bloqueo de llamadas. En las redes móviles actuales, existen algoritmos de planificación de paquetes (*packet scheduling*)

para controlar la calidad de servicio. Estos algoritmos de planificación de recursos asignan de forma dinámica los recursos radio a las peticiones de datos de los usuarios dependiendo de criterios de QoS [103] [104]. Sin embargo, también existen planificadores de recursos radio más sofisticados que buscan explotar la ganancia de la diversidad multiusuario para conseguir un rendimiento óptimo de la red y asegurar justicia entre los usuarios al mismo tiempo [105]. Del mismo modo, distintos autores han propuesto planificadores de recursos radio que tienen en cuenta la QoE con el objetivo doble de optimizar la QoE media de la red asegurando una calidad de experiencia mínima para todos los usuarios. Estos planificadores avanzados suelen diseñarse para servicios particulares (p. ej., para un servicio de navegación web [106], un servicio de transmisión progresiva de vídeo [107] o un servicio de transmisión de vídeo con calidad adaptativa [108], [109], [110]). No obstante, el objetivo de la mayoría de los planificadores dinámicos de recursos en la interfaz radio es asegurar un nivel mínimo de QoS (o QoE) a aquellos usuarios con niveles más bajos o equilibrar la QoS (o QoE) media de cada servicio dentro de una misma celda, en lugar de igualar la QoE media de los servicios entre las distintas celdas de la red. De esta forma, no se garantiza el equilibrio de QoE entre los usuarios o servicios en el dominio espacial. Una red en la que la QoE esté desequilibrada entre sus celdas conlleva una asignación injusta de recursos radio: mientras que los usuarios plenamente satisfechos en celdas infrautilizadas malgastan los recursos disponibles, sin que ello suponga un incremento de su QoE, existen otros usuarios absolutamente insatisfechos en celdas muy sobrecargadas que necesitan esos recursos. En último término, esta distribución irregular de la satisfacción de los usuarios en las diferentes celdas del sistema puede traducirse en mayores tasas de abandono y pérdida de ingresos.

Además, la implementación de planificadores avanzados basados en QoE mencionados anteriormente requeriría la actualización de los equipos de red (en este caso, las estaciones base), algo indeseado por los operadores, que ya han realizado una importante inversión para actualizar la red a la última tecnología de acceso radio. Como alternativa, se puede ajustar un planificador dinámico estándar para mejorar la QoE del sistema. En esta línea, en [37], se propone un algoritmo de ajuste automático de los parámetros de un planificador multiservicio clásico con el objetivo de equilibrar la QoE entre distintos servicios dentro de una celda LTE cambiando las prioridades de los usuarios.

De forma similar, en [38], los mismos autores proponen un algoritmo de autoajuste diferente para el mismo planificador de recursos radio, cuyo diseño está basado en criterios de optimalidad. De esta forma, se asegura el mejor rendimiento del sistema en términos de QoE controlando el ajuste con estadísticas de rendimiento de la red. Sin embargo, ninguno de estos esquemas de autoajuste es capaz de equilibrar la QoE a lo largo de las celdas de la red. Aunque las técnicas clásicas de reparto de tráfico, que buscan equilibrar la carga media de celda, pueden potencialmente reducir las diferencias de QoE entre celdas, hasta donde se conoce, en la literatura no se han propuesto algoritmos de reparto de tráfico que tengan en cuenta la QoE explícitamente.

En este capítulo, se proponen dos nuevos algoritmos de reparto de tráfico basados en QoE para redes LTE. A diferencia de las técnicas tradicionales, el primer algoritmo tiene como objetivo minimizar las diferencias de QoE entre celdas y servicios ajustando los parámetros de traspaso adyacencia por adyacencia. Este ajuste se lleva a cabo mediante un controlador de lógica difusa dirigido por estimas de la QoE obtenidas a partir de indicadores de rendimiento de cada servicio. El segundo enfoque usa un algoritmo clásico de ascenso de gradiente para asegurar que los cambios de margen de traspaso, realizados por adyacencia, siempre mejoran la QoE global del sistema. Ambos algoritmos se validan en un simulador dinámico de nivel de sistema de una red LTE que implementa un escenario macrocelular realista. Las principales contribuciones de este capítulo son: a) descubrir las limitaciones de las técnicas clásicas de reparto de tráfico en redes celulares desde la perspectiva de la QoE, b) proponer un nuevo algoritmo de autoajuste de los márgenes de traspaso para equilibrar la QoE entre celdas vecinas mediante lógica difusa, c) proponer un algoritmo alternativo de autoajuste de los márgenes de traspaso para optimizar la QoE de los usuarios del servicio FTP mediante un algoritmo de ascenso de gradiente, y d) validar ambos algoritmos mediante simulaciones en un escenario macrocelular LTE realista.

3.2. Formulación del problema

En redes móviles, el proceso de traspaso asegura una conexión continua sin interrupciones a los usuarios que se desplazan entre celdas adyacentes. Específicamente, un traspaso tiene lugar cuando se cumple la siguiente condición

$$P_{rx}(j) - P_{rx}(i) \geq HOM(i,j)\,, \tag{3.1}$$

donde $P_{rx}(j)$ es el nivel de la señal piloto recibida desde la celda vecina j, $P_{rx}(i)$ es el nivel de la señal piloto recibida desde la celda servidora i, y $HOM(i,j)$ es el margen de traspaso (*HO Margin*) definido para la adyacencia entre las celdas i y j (es decir, un valor por cada par de celdas y sentido de la adyacencia). En la mayoría de los casos, los márgenes de traspaso se establecen de forma complementaria en ambas direcciones de la adyacencia para evitar inestabilidades, conocidas vulgarmente como efecto ping-pong. Así,

$$HOM(i,j) + HOM(j,i) = H\,, \tag{3.2}$$

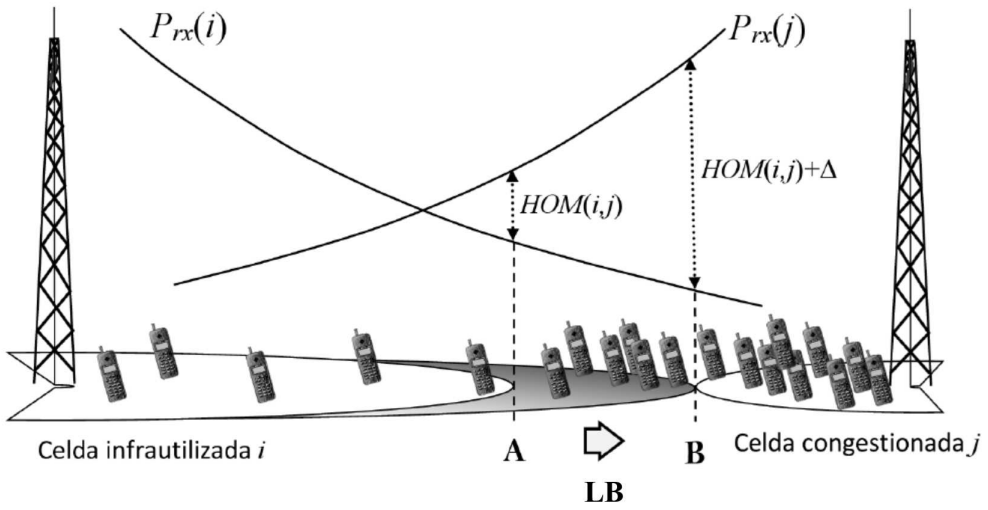


Figura 3.1: Reparto de tráfico mediante el ajuste de los márgenes de traspaso [21].

donde H representa el valor de histéresis.

La Figura 3.1 ilustra cómo la modificación de los márgenes de traspaso puede usarse como una técnica de reparto de tráfico [21]. Un incremento de Δ dB en $HOM(i,j)$ agranda el área de servicio de la celda i y reduce el de la celda j. Como resultado, el tráfico cursado (y la carga) total en la celda i aumenta, mientras que el tráfico cursado (y la carga) en la celda j disminuye. Al contrario, cuando se decrementa $HOM(i,j)$, se reduce el área de servicio de la celda i, mientras que se aumenta la de la celda j.

Las técnicas tradicionales de balance de carga (*Load Balance*, LB) modifican los márgenes de traspaso para igualar la carga entre celdas adyacentes con la esperanza de mejorar la tasa global de bloqueo de llamadas (o algún otro indicador de QoS) [12] [16]. Este efecto positivo, a menudo se consigue a expensas de deteriorar la eficiencia espectral de la red, ya que los usuarios se reasignan a celdas que no ofrecen el mayor nivel de señal [21].

Sin embargo, una red cargada uniformemente no implica necesariamente que la red esté equilibrada en términos de QoE. La satisfacción de usuario depende del servicio demandado, provocando que la QoE de los usuarios de diferentes servicios difiera significativamente, aun recibiendo la misma cantidad de recursos. Y, lo que es más importante, equilibrando la carga entre celdas vecinas no se reducen necesariamente las diferencias de QoE entre usuarios de diferentes celdas, debido a que algunos servicios son más sensibles a los incrementos de carga que otros (como se mostrará en

el siguiente apartado). Como consecuencia, equilibrar la carga entre celdas adyacentes no reduce necesariamente las diferencias de QoE entre usuarios de diferentes celdas si la mezcla de servicios (es decir, el porcentaje de conexiones de cada servicio) no es exactamente la misma en todas las celdas, como suele ocurrir en la práctica.

Estas consideraciones previas sugieren que un esquema tradicional de reparto de tráfico, basado en QoS (p. ej., tasa media de transmisión de datos por usuario), dará como resultado una red con carga equilibrada entre celdas, pero puede llevar a una situación en la que la distribución de la QoE de celda no esté equilibrada. Ésta es la hipótesis principal sobre la que se basa el primer algoritmo de balance de QoE de celda. El primer algoritmo propuesto tratará de mejorar la QoE de las celdas que presenten valores más bajos de satisfacción a costa de degradar la QoE de celdas vecinas donde los usuarios estén plenamente satisfechos.

No obstante, no existen garantías de que, equilibrando la QoE entre las celdas del sistema, se maximice (o incluso mejore) la QoE global del sistema. Por ejemplo, podría darse el caso de que, reasignando usuarios de las celdas sobrecargadas hacia celdas infrautilizadas vecinas, los usuarios que se mantienen en la celda congestionada siguieran insatisfechos y los usuarios de las celdas colindantes degradaran enormemente su rendimiento. Por ello, el segundo algoritmo de reparto de tráfico persigue optimizar la QoE global del sistema, incluso cuando dicha optimización pueda suponer una distribución menos justa de la QoE media de celda en la red. Para dotar al método de cierta garantía de optimalidad, este segundo algoritmo se diseña siguiendo una estrategia de ascenso de gradiente. Con ello, se asegura que cada pequeña modificación de los márgenes de traspaso conlleva una mejora de la QoE global del sistema, algo muy valorado por los operadores de redes comerciales.

3.3. Algoritmo de equilibrio de QoE

En esta sección se presenta un nuevo algoritmo de autoajuste cuyo objetivo es igualar la QoE entre las celdas de una red LTE. El algoritmo propuesto, de aquí en adelante llamado EB (*Experience Balancing*), se basa en el algoritmo de MLB descrito en [18] (de aquí en adelante llamado LB, *Load Balancing*). Al igual que LB, EB se basa en un conjunto de reglas de control implementadas mediante controladores de lógica difusa (*Fuzzy Logic Controllers*, FLC). Estos controladores deciden si incrementar o no el margen de traspaso, HOM, para cada adyacencia particular. A diferencia de los controladores clásicos Proporcionales-Integrativos-Derivativos (PID), los controladores de lógica difusa son más sencillos y fáciles de entender, ya que describen las reglas de control en lenguaje natural aprovechando la experiencia de los operadores de red. A diferencia de LB, EB busca igualar

la QoE media de celda, en vez de la carga media de celda, entre todas las celdas de la red. Con este objetivo, los usuarios se traspasan desde una celda con baja QoE media a otra celda vecina con mayor QoE media.

Se definen dos variantes del algoritmo EB: EB-C (C de celda) y EB-CS (C de celda y S de servicio).

3.3.1. Algoritmo EB-C

En la primera variante, EB-C, el indicador que se equilibra es la QoE media de celda, definida como

$$\overline{QoE}(i) = \frac{\sum_{s} \overline{QoE}(i,s)}{N_s(i)} \,, \tag{3.3}$$

donde $\overline{QoE}(i,s)$ es la QoE media de los usuarios del servicio s en la celda i, definida como

$$\overline{QoE}(i,s) = \frac{\sum_{\forall u \in i\,, S(u)=s} QoE^{(s)}(u)}{N_u(i,s)} \,, \tag{3.4}$$

donde $QoE^{(s)}(u)$ es la calidad de experiencia para el usuario u del servicio s, calculada como se indica en (2.1)-(2.4), N_s es el número de servicios demandados en la celda i y $N_u(i,s)$ es el número de usuarios en la celda i del servicio s. En (3.3), se asume de forma implícita que, para el operador, todos los servicios tienen la misma importancia. Esto está en consonancia con el objetivo de asegurar que todos los usuarios perciban la misma experiencia de su servicio, independientemente del servicio que se trate.

Finalmente, la QoE media de celda global de la red se define como

$$\overline{QoE} = \frac{\sum_{i=1}^{N_c} \overline{QoE}(i)}{N_C} \,, \tag{3.5}$$

donde N_C es el número de celdas en la red.

Como se indicó anteriormente, el objetivo de EB-C es equilibrar la QoE media de celda, descrita en (3.3). Por ello, se define como entrada del algoritmo EB-C un indicador en diferencias definido de la siguiente forma

$$QoE_{diff}(i,j) = \overline{QoE}(j) - \overline{QoE}(i) \quad . \tag{3.6}$$

La Figura 3.2 muestra la estructura del controlador de lógica difusa de EB-C, que se diseña como un controlador incremental que se ejecuta por adyacencia. La entrada es la diferencia entre las QoE media de las celdas de la adyacencia y la salida es el incremento/decremento del HOM entre las celdas vecinas i y j, $\Delta HOM(i,j)$. Como se aprecia en la figura, el controlador FLC se compone de tres etapas: la etapa de fuzzificación, la de inferencia y la de defuzzificación.

En la primera etapa de fuzzificación, el valor de la entrada, $QoE_{diff}(i,j)$, se asocia a adjetivos calificativos (p. ej., muy negativo, negativo, cero...) [66]. Esta asociación se lleva a cabo a través de las funciones de pertenencia de entrada, μ_x, como se muestra en la Figura 3.3a (x denota la variable lingüística particular). Por simplicidad, en esta tesis, se han seleccionado funciones de pertenencia de entrada triangulares. La experiencia demuestra que, en redes móviles, la flexibilidad que ofrecen funciones de pertenencia más complejas, no se traduce en un control más fino debido al ruido estocástico de las medidas utilizadas como entrada. Cabe reseñar que las funciones se superponen, lo que provoca que un valor de entrada numérico se asocie a uno o más adjetivos al mismo tiempo con distintos grados.

En la etapa de inferencia, un conjunto de reglas «SI-ENTONCES» definen la conversión de la entrada (medidas) en la salida (valor del parámetro regulado), todavía en términos lingüísticos. Una regla tiene la forma siguiente: 'si x es A, entonces y es B', donde A y B son adjetivos que describen la entrada y salida, respectivamente (p. ej., muy negativo, negativo, cero, positivo y muy positivo). La primera parte de la regla (x es A) es el *antecedente*, mientras que la segunda parte de la regla (y es B) es el *consecuente*. Al contrario de lo que ocurre en los sistemas expertos tradicionales, varias reglas pueden cumplirse al mismo tiempo en la fase de inferencia de un sistema de lógica difusa. La fuerza con la que se cumple cada regla depende del grado en que se satisfacen los antecedentes (denominados como el valor de *verdad* de la regla). Esta característica de los controladores de lógica difusa permite un proceso de regulación de parámetros más gradual.

La Figura 3.3c resume el conjunto de reglas que describe el proceso de ajuste en EB-C. A grandes rasgos, HOM se decrementa (es decir, $\triangle HOM(i,j)$ es N o MN) cuando la QoE media de la celda destino es mejor que la de la celda origen (es decir, $QoE_{diff}(i,j)$ es P o MP), y viceversa.

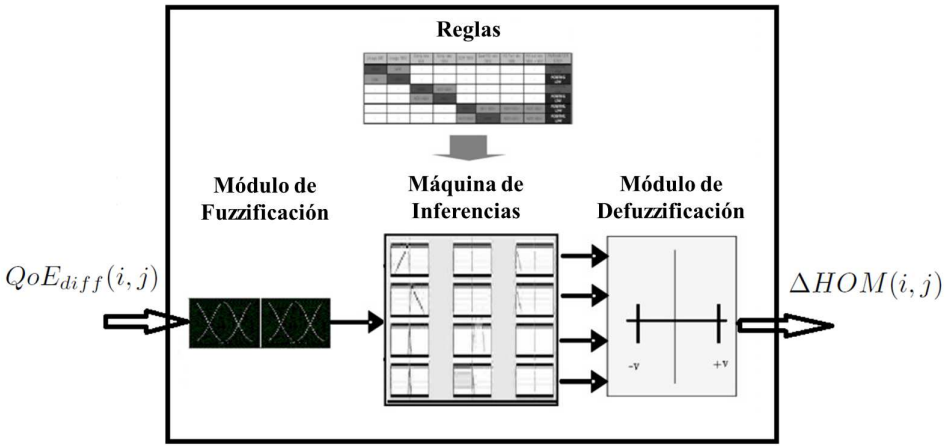


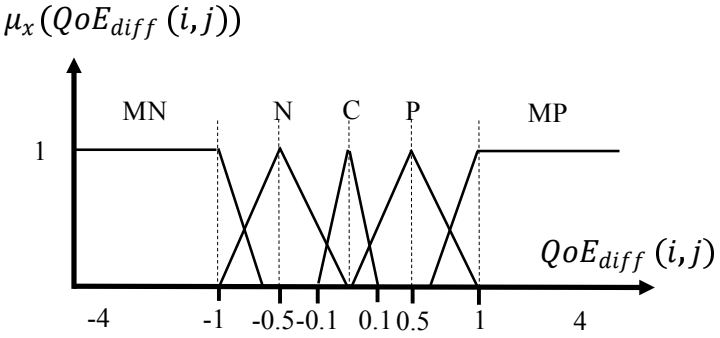
Figura 3.2: Estructura del controlador de lógica difusa [18].

Finalmente, en la etapa de defuzzificación, el valor numérico de salida se obtiene de la agregación de las reglas. En este trabajo, se aplica el método del *centro de gravedad* [111] para calcular el valor final de salida como una media ponderada de las salidas de las diferentes reglas que se disparan. Los pesos se obtienen a partir del valor de verdad de cada regla, calculados mediante el grado de cumplimiento de sus antecedentes. Por simplicidad, se usa el método Takagi-Sugeno [66], en el cual, las funciones de pertenencia de salida son constantes, como se observa en la Figura 3.3b. De esta forma, se consigue una representación más compacta y fácil de ajustar, reduciendo así la carga computacional. El uso de funciones de pertenencia de salida más complejas tiende a dar resultados similares.

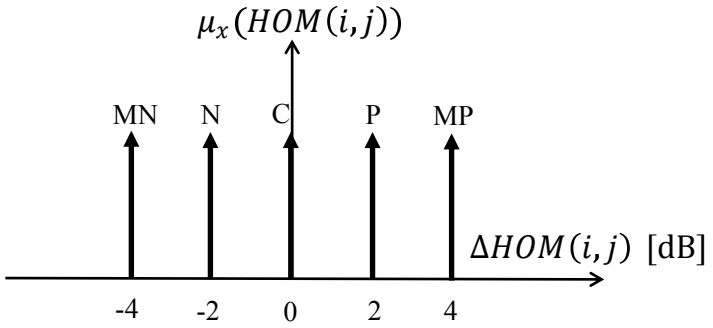
La máquina FLC definida anteriormente se introduce en un esquema iterativo de modificación de parámetros como sigue: se recopilan los indicadores de rendimiento de la red, se introducen sus valores en el FLC y se calcula el cambio de parámetros para la siguiente iteración. El valor de $HOM(i,j)$ para la siguiente iteración del algoritmo (de aquí en adelante, llamada lazo de optimización), se calcula como

$$HOM^{(n+1)}(i,j) = \min(\max(\text{round}(HOM^{(n)}(i,j) + \Delta HOM^{(n)}(i,j)), -7), 13)\,, \qquad (3.7)$$

donde los superíndices n y $n+1$ denotan el número de iteración de optimización, con todos los parámetros expresados en dB. En esta trabajo, los cambios de HOM se limitan al rango $[-7\,,\,13\,]$ dB. El límite inferior es el valor mínimo, para la mayoría de fabricantes, de la relación señal-interferencia-más-ruido (SINR) necesaria para que el planificador asigne recursos radio a una conexión. El límite



(a) Funciones de pertenencia de entrada.



(b) Funciones de pertenencia de salida.

$QoE_{diff}(i,j)$	$\Delta HOM(i,j)$
MP	MN
P	N
C	C
N	P
MN	MP

(c) Reglas.

Figura 3.3: Controlador de lógica difusa del algoritmo EB-C.

superior se calcula con (3.2) para asegurar un nivel de histéresis de $H = 6$ dB. De (3.2), se deduce también que cualquier modificación en $HOM(i, j)$ implica automáticamente un cambio en el sentido opuesto en $HOM(j, i)$. Como resultado, se necesita un único FLC en ambas direcciones de cada adyacencia. En consecuencia, el número necesario de FLC es igual al número de adyacencias en el sistema.

Por sus similitudes, EB-C y LB deberían funcionar del mismo modo (es decir, proponer los mismos cambios de HOM) si la mezcla de servicios fuera exactamente igual en todas las celdas. No obstante, como se explicará más adelante, la mezcla de servicios es diferente entre celdas en redes en explotación, lo que lleva a que EB-C y LB propongan acciones de ajuste diferentes.

Es importante destacar que, incluso cuando EB-C asegura que la QoE media de celda sea la misma en todas las celdas de la red (es decir, $\overline{QoE}(i) \simeq \overline{QoE}(j)\ \forall\ i, j$), algunos servicios pueden experimentar valores más altos de QoE que otros dentro de una misma celda (es decir, $\overline{QoE}(i, s_1) \neq \overline{QoE}(i, s_2)$), o diferente a la QoE que experimenta el mismo servicio en otras celdas (es decir, $\overline{QoE}(i, s_1) \neq \overline{QoE}(j, s_1)$). Para resolver esto, se propone una segunda variante del algoritmo de balance, denominada EB-CS, cuyo objetivo es equilibrar la QoE media de un servicio en una celda con la QoE media de las celdas adyacentes.

3.3.2. Algoritmo EB-CS

EB-CS aprovecha el hecho de que algunos fabricantes dan flexibilidad a los operadores para establecer valores de HOM diferentes para distintos servicios en una misma adyacencia. Por consiguiente, el ajuste de HOM puede llevarse a cabo por adyacencia y por servicio, guiado por un nuevo indicador en diferencias

$$QoE_{diff}(i, j, s) = \overline{QoE}(j) - \overline{QoE}(i, s)\ . \tag{3.8}$$

Al igual que para EB-C, $QoE_{diff}(i, j, s)$ en EB-CS se define por adyacencia, pero a a diferencia de lo que ocurre en EB-C, también se segrega por servicio. Por ende, en EB-CS, se necesita un FLC por servicio y adyacencia, encargado de proponer el cambio de margen de traspaso específico para cada servicio, $HOM(i, j, s)$. EB-CS se puede entender como cuatro mecanismos de equilibrio de QoE por adyacencia (uno por servicio), que pueden empujar a los usuarios de una celda en direcciones diferentes; por ejemplo, los usuarios del servicio WEB pueden traspasarse desde la celda i a la celda j, mientras que los usuarios del servicio VIDEO se traspasan desde la celda j hacia la

i. Las funciones de pertenencia y las reglas de inferencia son idénticas a las de EB-C, mostradas en la Figura 3.3, con la única diferencia de que el parámetro de entrada se desglosa por servicio, $QoE_{diff}(i,j,s)$, en vez de $QoE_{diff}(i,j)$.

Otra diferencia significativa de EB-CS frente a EB-C es la pérdida de simetría. El indicador en diferencias en (3.8) se calcula restando $\overline{QoE}(i,s)$ a $\overline{QoE}(j)$, con lo cual $QoE_{diff}(i,j,s) \neq QoE_{diff}(j,i,s)$. Por ello, al contrario de lo que ocurre en EB-C, en EB-CS se han de ejecutar dos FLCs para la misma adyacencia, una por cada sentido. La ejecución independiente de los controladores en dos celdas i y j puede llevar a cambios de diferente magnitud en ambos sentidos de la misma adyacencia, $\Delta HOM(i,j,s)$ y $\Delta HOM(j,i,s)$, haciendo que se incumpla (3.2). Para mantener un nivel de histéresis constante, en EB-CS se calcula un valor promedio de ΔHOM para ambos sentidos de la adyacencia como

$$\overline{\Delta HOM}^{(n)}(i,j,s) = -\overline{\Delta HOM}^{(n)}(j,i,s) = \frac{\Delta HOM^{(n)}(i,j,s) - \Delta HOM^{(n)}(j,i,s)}{2}\,. \tag{3.9}$$

Podría argumentarse que los esquemas propuestos están basados en QoS más que en QoE, pues las medidas de QoE usadas para dirigir el proceso de ajuste se derivan de medidas de QoS. Sin embargo, debe resaltarse que las funciones de utilidad que relacionan QoS con QoE no son lineales, con lo que un gran incremento de QoS no conduce necesariamente a un gran incremento de QoE. Esta no linealidad es crítica cuando se evalúa la ganancia del rendimiento global del sistema (en términos de satisfacción de usuario) originada por la reasignación de usuarios a una celda diferente. Este problema se evita registrando la QoE de forma explícita.

3.4. Análisis del rendimiento

Los algoritmos propuestos para equilibrar la QoE media de celda u optimizar la QoE global del sistema se prueban en un simulador dinámico de nivel de sistema de una red LTE [101]. En primer lugar, se esboza la metodología de evaluación, incluyendo la herramienta de simulación y una somera descripción de los experimentos realizados para analizar el rendimiento de los algoritmos propuestos. A continuación, y para mayor claridad, para cada experimento se presenta primero la configuración de simulación y después los resultados. Por último, se plantean algunas consideraciones de implementación relativas al coste computacional de ambos algoritmos.

Tabla 3.1: Parámetros de simulación.

Resolución temporal	10 TTI (10 ms)
Modelo de propagación	Pérdidas de trayecto COST 231 Hata, desvanecimiento lento (lognormal σ = 8 dB, d_{corr} = 20 m), desvanecimiento rápido (modelo ETU)
Modelo de estación base	Antena Tri-sectorial, DL MIMO 2x2, BW = 5 MHz (25 PRB), $f_{portadora}$ = 2 GHz, PIRE$_{max}$= 67 dBm.
Planificador	Exponencial clásico/proportional fair [105]
Adaptación al enlace	Basado en CQI

3.4.1. Metodología de evaluación

Las pruebas se han realizado con un simulador de red LTE dinámico de nivel de sistema cuyos parámetros más relevantes se muestran en la Tabla 3.1. Para el control de las condiciones de tráfico, el simulador implementa un escenario realista en el que se regula la carga ofrecida al sistema, la distribución espacial de usuarios y la distribución de servicios por celda.

Se llevan a cabo tres experimentos diferentes. El objetivo del primer experimento es comprobar que el resultado de equilibrar carga y equilibrar QoE es diferente en un escenario muy sencillo. Después, en un segundo experimento, se comparan las dos variantes del algoritmo de equilibrio de experiencia EB propuesto (EB-C y EB-CS) con otros métodos tradicionales de autoajuste en un escenario realista para evaluar su ganancia de rendimiento. El tercer experimento es una prueba de campo del algoritmo EB-C sobre la red LTE piloto del grupo de investigación *Mobilenet* de la Universidad de Málaga. Este tercer experimento pretende probar el algoritmo EB-C en una red real.

Por claridad, en todos los experimentos sólo se analiza el enlace descendente.

Primer experimento (prueba de concepto)

Configuración del experimento

Este experimento es una prueba preliminar cuyo objetivo es poner de manifiesto que equilibrar carga entre dos celdas adyacentes no implica equilibrar la QoE del sistema. La Figura 3.4 muestra el escenario usado en este primer experimento, que refleja un escenario regular trisectorial muy sencillo. En la figura se aprecia cómo, únicamente para esta prueba de concepto, la demanda de

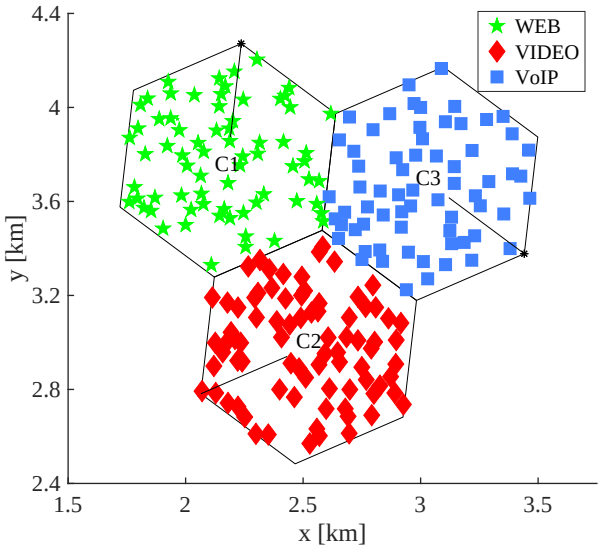


Figura 3.4: Escenario sencillo.

tráfico se confina en tres celdas (denotadas por C1, C2 y C3) y se fuerza a que todos los usuarios dentro de una misma celda correspondan al mismo servicio (WEB en C1, VIDEO en C2 y VoIP en C3). Las ubicaciones de los usuarios se representan por diferentes símbolos dependiendo de su servicio. Del mismo modo, se asume que todos los usuarios son estáticos en este escenario sencillo, para asegurar que los usuarios no cambian de celda por motivos de movilidad. Así, es más fácil segregar los usuarios por servicio, lo que facilita el análisis. Cabe destacar que, una vez que los usuarios se traspasan a otra celda como resultado del ajuste de HOM, en la misma celda se podrán encontrar usuarios de diferentes servicios.

En el escenario descrito, el primer experimento tiene dos fases. En la primera fase, se evalúa el impacto de la carga de red en la QoE de cada servicio. Con este fin, se realiza un barrido del tráfico ofrecido incrementando el número de usuarios en una de las tres celdas (p. ej., usuarios del servicio WEB en la celda C1) mientras que la intensidad del tráfico en las demás celdas se mantiene constante (p. ej.,VIDEO en C2 y VoIP en C3). Para cada valor de tráfico ofrecido, la carga de la celda i se mide como la utilización media de PRB (*Physical Resource Block*), $U(i)$, durante todo el periodo de simulación (es decir, 1 hora de tiempo de red). De igual modo, el tráfico ofrecido en las celdas con tráfico constante (VIDEO en C2 y VoIP en C3 en el ejemplo) se fija a un valor lo suficientemente elevado ($U(C2) \simeq U(C3) \simeq 80\,\%$) para generar un cierto nivel de interferencia de fondo. Los usuarios se distribuyen de manera uniforme dentro de cada celda. El mismo test se repite realizando un barrido de la intensidad de tráfico en las otras dos celdas. Es importante resaltar que,

en esta primera etapa, debido a la distribución espacial de los usuarios, $\overline{QoE}(i)$ coincide con la QoE media del servicio demandado en la celda bajo estudio (en este caso $\overline{QoE}(C1) = \overline{QoE}(C1, WEB)$).

En la segunda fase, el objetivo es demostrar las limitaciones del equilibrio de carga tradicional en términos de QoE. Con este fin, se realiza un barrido de $HOM(i, j)$ en pasos de 1 dB en una celda (p. ej., C3). De forma simultánea, se realiza un barrido de los márgenes de esta celda hacia las otras dos celdas (p. ej., $HOM(C3, C1) = HOM(C3, C2)$). La histéresis se mantiene sincronizando los cambios en ambos sentidos de las adyacencias para cumplir con (3.2).

Resultados

La Figura 3.5 muestra el análisis de sensibilidad de la QoE media de celda, $\overline{QoE}(i)$, con la carga de celda, $U(i)$ (es decir, la utilización media de PRB). Cada curva representa una de las tres celdas (servicios) en el escenario sencillo. Como era de esperar, condiciones de carga similares no conducen a los mismos valores de QoE en los tres servicios. Específicamente, $\overline{QoE}(C3) > \overline{QoE}(C1) > \overline{QoE}(C2)$ cuando $U(i) > 68\,\%$. Por lo tanto, se infiere que, para el algoritmo de *scheduling* implementado en el simulador, VoIP tiene mejor experiencia que WEB o VIDEO para altas cargas de celda. También puede observarse que la QoE del servicio VoIP se mantiene prácticamente constante y muy alta hasta cargas de celda muy elevadas (es decir, $\overline{QoE}(C3) \simeq 4.4\ \forall U(C3) \lesssim 97\,\%$). Lo mismo se puede decir de los servicios VIDEO y WEB, pero con umbrales de carga de celda más bajos ($U(i) \approx 58\,\%$ y $56\,\%$, respectivamente).

Los marcadores más grandes en la Figura 3.5 representan el punto de trabajo seleccionado para la siguiente fase del experimento, cuyo objetivo es mostrar el beneficio de equilibrar QoE. Esta configuración corresponde a una situación en la que las tres celdas tienen una carga similar cercana al $90\,\%$, pero completamente diferente en lo que se refiere a valores de QoE ($\overline{QoE}(C3) = 4.4$, $\overline{QoE}(C2) = 3.21$ y $\overline{QoE}(C1) = 2.71$). Dicha situación muestra un escenario de red con carga de celda equilibrada, pero desequilibrada en términos de QoE. Por ello, lo que se esperaría es que un algoritmo de equilibrio de carga no modificara los valores de HOM, aun existiendo grandes diferencias de QoE entre celdas. Por el contrario, un algoritmo de equilibrio de QoE cambiaría los márgenes de traspaso para igualar la QoE entre celdas.

En la segunda etapa de este experimento, los HOM se ajustan para mover el tráfico desde las celdas C1 y C2 (las dos celdas con peor QoE) hacia la celda C3 (la celda con mejor QoE). Este efecto se consigue agrandando el área de servicio de la celda C3, mediante el aumento de los HOMs en las adyacencias salientes de C3.

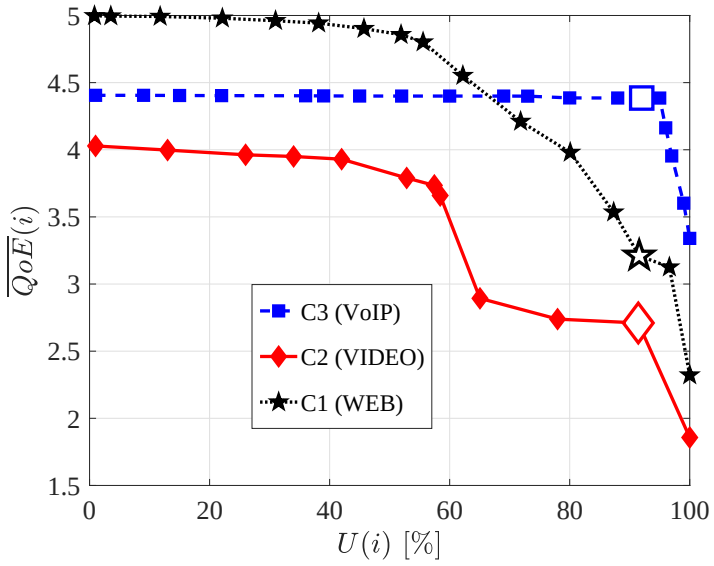


Figura 3.5: Dependencia de la QoE con la carga de celda.

La Figura 3.6 muestra la carga de celda (líneas continuas) y la QoE de celda (líneas discontinuas) para las tres celdas, cuando se realiza un barrido de $HOM(C3, C1)$ y $HOM(C3, C2)$ en pasos de 1 dB desde 3 (valor por defecto) hasta 11 dB. En la figura, se observa el comportamiento del desequilibrio de carga de celda y del desequilibrio de QoE.

El desequilibrio global de carga de celda se define como

$$\overline{U_{imb}} = \frac{1}{N_C}\sum_i |U_{imb}(i)| = \frac{1}{N_C}\sum_i \left| U(i) - \frac{\sum_{j\in A(i)} U(j)}{N_{ady}(i)} \right| , \tag{3.10}$$

donde $U_{imb}(i)$ es el desequilibrio de la utilización media de PRB de la celda i, obtenido comparando su utilización media de PRB con la de sus vecinas, $A(i)$ es el conjunto de celdas adyacentes de la celda i, $N_{ady}(i)$ es el número de celdas vecinas de la celda i y N_C es el número de celdas del escenario.

El desequilibrio global de QoE entre celdas y servicios del escenario se define como

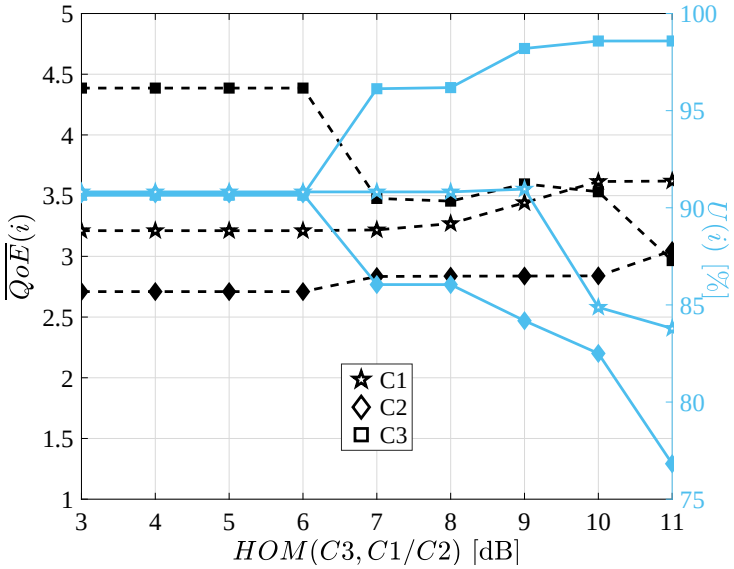


Figura 3.6: Sensibilidad de la carga de celda y la calidad de experiencia con los cambios de margen de traspaso.

$$\overline{QoE_{imb,c}} = \frac{1}{N_C}\sum_i |QoE_{imb,c}(i)| = \frac{1}{N_C}\sum_i \left| \overline{QoE}(i) - \frac{\sum_{j\in A(i)} \overline{QoE}(j)}{N_{ady}(i)} \right| , \tag{3.11}$$

donde $QoE_{imb,c}(i)$ es el indicador global de desequilibrio de QoE de la celda i, calculado mediante la comparación de su QoE media con la de sus celdas vecinas.

En este experimento, el desequilibrio de carga aumenta desde $\overline{U_{imb}} = 0.3\,\%$ hasta $12.7\,\%$ mientras que $\overline{QoE_{imb,c}}$ disminuye desde $0.96\,\%$ hasta $0.36\,\%$. Por tanto, un algoritmo de equilibrio de carga terminaría con $HOM(C3, C1)$=$HOM(C3, C2) = 3$ dB, sin embargo, un algoritmo de equilibrio de QoE lo haría en un punto de equilibrio completamente diferente con $HOM(C3, C1/C2) = 11$ dB. Ésta es una evidencia clara de que el equilibrio de carga y el equilibrio de QoE puede llevar al sistema a estados muy diferentes en presencia de diferentes mezclas de servicios en las distintas celdas.

Segundo experimento

El objetivo del segundo experimento es validar el comportamiento de las variantes del algoritmo de equilibrio de experiencia propuestas, comparando su rendimiento con el de otros algoritmos de reparto de tráfico propuestos en la bibliografía.

Durante el análisis, se comparan cinco algoritmos de autoajuste distintos. Los dos primeros son los algoritmos propuestos que persiguen igualar la experiencia de usuario entre celdas, EB-C, o entre celdas y servicios, EB-CS. Un tercer esquema es un algoritmo tradicional MLB [18], LB, cuyo objetivo es igualar la utilización media de PRB entre celdas vecinas. Para que la comparación sea justa, se incluye un cuarto algoritmo, denominado algoritmo de equilibrio de caudal (*throughput*), TB [17], para mostrar el beneficio de considerar la QoE de forma explícita en lugar de la QoS (*throughput* de usuario). TB persigue alcanzar el equilibrio del *throughput* medio de usuario $T(i)$, entre celdas modificando la configuración de los márgenes de traspaso para cada adyacencia. Con este propósito, se implementa un controlador de lógica difusa para dirigir a los usuarios desde celdas con menor *throughput* de usuario hacia celdas con mayor *throughput* de usuario. La entrada de este controlador es la diferencia de *throughput* medio de usuario entre celdas adyacentes y la salida es el cambio del margen de traspaso de la adyacencia. Finalmente, se incluye un quinto algoritmo de ajuste del *scheduler* destinado a la repriorización de servicios basada en QoE, QR [37]. QR persigue equilibrar la QoE de los usuarios dentro de una celda cambiando las prioridades de los servicios en un planificador clásico. El objetivo de introducir este algoritmo es mostrar el beneficio de redistribuir los usuarios entre celdas (como hacen EB-C y EB-CS) en lugar de re-priorizar los servicios dentro de una celda mediante la planificación dinámica de paquetes (como hace QR). Con este objetivo, se implementa un conjunto de cuatro controladores proporcionales (uno por servicio) para ajustar la prioridad de cada servicio a largo plazo de modo que se prioricen los usuarios con peor QoE. La entrada de cada controlador es la diferencia de QoE media de un servicio respecto al resto de servicios y la salida es el parámetro de prioridad de ese servicio en el planificador.

Para este segundo experimento, se implementa en el simulador un escenario basado en una red LTE real. La Figura 3.7 ilustra el escenario simulado, que consiste en 108-macroceldas (36 emplazamientos con 3 antenas trisectoriales por emplazamiento). La Tabla 3.2 muestra los principales parámetros de simulación, tomados de la red real. En este experimento, los usuarios se mueven en línea recta a una velocidad fija de 3 km/h con una dirección seleccionada aleatoriamente. De igual forma, los valores por defecto de los parámetros HOM y SPI (*Service Priority Index*) son 3 dB y 7, respectivamente. El parámetro SPI se asigna por servicio y permite al planificador dinámico de recursos radio re-priorizar cada uno de dichos servicios hasta cierto punto.

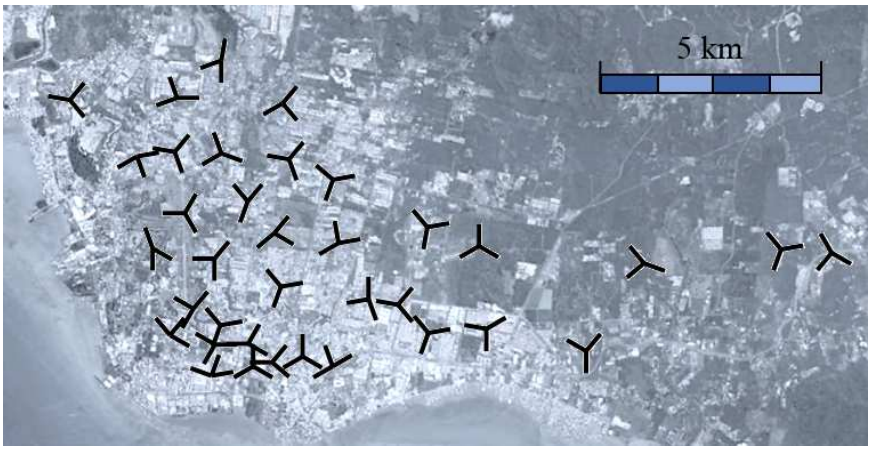


Figura 3.7: Escenario real.

Tabla 3.2: Parámetros del escenario real.

Ancho de banda	10 MHz (50 PRB)
Modelo de estación base	PIRE$_{max}$= 67 dBm, $f_{portadora}$= 1850 MHz
Modelo de tráfico	Distribución espacial de tráfico y mezcla de servicios basada en estadísticas de la red real recogidas por celda
Modelo de movilidad	Dirección aleatoria, velocidad constante, 3km/h
$HOM^{(0)}(i,j)$	3 dB $\forall\,(i,j)$
$SPI^{(0)}(i,s)$	7 $\forall\,(i,s)$

Los cinco algoritmos de autoajuste (LB, TB, QR, EB-C y EB-CS) se prueban a lo largo de 15 lazos de optimización. A posteriori, se comprueba que el sistema alcanza el régimen permanente tras las 15 iteraciones en los cinco algoritmos considerados. La duración de cada lazo de optimización (es decir, el ROP) es 1 hora, tiempo suficiente para asegurar estadísticas de rendimiento fiables. Al final de cada lazo de optimización, se recogen los indicadores usados como conductores del proceso ($U(i)$, $T(i)$, $\overline{QoE}(i,s)$, $\overline{QoE}(i)$ y $\overline{QoE}(i,s)$) y se ejecutan los algoritmos. Tras cada lazo de optimización, el sistema actualiza los valores de HOM o SPI y comienza un nuevo lazo de optimización. Para que la comparación sea justa, se asegura que todos los lazos de optimización para los cinco algoritmos se ejecutan bajo las mismas condiciones gracias a la pregeneración de una realización de todas las variables aleatorias. De esta forma, las diferencias de rendimiento entre lazos se deben solamente a diferentes configuraciones de HOM/SPI, y no a la naturaleza estocástica de la simulación. Como referencia para la evaluación del rendimiento de los distintos algoritmos, se toma el resultado del rendimiento de la red con la configuración por defecto de HOM/SPI.

El objetivo de los algoritmos propuestos (EB-C y EB-CS) es reducir las diferencias entre los usuarios de diferentes celdas y servicios. Esto se consigue mejorando la experiencia de los peores usuarios y servicios a costa de deteriorar la de los mejores usuarios y servicios. Por consistencia con el criterio de balance, en este experimento, la principal cifra de mérito es el percentil del 5 % de la distribución de QoE de todas las celdas y servicios de la red, $\overline{QoE}^{(5\,\%)}(i,s)$.

Una cifra de mérito secundaria es la QoE global del sistema, calculada como la media de todos los servicios y celdas del escenario,

$$\overline{QoE} = \frac{1}{N_C}\sum_i \overline{QoE}(i) = \frac{1}{N_C}\sum_i \frac{\sum_s QoE(i,s)}{N_s(i)} \,. \tag{3.12}$$

Además de los dos indicadores de rendimiento definidos en el apartado anterior, se definen tres indicadores de rendimiento adicionales para verificar las diferencias de rendimiento en las celdas y servicios de la red.

En primer lugar, se define un indicador global de desequilibrio de QoE intracelda como

$$\overline{QoE_{imb,f}} = \frac{1}{N_C}\sum_i |QoE_{imb,f}(i)| = \frac{1}{N_C}\frac{1}{N_s(i)}\sum_i\sum_k \left|\triangle\overline{QoE}(i,s_k)\right| \quad , \tag{3.13}$$

donde

$$\triangle\overline{QoE}(i,s_k) = \overline{QoE}(i,s_k) - \frac{\sum\limits_{s\neq s_k}\overline{QoE}(i,s)}{N_s(i)-1} \quad , \tag{3.14}$$

y $QoE_{imb,f}(i)$ es el desequilibrio de QoE entre servicios dentro de la celda i, calculado como el valor medio de la diferencia entre la QoE de un servicio y la QoE media del resto de servicios, como en (3.14).

De forma similar, se define un indicador global de desequilibrio de *throughput* como

$$\overline{T_{imb}} = \frac{1}{N_C}\sum_i |T_{imb}(i)| = \frac{1}{N_C}\sum_i \left|\overline{T}(i) - \frac{\sum\limits_{j\in A(i)}\overline{T}(j)}{N_{adj}(i)}\right| \quad , \tag{3.15}$$

donde $T_{imb}(i)$ es el indicador de desequilibrio de *throughput* de la celda i, obtenido mediante la comparación de su *throughput* medio con el de sus celdas adyacentes, y el *throughput* medio de la celda i, $\overline{T}(i)$, se define como

$$\overline{T}(i) = \frac{\sum\limits_s \overline{T}(i,s)}{N_s(i)} \, , \tag{3.16}$$

donde

$$\overline{T}(i,s) = \frac{\sum\limits_{u\in(i,s)} T(u)}{N_u(i,s)} \, , \tag{3.17}$$

y $T(u)$ es el *throughput* de conexión del usuario u. Igualmente, se define un indicador global de desequilibrio de QoE en las celdas del escenario como

Finalmente, se define un indicador global de desequilibrio de QoE entre las celdas del escenario como

Tabla 3.3: Rendimiento de red de referencia.

Indicador	Media	Máx	Mín
$N_u(i,VoIP)/N_u(i)\,[\%]$	1.8e-4	2.7e-3	0
$N_u(i,VIDEO)/N_u(i)\,[\%]$	37.21	54.96	5.52
$N_u(i,FTP)/N_u(i)\,[\%]$	27.09	72.14	1.83
$N_u(i,WEB)/N_u(i)\,[\%]$	35.79	43.22	22.33
$U(i)\,[\%]$	75	100	11.9
$\overline{QoE}(i)$	3.02	4.42	1.89

$$\overline{QoE_{imb,s}} = \frac{1}{N_C}\sum_i |QoE_{imb,s}(i)| = \frac{1}{N_C}\frac{1}{N_s(i)}\sum_i\sum_s \left|\overline{QoE}(i,s) - \frac{\sum_{j\in A(i)}\overline{QoE}(j)}{N_{ady}(i)}\right|, \tag{3.18}$$

donde $QoE_{imb,s}(i)$ es el indicador de desequilibrio de QoE de celda y servicio de la celda i, calculado comparando la QoE media de celda y servicio con la QoE media de sus celdas adyacentes.

Para mayor claridad, la Tabla 3.3 muestra algunos indicadores de rendimiento de la red con la configuración por defecto de HOM/SPI. Tanto la distribución espacial de los usuarios como la mezcla de servicios, incluida la probabilidad de que un usuario de un servicio inicie una conexión en una celda, se han tomado de las estadísticas de la red real del escenario. Únicamente la tasa de llegada de llamadas se ha modificado artificialmente (en este caso, aumentado) para generar un escenario muy cargado. De la tabla, se deduce que el servicio VIDEO es el más popular en toda la red, seguido por el servicio WEB. Igualmente, con la configuración por defecto de HOM y SPI, la diferencia de carga de dos celdas puede alcanzar un 88.1 % y la QoE media de celda puede variar hasta en 2.53 puntos de MOS, justificando la necesidad del balance de QoE. Se debe señalar que, en el escenario realista considerado, el tráfico del servicio VoIP es extremadamente bajo y se encuentra disperso entre unas pocas celdas de la red. Esta situación puede llevar a obtener estadísticas de QoE poco fiables, por lo que no se permite que el algoritmo EB-CS cambie los valores de HOM para este servicio (es decir, $HOM(i,j,VoIP) = 3$ en EB-CS para todas las iteraciones). Por el mismo motivo, tampoco se permite al algoritmo QR cambiar los valores de SPI para este servicio (es decir, $SPI(i,VoIP) = 7$ en QR para todas las iteraciones).

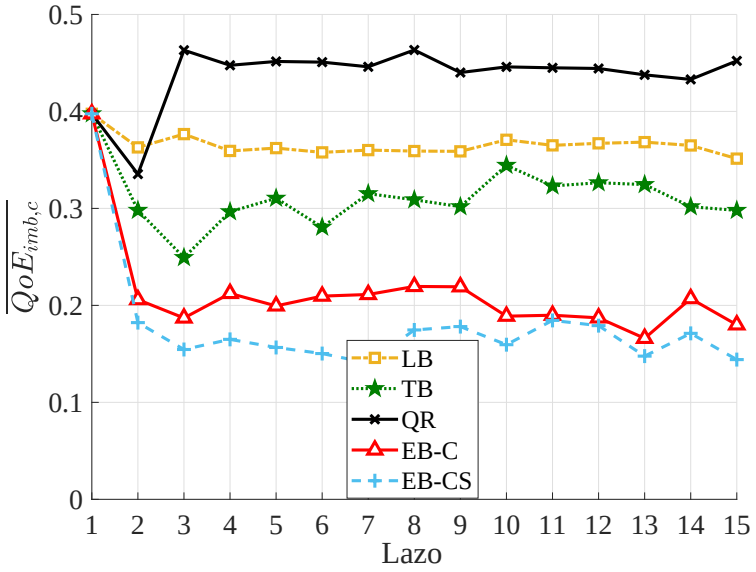


Figura 3.8: Evolución del desequilibrio de QoE.

Resultados

La Figura 3.8 presenta el impacto de los algoritmos LB, TB, QR y EB en el desequilibrio de QoE entre celdas a lo largo de 15 lazos de optimización. Como se ilustra, LB no produce cambios significativos de $\overline{QoE_{imb,c}}$, QR aumenta $\overline{QoE_{imb,c}}$ y TB consigue una leve reducción de $\overline{QoE_{imb,c}}$. Por el contrario, tanto EB-C como EB-CS reducen el desequilibrio inicial de QoE en más de la mitad.

Para mostrar las diferencias entre EB-C y EB-CS, la Figura 3.9 muestra la evolución del desequilibrio entre celdas y servicios, $\overline{QoE_{imb,s}}$, a lo largo de las iteraciones para ambos algoritmos. Tal como se esperaba, EB-CS equilibra mejor la QoE entre celdas y servicios de lo que lo hace EB-C, debido a su diseño centrado en los servicios. Para aclarar esta capacidad, se calcula la desviación media de los valores de HOM con respecto a su valor por defecto para EB-C y EB-CS como

$$\overline{\delta HOM} = \frac{\sum\limits_{(i,j,s)} \delta HOM(i,j,s)}{N_{adys}} = \frac{\sum\limits_{(i,j,s)} |HOM(i,j,s) - 3|}{N_{adys}}, \tag{3.19}$$

donde N_{adys} es el número total de adyacencias en la red.

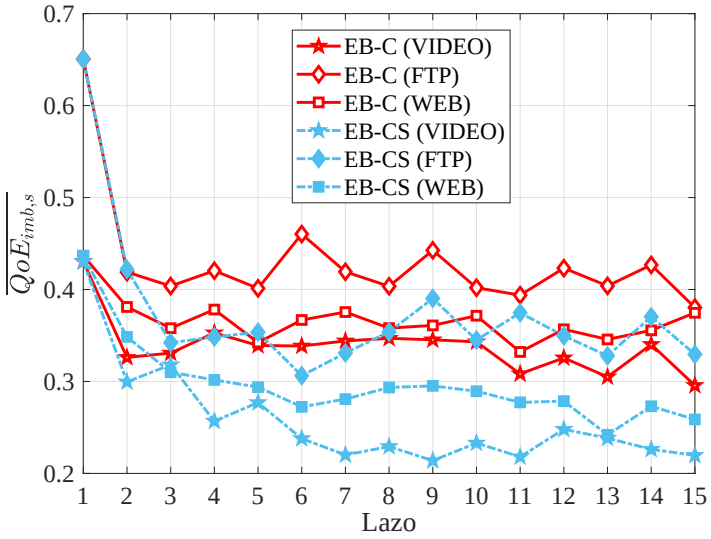


Figura 3.9: Evolución del desequilibrio de QoE por servicio.

La Figura 3.10 ilustra la desviación media de HOM a lo largo de las iteraciones para todos los algoritmos. En el lazo número 15, $\overline{\delta HOM}$ alcanza 7 dB, 5.6 dB y 0 dB para LB, TB y QR, respectivamente, lo que pone de manifiesto que LB y TB provocan un desplazamiento significativo de HOMs en un gran número de adyacencias. Por el contrario, como es lógico, QR no cambia los valores de HOM. EB-C y EB-CS también producen una desviación de HOMs en muchas adyacencias. En EB-C, $\overline{\delta HOM}$ alcanza 4.6 dB en el lazo número 15, que es un valor menor que el de la desviación que necesitan LB y TB para conseguir el equilibrio de carga o de *throughput*. Esto prueba que una red en la que la carga o el *throughput* de celda esté equilibrado no implica necesariamente que la QoE también esté equilibrada. Para EB-CS, $\overline{\delta HOM}$ varía desde 0 hasta 6 dB dependiendo del servicio concreto. Específicamente, el servicio de VIDEO requiere mayores desviaciones de HOM, lo que indica que, con la mezcla de servicios actual, se necesita (en media) traspasar una cantidad mayor de usuarios del servicio de VIDEO que de FTP o WEB para alcanzar el equilibrio de QoE entre servicios de celdas adyacentes.

Para facilitar la comparación, la Tabla 3.4 resume los principales indicadores de rendimiento al principio (columna *Inicial*) y final del proceso de ajuste (lazo número 15) para los distintos algoritmos. De acuerdo con lo esperado, LB consigue el mejor equilibrio de carga entre celdas, $\overline{U_{imb}}$, TB consigue el mejor equilibrio de *throughput* medio de usuario entre celdas, ($\overline{T_{imb}} = 0.28$ Mbps) y QR consigue el menor desequilibrio de QoE entre servicios dentro de una celda, ($\overline{QoE_{imb,f}} = 0.32$). EB-C alcanza mejor equilibrio de QoE que LB, TB o QR, ($\overline{QoE_{imb,c}} = 0.18$). Sin embargo, EB-CS

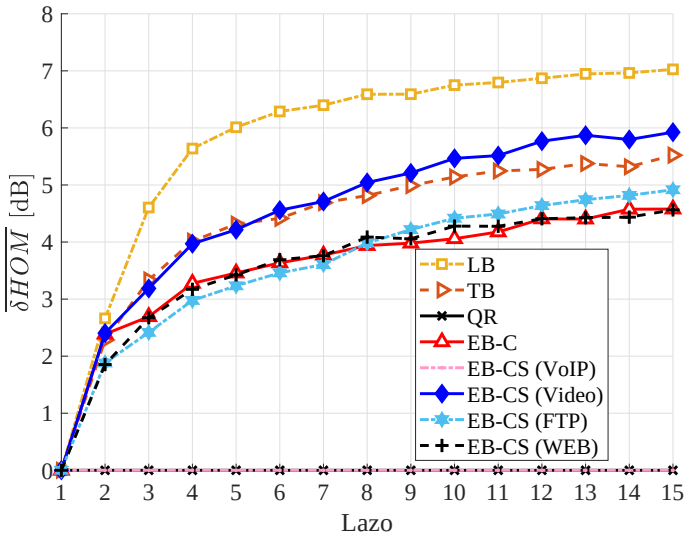


Figura 3.10: Evolución de la desviación media de HOM.

Tabla 3.4: Principales indicadores de rendimiento.

Indicador	Inicial	LB	TB	QR	EB-C	EB-CS
$\overline{U_{imb}}[\%]$	15.2	6.3	15.2	14.9	17	16.6
$\overline{T_{imb}}\ [Mbps]$	1.01	4.57	0.28	0.52	4.09	0.79
$\overline{QoE_{imb,f}}$	0.51	0.50	0.49	0.32	0.46	0.33
$\overline{QoE_{imb,c}}$	0.4	0.45	0.30	0.35	0.18	0.14
$\overline{QoE_{imb,s}}$	0.51	0.56	0.45	0.42	0.35	0.26
$\overline{QoE}$	3.02	2.94	2.96	2.99	2.94	2.93
$\overline{QoE}^{(5\%)}(i,s)$	1.89	1.90	2.01	2.07	2.10	2.36

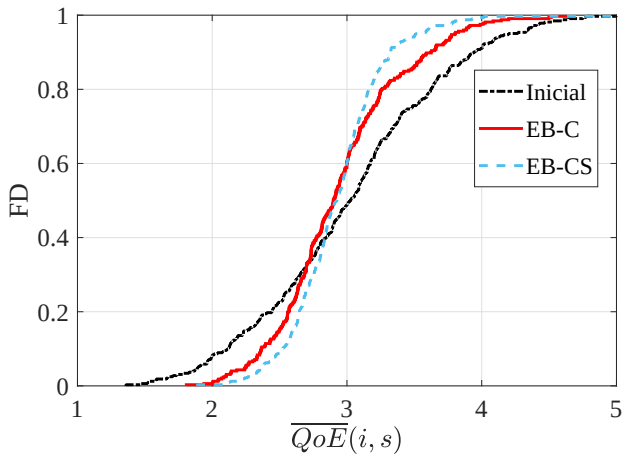


Figura 3.11: Distribución de la QoE de los servicios en las celdas.

consigue el menor desequilibrio de QoE entre celdas ($\overline{QoE_{imb,c}} = 0.14$) y servicios ($\overline{QoE_{imb,s}} = 0.26$). Este resultado proviene de ajustar los márgenes de traspaso por adyacencia y servicio.

Finalmente, la Figura 3.11 muestra la función de distribución de $\overline{QoE}(i,s)$ conseguida por EB y EB-CS. Se observa que ambos algoritmos de balance deterioran la QoE de las mejores celdas y servicios para mejorar la de las peores celdas y servicios. Centrando la atención en las peores celdas y servicios (la parte inferior izquierda), se observa que EB-CS alcanza la mayor mejora para aquellas celdas y servicios con valores más bajos de QoE. Este resultado se confirma en la Tabla 3.4, donde EB-CS tiene el valor más alto del indicador de QoE de borde de celda que representa a los peores usuarios (es decir, $\overline{QoE}^{(5\,\%)}(i,s) = 2.36$).

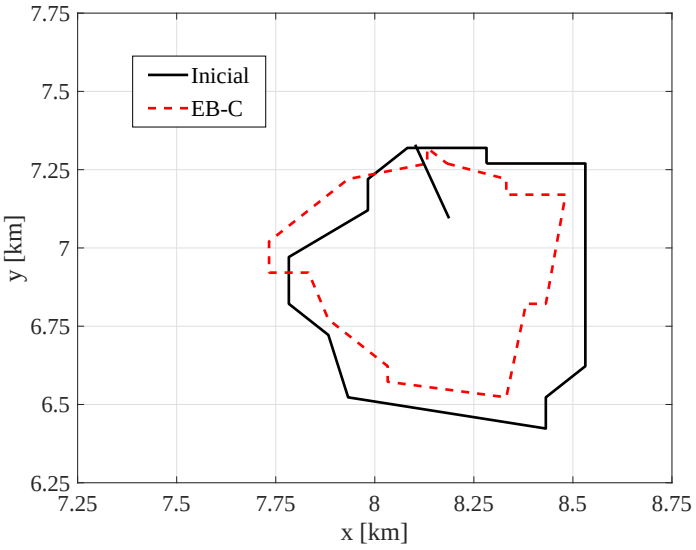
El mayor equilibrio de QoE con EB-CS se consigue alterando el área de servicio de las celdas no sólo por celda sino también por servicio. La Figura 3.12a ilustra la forma en que el área de servicio de una celda se modifica por el algoritmo EB-C al final del proceso de ajuste (en el lazo número 15). El algoritmo EB-C regula los márgenes de traspaso solamente por adyacencia. Por ello, el área de servicio de la celda es la misma para todos los servicios. La Figura 3.12b representa el área de servicio de la misma celda con EB-CS en el lazo número 15. El algoritmo EB-CS modifica los márgenes de traspaso por adyacencia y por servicio. Por tanto, el área de servicio de la celda es diferente dependiendo del servicio (VIDEO, FTP o WEB). No se presenta el caso del servicio VoIP porque el tráfico VoIP en la red analizada es despreciable. Cabe señalar que el área de servicio obtenida

Tabla 3.5: Rendimiento del sistema ante cambios en la red.

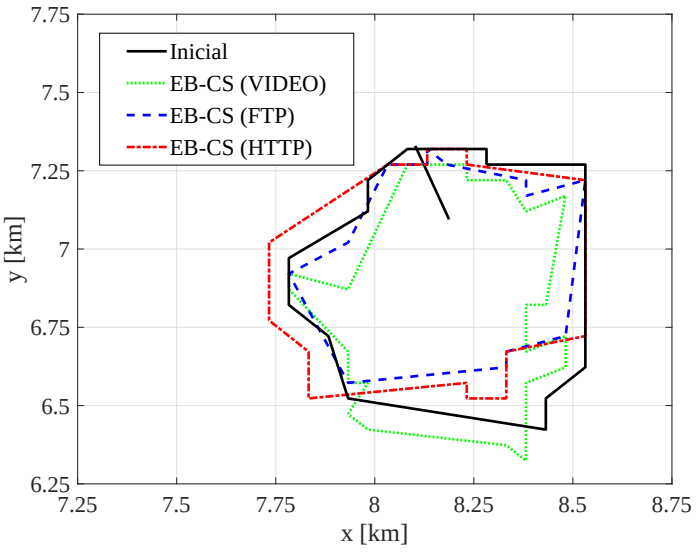
Etapa	Condiciones iniciales de red		Aumento de la carga de red, $\Delta\lambda$		Cambio en la mezcla de servicios, $\Delta\lambda(s)$	
Lazo de optimización	1	15	16	26	27	45
$\overline{QoE_{imb,s}}$	0.51	0.26	0.43	0.26	0.48	0.29
$\overline{QoE_{imb,c}}$	0.4	0.14	0.29	0.15	0.32	0.15

con EB-C, que se representa en la Figura 3.12a, es completamente diferente de las obtenidas con de EB-CS, presentadas en la Figura 3.12b. Así, la flexibilidad que permite el hecho de poder cambiar el área de servicio de las celdas por adyacencia y por servicio es la razón de la superioridad de EB-CS a la hora de igualar la QoE entre celdas y servicios.

Dentro de este segundo experimento, se ha llevado a cabo una comprobación adicional para comprobar la capacidad de adaptación del algoritmo iterativo propuesto (EB-CS) a cambios en el número de usuarios y en la mezcla de servicios. Para ello, una vez que el sistema ha alcanzado el equilibrio de QoE, se modifica el número de usuarios incrementando la tasa de llegada de la distribución de Poisson de cada servicio en un 4 %. Este cambio modifica el punto de equilibrio, con lo que se espera que aparezca de nuevo desequilibrio. Si así ocurre, EB-CS comenzará de nuevo a modificar los márgenes de traspaso buscando el nuevo punto de equilibrio. Tras esta segunda etapa de búsqueda del equilibrio, se hace una nueva modificación, esta vez alterando la mezcla de servicios cambiando el porcentaje de los servicios FTP, VIDEO y WEB. El tráfico WEB se reduce en un 24 %, mientras que el tráfico FTP y VIDEO se incrementan cada uno en un 12 %. La Tabla 3.5 resume los resultados con estos cambios de tráfico, mostrando el indicador de desequilibrio de QoE, $\overline{QoE_{imb,c}}$, y el valor de desequilibrio de QoE por servicio, $\overline{QoE_{imb,s}}$, al comienzo/final de cada etapa. Se observa que cualquier cambio en las condiciones de la red provoca un desequilibrio temporal de QoE entre celdas y servicios. Específicamente, $\overline{QoE_{imb,s}}$ pasa de 0.26 a 0.43 entre los lazos 15 y 16, mientras que $\overline{QoE_{imb,c}}$ pasa de 0.19 a 0.29. Con la alteración de mezcla de servicios (lazos 26 y 27), $\overline{QoE_{imb,s}}$ pasa de 0.26 a 0.48 y $\overline{QoE_{imb,c}}$ pasa de 0.15 a 0.32. Estos desequilibrios son corregidos satisfactoriamente por EB-CS tras algunos lazos (10 lazos para el primer cambio y 18 para el segundo), llevando el rendimiento de la red a un punto de equilibrio similar al de la primera etapa del experimento en el lazo de optimización número 15. Más concretamente, $\overline{QoE_{imb,s}} = 0.26$ y 0.29 en los lazos 26 y 45, respectivamente, frente a 0.26 en el lazo 15, y ($\overline{QoE_{imb,c}} = 0.15$ en ambos lazos 26 y 45, frente a 0.14 en el lazo 15. En consecuencia, se demuestra que EB-CS es capaz de manejar fluctuaciones en la demanda del tráfico durante el día.



(a) Inicial y EB-C.



(b) EB-CS.

Figura 3.12: Área de celda por servicio.

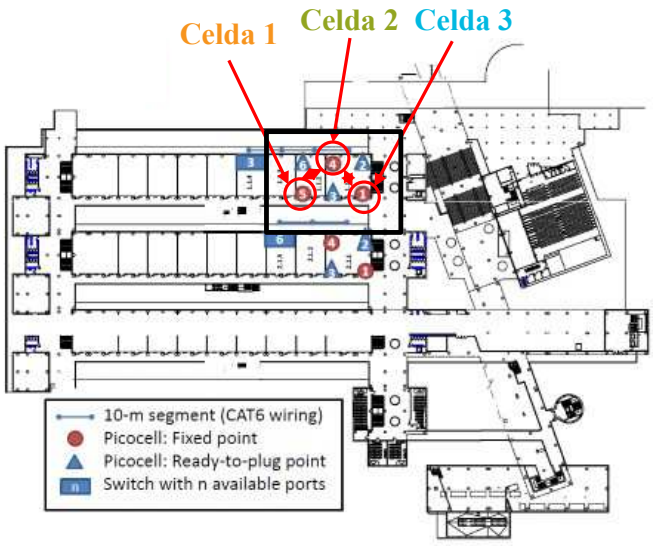


Figura 3.13: Red LTE piloto.

Tercer experimento

Este tercer experimento pretende evaluar el rendimiento del algoritmo EB-C en una red real. Como sistema de prueba, se utiliza la red LTE piloto del grupo de investigación *Mobilenet* de la Universidad de Málaga. Dicha red está compuesta por 12 picoceldas de interior (modelo BTS3911B de Huawei) conectadas a un núcleo de red compacto (modelo eCNS600 de Huawei), un servidor Wifi y un equipo de gestión de red (modelo RH5885 de Huawei). Todos estos elementos están conectados a través de un conmutador. Las tareas de gestión de la red de forma remota, entre las que se encuentra la modificación de los márgenes de traspaso, se realiza mediante la aplicación iManager U2000. Los equipos están localizados en distintas estancias de la Escuela Técnica Superior de Ingeniería de Telecomunicación en la Universidad de Málaga. La Figura 3.13 muestra las 3 picoceldas ubicadas en la primera planta del edificio que se utilizan en los experimentos. La celda 1 está en el laboratorio 1.1.2 y las celdas 2 y 3 en el laboratorio 1.1.1. La configuración de las picoceldas de la red se presenta en la Tabla 3.6. La potencia de la señal de referencia se ha disminuido para minimizar el solape de las áreas de celdas vecinas, y así limitar la zona en que se produce el traspaso al cambiar los márgenes.

En este experimento se persigue conseguir un escenario con una celda sobrecargada y dos celdas infrautilizadas. Para ello, se posicionan de forma espacialmente irregular 3 usuarios de VIDEO conectados inicialmente a la celda 2 como se ilustra en la Figura 3.14. Uno de estos usuarios se localiza en el pasillo, en el borde de la celda 2 con la celda 1, otro en el borde de la celda 2 con la celda

Tabla 3.6: Configuración de las picoceldas.

Ancho de banda	5 MHz (25 PRB)
Potencia de la señal de referencia	-20 dBm
Potencia de transmisión eNB	23 dBm
Frecuencia de portadora	2.516 GHz
Algoritmo de *scheduling*	Proportional Fair (PF)
HOM por defecto	3 dB

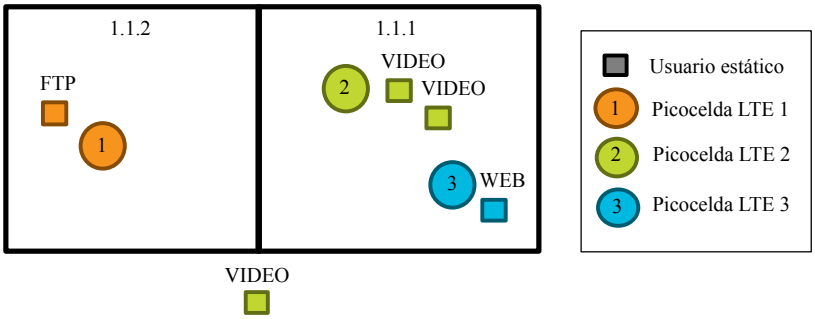


Figura 3.14: Plano de la situación inicial de la red.

3 y el tercer usuario se posiciona aproximadamente en el centro de la celda 2. La celda 1 tiene un usuario FTP conectado y la celda 3 da servicio a un usuario WEB. Todos los usuarios son estáticos. El terminal de usuario utilizado es el modelo Huawei P8 con la aplicación de monitorización del rendimiento G-MON [112]. Dicha aplicación permite medir la cobertura, capacidad y QoS de una red celular. El tráfico de usuario, así como la obtención de los S-KPIs necesarios (*throughput*, *stallings*), se controlan y registran mediante un programa externo desarrollado en Linux. La QoE de los servicios se estima mediante las funciones de utilidad 2.2-2.4. Los principales parámetros de los modelos de tráfico de los servicios en este programa se muestran en la Tabla 3.7. En esa situación inicial, la QoE de la celda 2 será mucho menor que la de las celdas 1 y 3. Esto se debe a que el servicio de VIDEO genera mayor cantidad de tráfico que los servicios FTP y WEB y a que la celda 2 da servicio a dos usuarios más que las celdas 1 y 3. Modificando los márgenes de traspaso según EB-C, se reducirá el desequilibrio de QoE hasta alcanzar un punto de equilibrio. El experimento llevado a cabo tiene una duración de 10 minutos. Durante este tiempo, el vídeo, cuya duración es algo mayor que 10 minutos, se reproduce una vez, el fichero FTP se descarga una vez y la página web de la Universidad de Málaga se descarga 800 veces de forma continua.

Tabla 3.7: Parámetros de los modelos de tráfico.

VIDEO	Duración de la secuencia: 600 s. Resolución: 3840x2160 píxels. Régimen binario medio: 14931 kbps. Tasa de fotograma: 25 cuadros/s.
FTP	Tamaño fijo de fichero: 900 MB.
WEB	800 descargas continuas de www.uma.es.

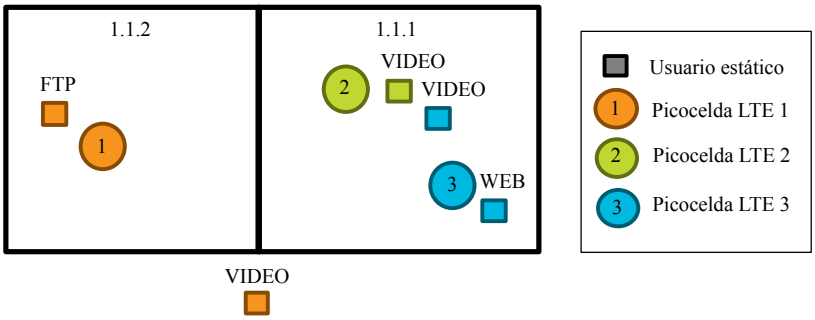


Figura 3.15: Situación de la red tras EB-C.

Resultados

La Figura 3.14 muestra la situación de la red cuando los márgenes de traspaso entre todas las celdas tiene su valor por defecto (3 dB). A partir de los indicadores de rendimiento del servicio, se calculan los valores de QoE para cada usuario. Se comprueba que, con esta disposición inicial, $\overline{QoE}(C1) = 4.25$, $\overline{QoE}(C2) = 1$, y $\overline{QoE}(C3) = 5$, con lo que $\overline{QoE_{imb,c}} = 3.625$.

La Figura 3.15 muestra la situación de la red tras ejecutar dos lazos de EB-C. Al finalizar el proceso de optimización, los márgenes de traspaso de la celda 2 a las celdas 1 y 3 se reducen, contrayendo el área de servicio de la celda 2. Concretamente, $HOM(C2, C1) = -5$ dB y $HOM(C2, C3) = -5$ dB. Los márgenes de traspaso en el sentido contrario de la adyacencia se ajustan según (3.2), $HOM(C1, C2) = 11$ dB y $HOM(C3, C2) = 11$ dB. De esta forma, se consigue traspasar un usuario de VIDEO a la celda 1 y otro a la celda 3, que ven sus áreas de servicio expandidas. Con esta configuración, $\overline{QoE}(C1) = 2.62$, $\overline{QoE}(C2) = 2.3$, y $\overline{QoE}(C3) = 3$, con lo que $\overline{QoE_{imb,c}} = 0.36$, demostrando la capacidad de EB-C de reducir el desequilibrio de QoE entre celdas vecinas.

3.4.2. Consideraciones de implementación

El algoritmo EB-C se ejecuta por adyacencia, y, por lo tanto, su complejidad temporal asintótica es $O(N_{adjs})$. Por el contrario, el algoritmo EB-CS se ejecuta por adyacencia y por servicio, con lo que su complejidad temporal es $O(N_{adjs} \times N)$. En este trabajo, ambos algoritmos se implementan con el Toolbox Fuzzy Logic de Matlab. En el escenario considerado de 108 celdas, 11664 adyacencias y 3 servicios, el tiempo medio de ejecución de todas las iteraciones de EB-C y EB-CS es 2.8 y 6.5 segundos en un ordenador personal con un procesador de ocho núcleos de 3.6-GHz de frecuencia de reloj y 16 GB de RAM.

3.5. Algoritmo de optimización de QoE

Los algoritmos presentados en la sección anterior pretenden equilibrar la QoE entre celdas de distintas formas. Como contrapartida, se deteriora la experiencia de usuario en algunas celdas y servicios para poder mejorar la experiencia de aquellos usuarios con peor rendimiento. La Figura 3.11, mostrada anteriormente, ilustraba la forma en que la QoE se ve afectada por EB-C y EB-CS. Para ello, se presenta la función de distribución de $\overline{QoE}(i,s)$ antes y después del proceso de balance. Los resultados muestran que, con EB-C y EB-CS, se consigue una distribución equilibrada de QoE, pero la QoE global del sistema se degrada, como puede intuirse del desplazamiento hacia la izquierda de la mediana en la Figura 3.11. Forzando el sistema para que todas las celdas (o celdas y servicios en el algoritmo EB-CS) tengan la misma QoE, se da prioridad a las celdas más cargadas con respecto a las poco cargadas. Sin embargo, y en el escenario probado, el incremento de QoE en las primeras celdas es menor que el decremento de QoE en las últimas.

De forma alternativa, es posible modificar los márgenes de traspaso con el objetivo de optimizar la QoE global, si se dispone de un modelo analítico de rendimiento de la red para evaluar las posibles modificación de los márgenes. De esta forma, se incluye un criterio de optimalidad en el proceso de ajuste de parámetros. Desafortunadamente, el número de factores que influyen en la QoE es enorme, por lo que solamente se pueden utilizar modelos aproximados de rendimiento. Aun así, siempre que el modelo aproximado sea capaz de encontrar estimaciones razonables de la función objetivo, la QoE global, puede usarse un algoritmo clásico de ascenso de gradiente para mejorar progresivamente dicha cifra de mérito global. De este modo, se alcanza un máximo local del problema, que en la mayoría de los casos es suficiente para el operador. La regla de ascenso de gradiente asegura que ningún cambio pequeño del margen de traspaso degrada el rendimiento del sistema, lo cual es fundamental para los operadores de red.

El algoritmo de optimización de QoE que se diseña, de aquí en adelante llamado OE (Optimización de la Experiencia), modifica los márgenes de traspaso entre las celdas vecinas$i$ y j, $HOM(i,j)$, con el objetivo de maximizar la QoE global de usuario, $\overline{QoE}_u$, definida como

$$\overline{QoE}_u = \frac{\sum_u QoE(u)}{N_u}\,, \tag{3.20}$$

donde N_u es el número de usuarios en el sistema y $QoE(u)$ es la QoE que experimenta cada usuario u en su conexión.

Para optimizar la QoE global de usuario, el algoritmo OE se basa en un algoritmo de ascenso de gradiente, en el que se cambia el margen de traspaso $HOM(i,j)$ en función de estimas del gradiente de la función objetivo, $\overline{QoE}_u$, calculadas por adyacencia mediante un modelo analítico. En cada iteración (lazo de optimización), los márgenes de traspaso de todas las adyacencias se ajustan de forma incremental según la ecuación

$$HOM^{(n+1)}(i,j) = HOM^{(n)}(i,j) + \Delta HOM^{(n)}(i,j) = HOM^{(n)}(i,j) + f(\frac{\delta\overline{QoE}_u}{\delta HOM(i,j)})^{(n)})\,, \tag{3.21}$$

donde los superíndices (n) y $(n+1)$ denotan el índice del lazo de optimización, $\frac{\delta\overline{QoE}_u}{\delta HOM(i,j)}$ es el gradiente de la función objetivo en la dirección de la variable de decisión $HOM(i,j)$, que cuantifica el impacto de incrementar el margen de traspaso en una adyacencia sobre la QoE global del sistema, y $f(\cdot)$ es la función que describe el comportamiento entrada-salida del controlador que regula los márgenes de traspaso. Los valores de HOM resultantes se limitan dentro del intervalo $[-7, 13]$ dB para asegurar un mínimo valor de calidad de señal después del traspaso [14]. El límite inferior es el mínimo valor de SINR requerido por el planificador de recursos para asignar recursos a una conexión. El límite superior se calcula mediante (3.2) para garantizar un nivel de histéresis de $H = 6$ dB. $\Delta HOM^{(n)}(i,j)$ es el valor a estimar a partir del modelo analítico descrito a continuación.

3.5.1. Modelo analítico de rendimiento del sistema

El objetivo del algoritmo de optimización es maximizar la QoE media de todos los usuarios de la red. Para ello, se ha desarrollado un modelo analítico de rendimiento del sistema que permite realizar estimaciones del gradiente de la función objetivo, permitiendo así sólo aquellas modificaciones

de márgenes de traspaso que mejoren dicha función objetivo. Por simplicidad, en su desarrollo, se supone que los usuarios activos tienen infinitos datos que transmitir, pudiendo modelarse su demanda de recursos como una fuente de tráfico de búfer lleno (*full buffer*). Asimismo, para enriquecer el modelo, en la formulación del problema se considera que la experiencia de usuario depende del contexto de usuario (interior o exterior), que, en la práctica, se puede inferir analizando las trazas de conexión [113]. Para este análisis, por tanto, se usan dos funciones de utilidad para el mismo servicio, dependiendo de la ubicación del usuario [38]. Para los usuarios de exterior, la QoE se estima como

$$QoE_{exterior}^{(FTP)}(u) = \text{máx}(1, \text{mín}(5, 6.5 \cdot T(u) - 0.54)) \ , \tag{3.22}$$

donde T es el *throughput* medio de usuario del enlace descendente en Mbps. Para los usuarios de interior, la QoE se estima como

$$QoE_{interior}^{(FTP)}(u) = \text{máx}(1, \text{mín}(5, 6.5 \cdot \frac{T(u)}{1.5} - 0.54)) \ . \tag{3.23}$$

En (3.22)-(3.23), la QoE se limita a la escala MOS (de 1 a 5). Comparando (3.22) y (3.23), se observa que, en la segunda ecuación, el *throughput* T se divide entre 1.5, reflejando que los usuarios de interior experimentan peor QoE para el mismo valor de T, debido a las mayores expectativas de los usuarios de interior con relación al servicio.

En las funciones de utilidad descritas, la QoE de usuario depende únicamente del caudal de descarga, T. Por lo tanto, el modelo analítico ha de establecer solamente la relación entre los cambios de HOM y de T, que se pueden traducir directamente a cambios de QoE. Concretamente, el gradiente de la QoE media de usuario en toda la red se calcula por adyacencia agregando el impacto de los cambios en todos los usuarios de la adyacencia, según la ecuación

$$\begin{aligned} \frac{\delta \overline{QoE}_u}{\delta HOM(i,j)} &= \sum_u \frac{\delta QoE(u)}{\delta HOM(i,j)} = \\ &= \sum_u \left[\frac{\delta QoE(u)}{\delta T(u)} \frac{\delta T(u)}{\delta EE(u)} \frac{\delta EE(u)}{\delta SINR(u)} \frac{\delta SINR(u)}{\delta HOM(i,j)} + \right. \\ &\left. + \frac{\delta QoE(u)}{\delta T(u)} \frac{\delta T(u)}{\delta BW(u)} \frac{\delta BW(u)}{\delta N_{su}(k)} \frac{\delta N_{su}(k)}{\delta A(k)} \frac{\delta A(k)}{\delta HOM(i,j)} \right] , \end{aligned} \tag{3.24}$$

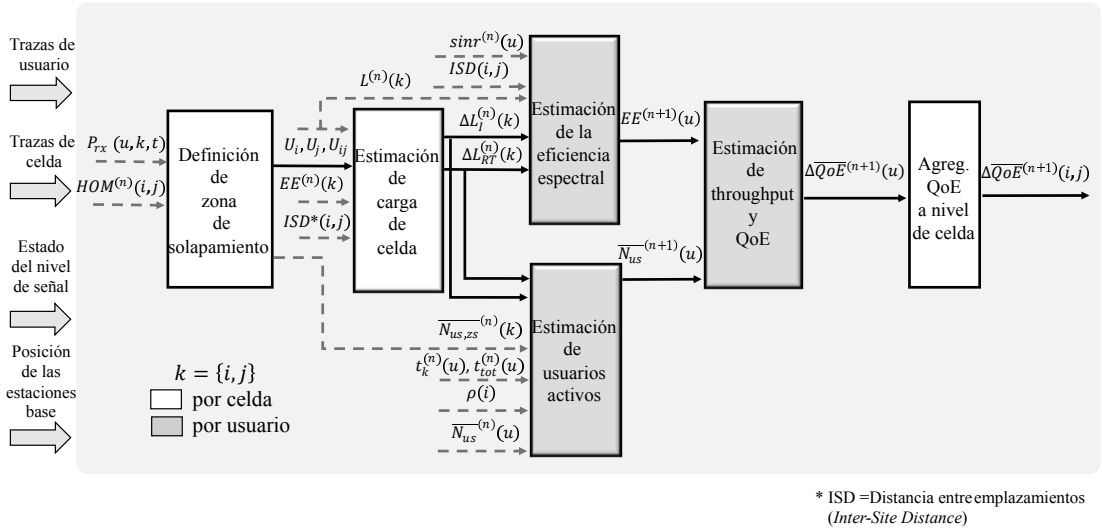


Figura 3.16: Diagrama de flujo del cálculo de cambios de margen de traspaso por adyacencia a partir del modelo analítico de rendimiento.

donde k es la celda servidora del usuario u (es decir, celda i o j), $BW(u)$ es el ancho de banda medio del sistema asignado al usuario, $N_{su}(k)$ es el número medio de usuarios activos simultáneos con el usuario u en la celda servidora (excluyendo los periodos de inactividad), $A(k)$ es el área de servicio de la celda k, y $EE(u)$ y $SINR(u)$ son la eficiencia espectral media y la calidad de señal del usuario u. La regla de la cadena en (3.24) refleja que cualquier cambio de *throughput* de usuario que se consigue mediante el reparto de tráfico se debe a dos motivos: a) un cambio en las condiciones del enlace radio (experimentado, por ejemplo, por un usuario reubicado en una celda nueva), o b) un cambio en el número de recursos disponibles para el usuario provocado por el nuevo número de usuarios activos simultáneos en la celda (originado por el nuevo tamaño de la celda o por el cambio del usuario estudiado hacia una nueva celda servidora). Para aumentar la robustez del método, el gradiente se aproxima a través de la estimación del impacto que tiene un gran cambio de margen de traspaso en la adyacencia bajo estudio (concretamente de, 3 dB) en la QoE global del sistema, $\Delta\overline{QoE}_u$. Esta gran perturbación permite anticipar efectos que podrían no observarse con cambios más pequeños (p. ej., de 1 dB).

La Figura 3.16 muestra un diagrama de flujo del algoritmo de ajuste completo, que incluye el modelo analítico para estimar el impacto potencial del reparto de tráfico en la QoE media de usuario en cada adyacencia. Para ello, se estima primero el impacto individual a nivel de conexión, que después se agrega a nivel de celda. Las entradas del modelo son: a) las trazas de usuario, que contienen medidas de rendimiento a nivel de conexión (p. ej., caudal de datos medio), b) las trazas de

celda, que contienen medidas periódicas de rendimiento a nivel de celda (p. ej., carga de celda cada hora), c) estadísticos de nivel de señal, que reflejan el nivel de las señales de referencia de la celda servidora y sus celdas vecinas cada 10 TTIs (*Time Transmission Interval*) (10 ms), y d) la ubicación geográfica de las estaciones base, a partir de las que se calcula la distancia entre emplazamientos (*Inter-Site Distance*, ISD). La salida del modelo es el impacto del cambio de márgenes en la QoE media del sistema, $\Delta\overline{QoE}^{(n+1)}(i,j)$.

En el dibujo, para mayor claridad, las variables tomadas directamente de medidas se representan en líneas grises discontinuas, para aislarlas de las variables estimadas, que se representan con líneas continuas negras. Del mismo modo, las fases que tratan estadísticos a nivel de celda tienen fondo blanco y las fases que tratan estadísticos a nivel de conexión tienen fondo gris. De aquí en adelante, por brevedad, k denota a cualquiera de las celdas origen y destino de la adyacencia, i y j. A continuación, se describe cada uno de los bloques del algoritmo reflejados en la figura.

Paso 1: Definición de conexiones en la zona de solapamiento

En este paso se calculan aquellas conexiones, o la porción de ellas, que se producen en la zona de solapamiento. Esta zona de solapamiento está comprendida entre el punto en que tiene lugar el traspaso entre las dos celdas de una adyacencia y el punto en que tendría lugar si el margen de traspaso, $HOM(i,j)$, se redujese en 3 dB.

El primer paso es estimar la cantidad de tráfico reubicado mediante el cambio de margen de traspaso, que determina el factor $\frac{\delta A(k)}{\delta HOM(i,j)}$ en (3.24). Para ello, los usuarios (conexiones) de las celdas i y j se clasifican en tres conjuntos, dependiendo de si cambian o no de celda servidora debido al reparto de tráfico. Por un lado, U_i y U_j denotan la parte de la conexión en la celda i y j que se mantiene servida por i y j tras el cambio de margen. Por otro lado, U_{ij} denota la parte de las conexiones que se reubicarían por el algoritmo de reparto de tráfico. Como en [114], U_{ij} se identifica de forma precisa a partir de las estadísticas de nivel de señal piloto recogidas de las estaciones base, como aquellos usuarios (o, de forma precisa, segmentos de su conexión) que cumplen que

$$P_{rx}(u,j,t) \geq P_{rx}(u,i,t) + HOM(i,j) - 3 \ , \ \ j \neq i \ , \tag{3.25}$$

donde $P_{rx}(u,i,t)$ es la potencia recibida de la señal de referencia (*Reference Signal Received Power*, RSRP) recibida por el usuario u de la celda i en el instante de tiempo t. Mediante la agregación del tiempo en que esos usuarios están en la zona de solapamiento entre celdas adyacentes, el método calcula el número medio de conexiones simultáneas eliminadas por el reparto de tráfico en periodos

activos de la celda i en la iteración n, $\overline{N_{us,zs}}^{(n)}(i)$. El número de conexiones eliminadas por el reparto de tráfico en periodos activos de la celda j se calcula del mismo modo, pero intercambiando los índices i y j .

Paso 2: Estimación de la carga de celda

En el siguiente paso, se estiman los cambios en la carga media de las celdas i y j debido al cambio de usuarios por la modificación del HOM. Las entradas de esta fase son los tres conjuntos de usuarios definidos en el paso anterior, $U_x(x \in \{i, j, ij\})$, la carga media de celda y la eficiencia espectral medidas antes de los cambios de margen de traspaso, $L^{(n)}(k)$ y $EE^{(n)}(k)$, y la distancia entre emplazamientos en la adyacencia, $ISD(i, j)$. La salida principal de esta fase son las nuevas cargas medias de celda tras el reparto de tráfico, $L^{(n+1)}(k)$, que se utilizará como entrada en los pasos 3 y 4.

Si los usuarios en la zona de solapamiento se traspasan desde la celda i hacia la celda j, la carga de la celda i disminuye y la de la celda j aumenta. Al mismo tiempo, la interferencia que recibe la celda i desde la celda j, aumenta, debido al aumento de carga de esta última. Por tanto, la eficiencia espectral de la celda i podría decrecer, provocando un aumento de la carga de celda que podría contrarrestar el efecto del alivio de congestión del reparto de tráfico. El efecto contrario se observa en la eficiencia espectral de la celda j. Para modelar ambos efectos, los cambios de carga se desglosan en dos componentes como

$$\Delta L^{(n)}(k) = \Delta L_{RT}^{(n)}(k) + \Delta L_{I}^{(n)}(k) \ , \ k \in \{i, j\} \ , \tag{3.26}$$

donde $\Delta L_{RT}^{(n)}(k)$ refleja el cambio debido al distinto número de usuarios provocado por el traspaso de usuarios de la celda i a la celda j, y $\Delta L_{I}^{(n)}(k)$ refleja el cambio por la diferente calidad de la señal (eficiencia espectral) causada por las nuevas condiciones de interferencia.

Específicamente, el cambio de carga en la celda origen i debido al reparto de tráfico se calcula como

$$\Delta L_{RT}^{(n)}(i) = -\sum_{u_{ij}} \Delta L^{(n)}(u_{ij}, i) \ , \tag{3.27}$$

donde $L^{(n)}(u_{ij}, i)$ es la parte de la carga de la celda i eliminada mediante la reubicación del usuario

u_{ij} en la celda j, deducida del tiempo total de conexión en la zona de solapamiento observado en las medidas de nivel de señal. A partir de este cambio de carga en la celda i, se estima el cambio de carga en la celda destino j reescalando la carga eliminada de la celda origen, considerando la eficiencia espectral en ambas celdas como

$$\Delta L_{RT}^{(n)}(j) = \Delta L_{RT}^{(n)}(i) \frac{EE^{(n)}(i)}{EE^{(n)}(j)} \ . \tag{3.28}$$

Por otra parte, los cambios de carga por cambios de interferencia se estiman dependiendo de la distancia entre emplazamientos. De esta forma, se intenta diferenciar entre escenarios limitados por interferencia y limitados por ruido. En la celda origen, el cambio de carga por interferencia se estima como

$$\Delta L_{I}^{(n)}(i) = \begin{cases} L^{(n)}(i) \left[\frac{EE^{(n)}(i)}{EE^{(n+1)}(i)} - 1 \right] & \text{si } ISD(i,j) \leq 1.25 \text{ km} \ , \\ 0 & \text{en otro caso} \ , \end{cases} \tag{3.29}$$

donde $EE^{(n)}(i)$ es la eficiencia espectral media de la celda i medida en el lazo de optimización n, y $EE^{(n+1)}(i)$ es la eficiencia espectral media de la celda i en el lazo de optimización $n+1$. En esta última eficiencia, por simplicidad, se supone que la interferencia es mucho mayor que el ruido (escenario limitado por interferencia) y que toda la interferencia recibida en la celda i proviene de la celda j (único interferente), con lo cual

$$EE^{(n+1)}(i) = EE^{(n)}(i) \frac{L^{(n)}(j)}{L^{(n)}(j) + \Delta L_{RT}^{(n)}(j)} \ . \tag{3.30}$$

De forma similar, el cambio de carga por interferencia en la celda destino se estima como

$$\Delta L_{I}^{(n)}(j) = \begin{cases} L^{(n)}(j) \left[\frac{EE^{(n)}(j)}{EE^{(n+1)}(j)} - 1 \right] & \text{si } ISD(i,j) \leq 1.25 \text{ km} \ , \\ 0 & \text{en otro caso} \ , \end{cases} \tag{3.31}$$

donde

$$EE^{(n+1)}(j) = EE^{(n)}(j) \frac{L^{(n)}(i)}{L^{(n)}(i) + \Delta L_{RT}^{(n)}(i)} \ . \tag{3.32}$$

Es importante resaltar que la suposición de un único interferente en (3.29) y (3.31) es el caso peor de los escenarios limitados por interferencia en los que el reparto de tráfico consigue el mínimo efecto de alivio de congestión, tratándose por tanto de una estimación conservadora. Por el contrario, la suposición de que la interferencia es mucho mayor que el ruido es un caso de los escenarios limitados por interferencia que facilita el alivio de congestión de los algoritmos de reparto de tráfico. Esta suposición de escenario limitado por interferencia, sin embargo, se justifica porque se usa únicamente si la distancia entre celdas adyacentes es menor que 1.25 km. Los resultados que se presentan más adelante muestran que dicha aproximación tiene un impacto despreciable en el rendimiento del método.

Paso 3: Estimación de la eficiencia espectral

A continuación, se estiman los cambios de eficiencia espectral por usuario, que determina el factor $\frac{\delta EE(u)}{\delta SINR(u)}$ en (3.24). Como información de entrada, se emplea la relación señal e interferencia más ruido por usuario (en unidades naturales), $sinr^{(n)}(u)$, la distancia entre los emplazamientos de las celdas i y j, $ISD(i,j)$, las cargas medias de celda antes de los cambios, $L^{(n)}(k)$, y las estimas de los cambios de carga por reparto de tráfico e interferencia en ambas celdas, $\Delta L_{RT}^{(n)}(k)$ y $\Delta L_{I}^{(n)}(k)$. La salida es la eficiencia espectral para el siguiente lazo de optimización estimada por usuario, $EE^{(n+1)}(u)$. Esta estimación se lleva a cabo solamente si la distancia de la celda i a la celda j es menor que 1.25 km. Bajo estas circunstancias, de acuerdo con un análisis realizado en el escenario considerado, se puede suponer que la interferencia I recibida de la celda adyacente j es mucho mayor que el ruido, con lo que $sinr^{(n)}(u) \simeq \frac{c}{i}^{(n)}(u)$, $\forall u \in \{U_i, U_j, U_{ij}\}$. En caso contrario, la eficiencia espectral permanece invariante. Específicamente, para los usuarios servidos por la celda i, la eficiencia espectral se estima como

$$EE^{(n+1)}(u_i) \simeq \begin{cases} \log_2(1 + \frac{c}{i}^{(n)}(u_i)\frac{L^{(n)}(j)}{L^{(n+1)}(j)}) & \text{si } ISD(i,j) \leq 1.25 \text{ km} , \\ EE^{(n)}(u_i) & \text{en otro caso} , \end{cases} \tag{3.33}$$

donde el factor $\frac{L^{(n+1)}(j)}{L^{(n)}(j)}$ refleja el aumento de la interferencia debido al aumento de carga de la celda vecina. La estimación de la carga de celda para el siguiente lazo de optimización, $L^{(n+1)}(k)$, puede calcularse fácilmente a partir de las estimas de cambio de carga de celda como

$$L^{(n+1)}(k) = L^{(n)}(k) + \Delta L_{RT}^{(n)}(k) + \Delta L_{I}^{(n)}(k). \tag{3.34}$$

De forma similar, para los usuarios servidos por la celda j, la eficiencia espectral se estima como

$$EE^{(n+1)}(u_j) \simeq \begin{cases} \log_2(1+\frac{c}{i}^{(n)}(u_j)\frac{L^{(n)}(i)}{L^{(n+1)}(i)})) & \text{si } ISD(i,j) \leq 1.25 \text{ km} , \\ EE^{(n)}(u_j) & \text{en otro caso} . \end{cases} \tag{3.35}$$

Finalmente, para los usuarios en la zona de solapamiento, la eficiencia espectral se estima como

$$EE^{(n+1)}(u_{ij}) \simeq \begin{cases} EE^{(n)}(u_{ij}(j)) & \text{si } u_{ij}(j) \neq 0 , \\ \overline{EE^{(n)}(u_{ij}(j))} & \forall\ u_{ij}(j) \neq 0 \text{ en otro caso} , \end{cases} \tag{3.36}$$

donde los usuarios en la zona de solapamiento, u_{ij}, se dividen en dos grupos: aquellos que efectuaron traspaso desde la celda i hacia la celda j en el lazo de optimización n, $u_{ij}(j) \neq 0$, y aquellos que no realizaron traspaso entre ambas celdas $u_{ij}(j) = 0$.

Paso 4: Estimación de usuarios activos

El cuarto paso aborda la estimación del número de usuarios simultáneos en el siguiente lazo de optimización, que determina el factor $\frac{\delta N_{su}(k)}{\delta A(k)}$ en (3.24). Un análisis preliminar (no presentado aquí) muestra que esta variable ha de estimarse por usuario para obtener estimas fiables de *throughput* y QoE de usuario. La entrada a esta fase son las estimas de cambio de carga de celda, junto con el número medio de usuarios simultáneos en la zona de solapamiento durante los periodos activos, $\overline{N_{us,zs}}^{(n)}(i)$, el tiempo que pasó el usuario en la celda $k = \{i, j\}$, $t_k^{(n)}(u)$, la duración total de la conexión, $t_{tot}^{(n)}(u)$, la tasa de actividad de la celda i (medida como la tasa de TTIs activos), $\rho(i)$, y el número medio de usuarios activos simultáneos cuando cada usuario está transmitiendo en el lazo de optimización actual, $\overline{N_{us}}^{(n)}(u)$. La salida es el número de usuarios activos con cada usuario en el siguiente lazo de optimización, $\overline{N_{us}}^{(n+1)}(u)$.

El número esperado de usuarios activos simultáneos con un usuario u en la celda origen i en el siguiente lazo de optimización se calcula a partir del valor en el lazo de optimización anterior como

$$\overline{N_{us}}^{(n+1)}(u_i) = \overline{N_{us}}^{(n)}(u_i) - \overline{N_{us,zs}}^{(n)}(i) + \overline{\Delta N_{us,I}}^{(n)}(i) , \tag{3.37}$$

donde $\overline{N_{us,zs}}^{(n)}(i)$ captura la disminución del número de usuarios activos debido al alivio de conges-

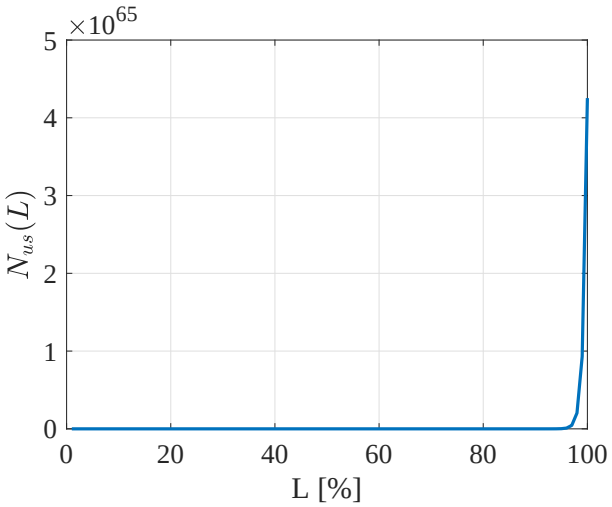


Figura 3.17: Número de usuarios simultáneos con la carga de celda.

tión por traspasar usuarios de la zona de solapamiento, y $\overline{\Delta N_{us,I}}^{(n)}(i)$ refleja el aumento del número de usuarios activos por la pérdida de eficiencia espectral por una mayor interferencia. Es importante hacer notar que, por definición, estas últimas cantidades se miden considerando únicamente los periodos de actividad de la celda (es decir, $\overline{N_{us}}^{(n)}(u_i) \geq 1$). Por la misma razón, $\overline{N_{us,zs}}^{(n)}(i)$ y $\overline{N_{us,I}}^{(n)}(i)$ se miden considerando solamente los periodos de actividad de la celda i.

Ante la dificultad de encontrar una expresión cerrada que relacione la carga de celda con el número de usuarios activos, se lleva a cabo un análisis de regresión con estadísticas de rendimiento para encontrar la expresión que relacione la carga de celda L con el número de usuarios activos N_{us} en la celda, $N_{us}(L)$. En la práctica, este análisis empírico se puede realizar de forma sencilla a partir de contadores agregados a nivel de celda que recoge el sistema de gestión de la red. En este trabajo, dicho análisis se lleva a cabo mediante simulaciones en las que las distintas celdas tienen cargas diferentes con distinto número de usuarios simultáneos. La curva de regresión que resulta es la función exponencial

$$N_{us}(L) = 1.851 \; e^{\,1.505 \; L} \; , \tag{3.38}$$

representada en la Figura 3.17.

Derivando la función exponencial anterior, se obtiene la sensibilidad del número de usuarios activos a los incrementos de la carga de celda, como

$$\frac{\delta N_{us}}{\delta L}(L) = 1.851 \cdot 1.505 \; e^{\; 1.505 \; L} \; . \tag{3.39}$$

Esta última expresión sirve para cuantificar el aumento del número de usuarios activos debido a la interferencia que supone el aumento de la carga de la celda por el reparto de tráfico, $\Delta L_{RT}^{(n)}(i)$, y la interferencia, $\Delta L_{I}^{(n)}(i)$, como

$$\overline{\Delta N_{us,I}}^{(n)}(i) = \Delta L_{I}^{(n)}(i) \; \frac{\Delta N_{us}}{\Delta L} \; (L^{(n)}(i) + \Delta L_{RT}^{(n)}(i)) \; . \tag{3.40}$$

Los cambios del número de usuarios activos en la celda origen i debido al reparto de tráfico (es decir, usuarios activos en la zona de solapamiento) pueden tomarse directamente de medidas de la red. Por el contrario, los cambios en la celda destino j han de estimarse. Para ello, se ha de tener en cuenta que los usuarios traspasados podrían requerir una cantidad diferente de recursos en la nueva celda, como resultado de las nuevas condiciones del enlace radio. Este hecho puede modelarse fácilmente multiplicando el número de usuarios activos en la zona de solapamiento por un factor. Este factor es el producto del número medio de usuarios activos simultáneos cuando cada usuario está transmitiendo en el lazo de optimización actual, $\rho(i)$, y la relación entre la eficiencia espectral media de la celda origen y la destino, $\frac{EE^{(n)}(i)}{EE^{(n)}(j)}$. Para tener en cuenta este efecto, el nuevo número medio de usuarios activos en la celda destino j se estima como

$$\overline{N_{us}}^{(n+1)}(u_j) = \overline{N_{us}}^{(n)}(u_j) - \overline{N_{us,zs}}^{(n)}(i) \; \rho(i) \; \frac{EE^{(n)}(i)}{EE^{(n)}(j)} + \overline{\Delta N_{us,I}}^{(n)}(j) \; , \tag{3.41}$$

siguiendo la misma estructura que en (3.37). Igual que en (3.40), $\overline{\Delta N_{us,I}}^{(n)}(j)$ se estima como

$$\overline{\Delta N_{us,I}}^{(n)}(j) = \Delta L_{I}^{(n)}(j) \; \frac{\Delta N_{us}}{\Delta L} \; (L^{(n)}(j) + \Delta L_{RT}^{(n)}(j)) \; , \tag{3.42}$$

usando la misma función derivada que en (3.39). Finalmente, para los usuarios en la zona de solapamiento u_{ij} que ya efectuaron traspaso desde la celda i hacia la celda j antes del reparto de tráfico, el número de usuarios activos en el lazo de optimización $n+1$ se estima como una media ponderada del número de usuarios simultáneos medido durante los segmentos de las conexiones en la celda i y j en el lazo anterior, u_i y u_j, ponderado por el tiempo en cada celda i y j , como

$$\overline{N_{us}}^{(n+1)}(u_{ij}) = \frac{t_i^{(n)}(u_{ij})}{t_{tot}^{(n)}(u_{ij})}\overline{N_{us}}^{(n+1)}(u_i) + \frac{t_j^{(n)}(u_{ij})}{t_{tot}^{(n)}(u_{ij})}\overline{N_{us}}^{(n+1)}(u_j)\ , \tag{3.43}$$

donde $t_i(u_{ij})$ y $t_j(u_{ij})$ es el tiempo que el usuario u_{ij} permanece en las celdas i y j, respectivamente, y $t_{tot}(u_{ij})$ es el tiempo total en que el usuario u es servido por ambas celdas.

Paso 5: Estimación del caudal de datos y QoE

Una vez estimado el número de usuarios activos para cada usuario en el siguiente lazo de optimización, $\overline{N_{us}}^{(n+1)}(u)$, así como la nueva eficiencia espectral, $EE^{(n+1)}(u)$, en este paso se estima el cambio de la QoE de usuario, $\Delta QoE^{(n+1)}(u)$. Para ello, primero se estiman las variaciones de *throughput* de usuario (que determinan los factores $\frac{\delta T(u)}{\delta EE(u)}$ y $\frac{\delta T(u)}{\delta BW(u)}$ en (3.24)), y después los cambios de QoE por usuario (que determinan el factor $\frac{\delta QoE(u)}{\delta T(u)}$ en (3.24)).

La estimación del *throughput* de usuario tras los cambios de margen se calcula a partir de la eficiencia espectral como

$$TH^{(n+1)}(u) = EE^{(n+1)}(u)BW^{(n+1)}(u) = EE^{(n+1)}(u)\ \frac{N_{PRB}}{\overline{N_{us}}^{(n+1)}(u)},\quad \forall u \in U_i, U_j, U_{ij} \tag{3.44}$$

donde N_{PRB} es el ancho de banda del sistema y $EE^{(n+1)}(u)$ es la eficiencia espectral del usuario, que caracteriza el *throughput* por PRB.

Los valores de *throughput* obtenidos con (3.44) pueden traducirse fácilmente a valores de QoE con (3.22)-(3.23). Entonces, el cambio de la QoE de usuario por los cambios de HOM se estima como

$$\Delta QoE^{(n+1)}(u) = QoE^{(n+1)}(u) - QoE^{(n)}(u),\quad \forall u \in U_i, U_j, U_{ij}\ . \tag{3.45}$$

Paso 6: Agregación de la QoE a nivel de celda

La variación de la QoE media de la red provocada por la modificación de $HOM(i,j)$ se calcula como

$$\Delta QoE^{(n+1)}(i,j) = \frac{\sum_{u \in U_i, U_j, U_{ij}} \Delta QoE^{(n+1)}(u)}{N_u(i) + N_u(j)} , \tag{3.46}$$

que refleja la agregación de los cambios de QoE individuales dividida por el número de usuarios en las celdas i y j, que se mantiene en los lazos n y $n+1$.

Todo el análisis anterior desarrollado en 6 pasos considera el caso en que se decrementa $HOM(i,j)$, facilitando el traspaso de usuarios de la celda origen a la celda destino de la adyacencia. El caso contrario, cuando se incrementa $HOM(i,j)$, puede evaluarse analizando la adyacencia en sentido opuesto, en la que $HOM(j,i)$ se decrementa para satisfacer (3.2). Todo el proceso de estimación explicado ha de repetirse para considerar un decremento similar de $HOM(j,i)$ de 3 dB, obteniendo una estimación de $\Delta QoE^{(n+1)}(j,i)$. Como resultado, se obtienen dos estimaciones de cambios de QoE media por adyacencia: $\Delta QoE^{(n+1)}(i,j)$ y $\Delta QoE^{(n+1)}(j,i)$, correspondientes a ambos movimientos de HOM, que consiguen una reducción del área de servicio de la celda i o la j, respectivamente. El algoritmo OE elige aquel cambio de HOM que obtiene el mayor valor de ΔQoE_u (es decir, la mayor ganancia, bien sea por mover tráfico desde la celda i hacia la j o viceversa). Si ambos movimientos degradan la QoE global de la adyacencia, no se implementa cambio de HOM alguno, siguiendo la estrategia de ascenso de gradiente (máximo local del problema).

Paso 7: Controlador no lineal

Para controlar la magnitud de los cambios de HOM, se implementa un controlador incremental, mostrado en la Figura 3.18. Se observa que el controlador modifica la ganancia del lazo de realimentación, controlando el compromiso entre velocidad de convergencia y estabilidad del sistema. Como se aprecia en la figura, como medida de estabilidad, no se implementa ningún cambio cuando los beneficios esperados de QoE están por debajo de 0.01. Así, el sistema de control alcanza antes el régimen permanente. Más allá de dicho valor, se usa una pendiente mayor para favorecer a las adyacencias de las que se espera obtener mayores beneficios de QoE. Para evitar inestabilidades, el cambio máximo de HOM por iteración se limita a 3 dB.

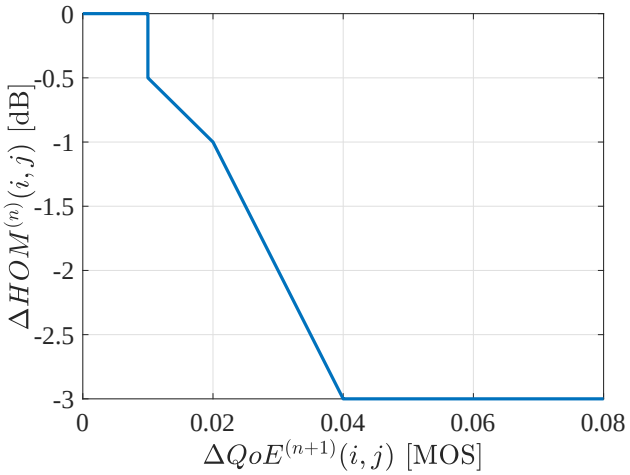


Figura 3.18: Controlador no lineal por adyacencia (i, j).

La Figura 3.18 muestra el caso en que $\Delta QoE^{(n+1)}(i,j) > 0$ y $\Delta QoE^{(n+1)}(i,j) > \Delta QoE^{(n+1)}(j,i)$ (es decir, reducir $HOM(i,j)$ es más beneficioso que reducir $HOM(j,i)$). En este caso, el controlador calcula primero el cambio de HOM en la adyacencia (i,j), $\Delta HOM^{(n)}(i,j)$, de forma que $HOM^{(n+1)}(i,j) = HOM^{(n)}(i,j) + \Delta HOM^{(n)}(i,j)$. Después, para mantener la histéresis, se calcula $HOM^{(n+1)}(j,i) = HOM^{(n)}(j,i) - \Delta HOM^{(n)}(i,j)$. En el escenario contrario, donde reducir $HOM(j,i)$ es más beneficioso que reducir $HOM(i,j)$, el controlador funciona del mismo modo simplemente intercambiando los índices i y j.

3.5.2. Análisis del rendimiento

En este apartado, se valida el algoritmo de optimización propuesto mediante simulaciones. Para mayor claridad, se presenta primero la metodología experimental y después los resultados de simulación. Por último, se tratan algunas consideraciones de implementación.

Metodología experimental

El escenario simulado es el mismo que se utilizó para validar el algoritmo EB en 3.4.1, mostrado en la Figura 3.7, cuyos parámetros se presentaron en la Tabla 3.2.

El modelo de tráfico se restringe al del servicio FTP, recogido en la Tabla 2.1. Dicho servicio modela la descarga de un fichero cuyo tamaño sigue una distribución lognormal de media 2 MB.

Se ha elegido el servicio FTP por ser representativo de un servicio de búfer completo (*full buffer*) durante el tiempo que dura la conexión.

La herramienta de simulación incluye dos tipos de usuarios: de interior y de exterior. Los usuarios de interior son estáticos y más exigentes con relación a la QoE, incluso cuando sus pérdidas de propagación son 15 dB mayores que las que experimentan los usuarios de exterior, debido a los muros de separación con el exterior. Los usuarios de exterior se mueven a 3 km/h siguiendo un camino rectilíneo. El porcentaje de usuarios de interior por celda depende de la ubicación de la celda. En este caso concreto, el 36 % de las celdas del escenario se categorizan como urbanas, el 46 % como suburbanas y el 18 % como rurales, en base al uso de suelo predominante en sus áreas de servicio. Posteriormente, se supone que las celdas urbanas tienen un 70 % de usuarios de interior, las suburbanas un 50 % , y las rurales un 10 % [113]. La demanda de tráfico global en el escenario se controla ajustando la tasa media total de llegada de usuarios de forma que se generen problemas de sobrecarga (y pérdida de QoE). La distribución espacial de tráfico por celda sigue el mismo perfil que la mostrada en la red real, deducida de estadísticas suministradas por el operador. Como punto inicial de trabajo, con la configuración de HOM por defecto, la carga media de celda de la red es de $\overline{U(i)} = 61\,\%$. No obstante, las cargas de celda mínima y máxima en el escenario son de 3.5 % y 100 %, respectivamente, evidenciando que la demanda de tráfico se distribuye de forma irregular, al igual que ocurría en el escenario considerado en 3.4.1 para validar el algoritmo de balance de QoE.

Para evaluar el rendimiento del algoritmo OE se compara su rendimiento con el de 3 métodos iterativos de ajuste de parámetros, ya presentados en la sección 3.3. Los tres métodos son algoritmos de balance que persiguen igualar algún indicador entre celdas vecinas mediante el ajuste de HOM por adyacencia. Estos métodos son el algoritmo de balance de carga (*Load Balancing*, LB), el algoritmo de balance de caudal (*Throughput Balancing*, TB) y el algoritmo de balance de QoE (EB-C), que igualan la utilización media de celda, el *throughput* medio de usuario por celda y la QoE media de celda, respectivamente. La modificación de los márgenes en LB, TB y EB-C se lleva a cabo mediante controladores de lógica difusa, que implementan sencillas reglas de control «SI-ENTONCES». Para todos los algoritmos, incluido OE, se simulan 14 lazos de optimización de 30 minutos de tiempo de red. Para evaluar los métodos, la principal cifra de mérito es la QoE global del sistema, $\overline{QoE}_u$, definida en (3.20). Para un análisis más detallado, se calcula también en cada iteración la desviación media de HOM respecto la configuración por defecto, según (3.19).

Al igual que en las pruebas del algoritmo EB-C, con la intención de verificar la capacidad de los métodos para equilibrar cada indicador particular, se usarán los indicadores de desequilibrio de carga, *throughput* y QoE media definidos en (3.10), (3.15) y (3.11).

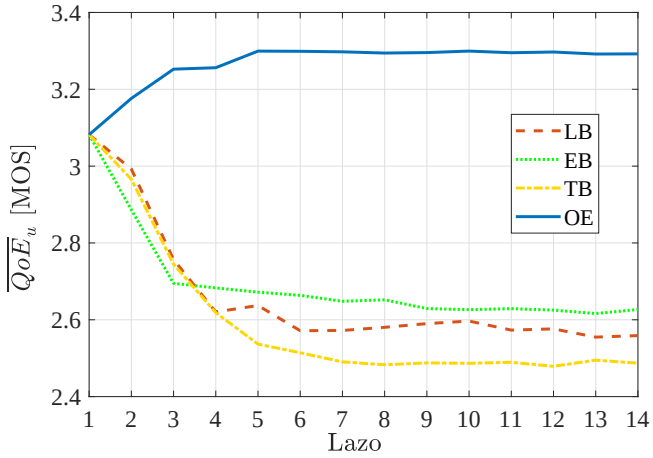


Figura 3.19: Evolución de la calidad de experiencia media en el escenario.

Resultados

La Figura 3.19 muestra la evolución de la QoE global del sistema con los 4 enfoques considerados. En la figura, se observa que $\overline{QoE}_u$ con OE tiene una tendencia claramente distinta a la que muestran los otros tres algoritmos de equilibrio, LB, TB y EB-C. En el estado inicial, con un valor por defecto de HOM de 3 dB en todas las adyacencias, $\overline{QoE}_u = 3.08$. Al final del proceso de optimización, OE alcanza un valor de $\overline{QoE}_u = 3.3$, mientras que LB, TB y EB-C decrecen hasta los valores de 2.56, 2.48 y 2.63, respectivamente. Así, OE supera al segundo mejor algoritmo (EB-C) en 0.6 puntos de MOS. Lo que es más importante, OE consigue aumentar $\overline{QoE}_u$ en más de 0.2 puntos de MOS en comparación con el estado inicial (3.3 versus 3.08), mientras que los demás métodos, degradan $\overline{QoE}_u$ en 0.52, 0.6 y 0.43 puntos de MOS, respectivamente. Como era de esperar, OE alcanza la mejor cifra de $\overline{QoE}_u$ debido a su formulación analítica, que considera de forma explícita aspectos como el nivel de señal, la interferencia, la eficiencia espectral o el reparto de recursos, adoptando un criterio de optimalidad.

Igualmente importante es que la mejora de la QoE global de usuario no se consigue a costa de degradar la QoE de los peores usuarios. Para confirmar esta afirmación, la Figura 3.20 compara la función de distribución de la QoE de usuario obtenida por todos los métodos tomando como referencia la obtenida con la configuración por defecto (curva *Inicial*). En la situación inicial, un porcentaje significativo de usuarios (35 %) tienen el mínimo valor posible de QoE ($QoE(u) = 1$), incluso cuando muchos de ellos (40 %) tienen el máximo valor posible de QoE ($QoE(u) = 5$). Este desequilibrio se debe a la elevada tasa de llegada de conexiones y la distribución espacial irregular

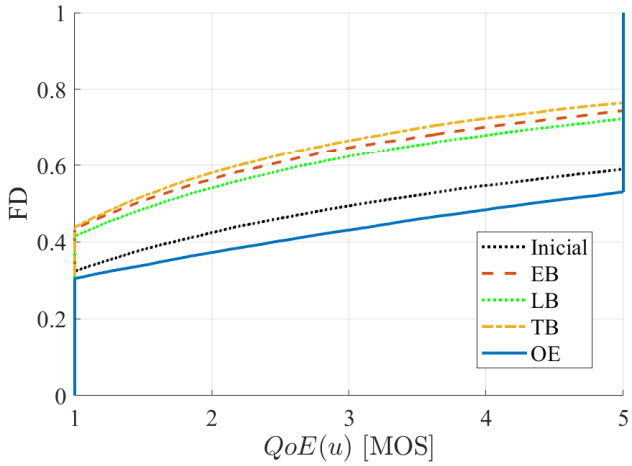


Figura 3.20: Función de distribución de la calidad de experiencia por usuario.

de los usuarios en el escenario. De forma inesperada, se observa que los algoritmos de balance diseñados para equilibrar el rendimiento a nivel de celda aumentan el porcentaje de usuarios con baja QoE. Un análisis más detallado (no presentado aquí) muestra que LB y TB intentan equilibrar indicadores de tráfico sin tener en cuenta la mezcla de servicios. Del mismo modo, EB-C no mejora la QoE media de usuario, porque al equilibrar la QoE media de celda $QoE(i)$ mejora los peores usuarios de las peores celdas a costa de degradar la experiencia de los mejores usuarios en las celdas vecinas. Por el contrario, OE reduce el número de usuarios completamente insatisfechos desde el 35 hasta el 30 %, al mismo tiempo que se aumenta el número de usuarios completamente satisfechos desde el 41 hasta el 42 %. Estas diferencias se mantienen en más percentiles de la distribución. Por ejemplo, el percentil 35 de $QoE(u)$ es 1.22 en la configuración inicial, 1 con LB, TB y EB-C, y 1.65 con OE.

La Tabla 3.8 muestra todos los indicadores de rendimiento para el estado inicial y final del proceso de optimización y para los algoritmos considerados (columnas LB, TB, EB-C y OE). Para mayor claridad, se resalta en gris el mejor algoritmo para cada indicador. Como podía esperarse, $\overline{U_{imb}(i)}$, $\overline{T_{imb}}$ y $\overline{QoE}_u$ alcanzan su mejor rendimiento con los algoritmos para los que fueron diseñados (LB, TB y OE, respectivamente). En particular, LB alcanza el menor desequilibrio de carga media de celda, es decir, $\overline{U_{imb}} = 7.43\,\%$ y TB alcanza el menor desequilibrio de *throughput* medio de celda ($\overline{T_{imb}} = 0.22$ Mbps).

No obstante, e inesperadamente, el mínimo valor de $\overline{QoE_{imb,c}}$ se alcanzado con TB ($\overline{QoE_{imb,c}} = 0.3$), y no con EB-C ($\overline{QoE_{imb,c}} = 0.53$), a costa de una mayor degradación de $\overline{QoE}$ (2.64 con TB frente a 2.88 con EB-C). Este comportamiento inesperado se debe a los límites de las funciones de

Tabla 3.8: Principales indicadores de rendimiento.

	Inicial	**LB**	**TB**	**EB-C**	**OE**
$\overline{U_{imb}}$ [%]	18.84	7.43	10.3	11.22	15.14
$\overline{T_{imb}}$ [$Mbps$]	0.90	0.36	0.22	0.36	0.85
$\overline{QoE_{imb,c}}$	0.71	0.55	0.3	0.53	0.71
$\overline{QoE}_u$	3.08	2.56	2.48	2.63	3.3
$\overline{QoE}$	3.73	2.62	2.64	2.88	3.84

utilidad (3.22)-(3.23), que alcanzan niveles de saturación (QoE = 1 ó 5) con valores muy diferentes de $T(u)$ (0.237 y 0.853 Mbps, respectivamente, para usuarios de exterior). Valores muy diferentes de caudal en las celdas vecinas pueden dar lugar a valores muy parecidos de QoE media de celda. Por ejemplo, dos celdas con $T(i) = 0.05$ Mbps y $T(j) = 0.26$ Mbps de media experimentarían $QoE(i) = 1$ y $QoE(j) = 1.1$, respectivamente. Bajo estas circunstancias, mientras que existe una diferencia insignificante de QoE entre ambas celdas, hay una gran diferencia en sus correspondientes caudales medios de celda. Por lo tanto, TB prosigue con sus acciones de ajuste degradando $QoE(j)$, mientras que continúa mejorando $\overline{QoE_{imb,c}}$, llegando a un valor de 0.3 puntos de MOS. De este modo, el valor de $\overline{QoE}_u$ conseguido por EB-C es 0.24 puntos de MOS mayor que el conseguido por TB. En una situación real, donde se mezclan diferentes servicios con distintas funciones de utilidad para calcular la QoE media de celda, la relación entre *throughput* y QoE es mucho más compleja que en el escenario considerado, provocando que ambos objetivos (balancear $T(i)$ y $QoE(i)$) generasen posiblemente trayectorias de optimización diferentes [115].

La Figura 3.21 ilustra la evolución de la desviación de HOM desde los valores por defecto en los cuatro algoritmos de autoajuste. Como se observa en la figura, los cambios introducidos por OE son menores que los provocados por los algoritmos de balance ($\delta\overline{HOM} = 4.8$ dB para OE al final del proceso de optimización, mientras que para LB, TB y EB-C se alcanzan 6, 7.3 y 6.4 dB, respectivamente). Por tanto, OE alcanza un mejor rendimiento de red con menores cambios de parámetros. Una menor intervención sobre la red es una ventaja muy importante del algoritmo OE, ya que los operadores suelen ser reacios a cambiar mucho los valores por defecto de los parámetros de la red. La eficiencia de OE viene directamente de su formulación analítica, con la que se modifican los valores de HOM en la cantidad exacta para aumentar de $\overline{QoE}_u$. Por el contrario, LB, TB y EB-C continúan aumentando el área de servicio de las celdas infrautilizadas, incluso aunque ello suponga la degradación de la QoE global del sistema.

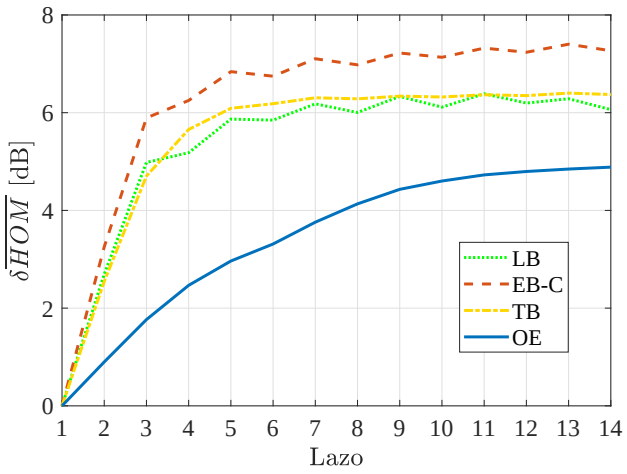


Figura 3.21: Evolución de los valores de HOM.

Consideraciones de implementación

El algoritmo OE se ejecuta por adyacencia. Por lo tanto, su complejidad temporal asintótica es $\mathcal{O}(N_{adjs})$. Para el escenario considerado, que consiste en 108 celdas y 11664 adyacencias, el tiempo medio de ejecución de 1 iteración de OE es de 4.35 minutos (22 ms por adyacencia) en un ordenador personal con un procesador de ocho núcleos con una frecuencia de reloj de 3.6 GHz y 24 GB de RAM. El tiempo de ejecución se puede reducir aún más restringiendo los cambios únicamente a las celdas vecinas más próximas.

3.6. Conclusiones

En este capítulo, se han propuesto varios algoritmos de ajuste automático de los márgenes de traspaso en una red LTE. A diferencia de los algoritmos clásicos, cuyo objetivo es equilibrar la carga entre celdas obviando la experiencia de los usuarios, el objetivo del primer algoritmo es equilibrar la QoE entre celdas y servicios. El algoritmo, que sigue un esquema iterativo, cambia los márgenes de traspaso entre celdas adyacentes para desplazar usuarios desde celdas con baja QoE hacia celdas vecinas con QoE más altas. Como primera contribución original de este capítulo, se han propuesto dos variantes de este primer algoritmo, dependiendo de si los márgenes de traspaso se modifican por adyacencia, o por adyacencia y servicio. La evaluación del algoritmo se ha realizado en un simulador dinámico de nivel de sistema que implementa un escenario macrocelular realista. Los

resultados muestran que el desequilibrio de la QoE media de las celdas se reduce tras el ajuste de los parámetros. Específicamente, la diferencia de QoE media entre celdas se reduce en 0.22 y 0.26 puntos de MOS con EB-C y EB-CS, respectivamente, con una desviación media de HOM entre 0 y 6 dB dependiendo del servicio.

De forma alternativa, como segunda contribución de este capítulo, se ha propuesto un algoritmo para optimizar la QoE media de usuario en una red LTE modificando los márgenes de traspaso. El algoritmo se basa en la técnica de ascenso de gradiente para asegurar que los cambios de parámetros siempre mejoren la QoE global del sistema. Para ello, se estima el impacto de dichos cambios de parámetros en la QoE del sistema con un modelo analítico del rendimiento de la red ajustado con estadísticos de la red real. Al igual que en el primer algoritmo propuesto, la evaluación del algoritmo se ha llevado a cabo en un simulador dinámico de nivel de sistema que implementa el mismo escenario, considerando un servicio de descarga de ficheros. Los resultados muestran que el algoritmo OE propuesto consigue aumentar la QoE media de usuario en la red en un 11.45 %, superando el rendimiento de los algoritmos clásicos de equilibrio. Igualmente importante es que OE alcanza un rendimiento óptimo con modificaciones menores del margen de traspaso (hasta en un 33 % menor que con otros algoritmos).

Los algoritmos EB y OE propuestos están concebidos como soluciones centralizadas para el sistema de gestión de la red, ya que las estadísticas de QoE que necesitan los algoritmos se obtienen actualmente mediante técnicas de inspección de paquetes en distintas interfaces de la red [116]. Los algoritmos iterativos subyacentes se han diseñado para ejecutarse tras cada periodo de informe de salida (p. ej., 1 hora). Esta ventana temporal asegura la obtención de medidas de QoE fiables para sesiones largas de transmisión de vídeo. Si se necesitan cambios de parámetros más rápidos, los algoritmos propuestos podrían ejecutarse con mayor frecuencia (p. ej., minutos), siempre que haya estimaciones de QoE fiables por conexión. Los futuros sistemas de comunicaciones móviles 5G contemplan proporcionar esta información a los sistemas de gestión automática basada en SON, como parte de la ingente cantidad de datos que manejan dichas plataformas.

Capítulo 4

Optimización del traspaso por movilidad basado en QoE

En este capítulo, se estudia la optimización del traspaso por movilidad (*Mobility Robustness Optimization*, MRO) basado en QoE para redes LTE. A diferencia de las técnicas presentadas en el Capítulo 3, cuyo objetivo es mejorar la QoE redistribuyendo el tráfico mediante el ajuste de márgenes de traspaso, en este capítulo se proponen técnicas para mejorar el mecanismo de traspaso en sí mismo para los usuarios que se desplazan, incluyendo consideraciones de calidad de experiencia.

Este capítulo se divide en cinco secciones. La Sección 4.1 revisa el estado de la técnica en MRO, prestando especial atención a los métodos que utilizan aprendizaje automático (*Machine Learning,* ML). La Sección 4.2 formula el problema de la regulación de parámetros de traspaso, explicando los eventos que pueden ocurrir por una mala configuración de los mismos y justificando la necesidad de considerar explícitamente la QoE. Tras presentar el problema, en la Sección 4.3 se describen dos variantes de un algoritmo MRO para redes LTE macrocelulares. Posteriormente, en la Sección 4.4 se describen los experimentos realizados para validar su rendimiento, y se plantean algunas consideraciones de implementación. Por último, en la Sección 4.5 se exponen las principales conclusiones de esta parte del trabajo.

4.1. Revisión del estado de la técnica

El procedimiento de traspaso (HO) es uno de los mecanismos más importantes en una red móvil. Gracias al traspaso, se asegura una conexión sin cortes a los usuarios que se mueven por la red. Sin embargo, si los parámetros de traspaso no están bien configurados, los traspasos se realizan demasiado pronto/tarde (*too early/late* HOs), provocando interrupciones del enlace radio (*Radio Link Failures*, RLF) o inestabilidades en forma de traspasos innecesarios (efecto Ping-Pong, PP). Para evitar estos efectos indeseados, surge la optimización del traspaso de movilidad (MRO). MRO es un caso de uso de autooptimización que ajusta los parámetros de traspaso para mejorar su rendimiento (es decir, minimizar PP y RLF) [117].

En trabajos anteriores anteriores, se ha probado que el rendimiento del traspaso puede mejorarse con técnicas muy diferentes. Un primer grupo de enfoques [118] [119] [120] [121] [122] emplean un modelo analítico para encontrar los valores óptimos de los parámetros de un algoritmo clásico de traspaso, formulando el problema de ajuste como un problema de optimización multiobjetivo en el que la función objetivo incluye el número de traspasos por llamada, la probabilidad de corte de la llamada, la eficiencia espectral de borde de celda o las tasas de interrupción de conexión. Un segundo grupo de técnicas, concebidas para redes ya desplegadas, usan métodos simples de regulación de parámetros de traspaso, que se disparan tras haber superado un umbral de fallos de traspaso [23] o de tasa de llamadas caídas [123]. Los cambios a realizar son propuestos por algoritmos de control iterativo, que basan sus decisiones en reglas heurísticas que reflejan el conocimiento de personas expertas. Dichas reglas pueden adaptarse a entornos de interior [123] o de exterior [23]. Este enfoque puede mejorarse ajustando los parámetros de traspaso de forma proactiva mediante predicciones de RLF [124]. Sin embargo, con reglas heurísticas, no se garantiza que se alcance la configuración óptima de los parámetros de traspaso. Un tercer conjunto de algoritmos rediseñan el mecanismo de traspaso por completo (en lugar de ajustar sus parámetros) procesando medidas instantáneas de señal para decidir cuándo disparar el traspaso para cada usuario individual [81] [28]. Estos mecanismos adaptativos de traspaso mejoran constantemente gracias al análisis de su comportamiento pasado, eliminando la necesidad de personas expertas y convirtiéndose en una poderosa herramienta de optimización de redes móviles. Su mayor inconveniente es que requieren actualizar el equipamiento de la red, mientras que los algoritmos analíticos y de autoajuste pueden seguir utilizando la infraestructura existente.

Para el diseño de mecanismos adaptativos de traspaso, se han utilizado técnicas de aprendizaje autónomo. En [81], se usa aprendizaje por refuerzo (concretamente, QL) para cambiar adaptativamente los parámetros de un algoritmo clásico de traspaso con el objetivo de reducir PP y las llamadas caídas. Del mismo modo, [29] usa QL para actualizar las reglas de un controlador de

lógica difusa con el objetivo de reducir el número de traspasos realizados sin éxito y la tasa de llamadas caídas, ajustando solamente los márgenes de traspaso y dejando fijo el temporizador de traspaso (*Time-To-Trigger*). De forma similar, en [28] se propone un esquema de traspaso adaptativo basado en QL para reducir RLF y PP incluyendo explícitamente ambos indicadores como entradas del proceso de aprendizaje. De esta forma, se pueden encontrar los mejores valores de los parámetros de traspaso por celda dependiendo de la velocidad de los usuarios en el área geográfica considerada. Otros esquemas de traspaso adaptativos más sofisticados se implementan con una red neuronal (*Artificial Neural Network*, ANN) [72] [73]. No obstante, incluso cuando todos estos algoritmos pueden mejorar la QoE potencialmente, hasta donde se conoce, ninguno de estos esquemas MRO basado en aprendizaje automático incluye criterios de QoE.

En este trabajo, se propone un nuevo algoritmo de MRO basado en técnicas de aprendizaje automático con criterios de QoE para redes LTE. A diferencia de estudios previos (p. ej., [28]), el algoritmo propuesto no sólo tiene como objetivo reducir el número de PP y RLF en la red, sino también aumentar la QoE de los usuarios de borde de celda que realizan traspaso. Como en otros trabajos, este objetivo se consigue desplazando el punto de disparo del traspaso modificando los parámetros de traspaso: el margen de traspaso, HOM, y el temporizador de traspaso (*Time-To-Trigger*), TTT. Para ello, se usa un esquema adaptativo de traspaso construido con QL y una ANN para ajustar los parámetros de traspaso por servicio a partir de estimas de la QoE de usuario.

El algoritmo resultante se valida en un simulador dinámico de nivel de sistema que implementa un escenario LTE macrocelular realista. Las principales contribuciones de este capítulo son: a) revelar las limitaciones de los algoritmos MRO tradicionales desde el punto de visto de la QoE, b) la inclusión de criterios de QoE en un esquema adaptativo de traspaso para aumentar la QoE de los usuarios de borde de celda y, al mismo tiempo, reducir RLF y PP, mediante la modificación de los márgenes y temporizadores del traspaso, y c) la validación del algoritmo mediante simulaciones en un escenario macrocelular LTE realista.

4.2. Formulación del problema

El mecanismo de traspaso asegura una conexión sin cortes a los usuarios que se mueven entre celdas vecinas. En su esquema más habitual, conocido comúnmente como traspaso por balance de potencia (*Power BudGeT*, PBGT), un traspaso se dispara en el instante de tiempo t_0 cuando se cumple la condición

$$P_{rx}(j) - P_{rx}(i) \geq HOM(i,j) \quad \forall\, t \in [t_0 - TTT(i,j), t_0] \ , \tag{4.1}$$

donde $P_{rx}(j)$ es la potencia recibida de la señal piloto de la ceda vecinas j, $P_{rx}(i)$ es la potencia recibida de la señal piloto de la celda servidora i, y $HOM(i,j)$ y $TTT(i,j)$ son el margen de HO y *Time-To-Trigger* de la celda i a la celda j. Ambos parámetros de traspaso, HOM y TTT, se definen por adyacencia.

Si los parámetros de traspaso no están bien configurados, puede producirse alguno de los siguientes eventos.

1. RLF por traspaso tardío (*Too Late HandOver*, LHO): tiene lugar cuando un usuario se mueve desde la celda origen i hacia la celda destino j, pero el traspaso se dispara demasiado tarde (o incluso no se dispara). En este caso, el nivel de la señal piloto recibida de la celda i cae por debajo de un determinado umbral durante un periodo específico de tiempo y la red determina que ha ocurrido un RLF en la celda i. Como resultado, el usuario se desconecta de la celda i y se reconecta a la celda j un tiempo después.

2. RLF por traspaso adelantado (*Too Early HandOver*, EHO): tiene lugar cuando un usuario se mueve desde la celda origen i hacia la celda destino j, pero el traspaso se dispara demasiado pronto. Después del traspaso, el nivel recibido de la señal piloto desde la celda j cae por debajo de un determinado umbral durante un periodo específico de tiempo y también ocurre un RLF. Como resultado, el usuario se desconecta de la celda j y se reconecta a la celda i (o a otra celda).

3. Ping-Pong (PP): cuando un usuario se mueve desde la celda origen i hacia la celda destino j, pero el traspaso se dispara pronto, el algoritmo de traspaso intenta traspasar el usuario de vuelta a la celda i dentro de una ventana temporal de duración determinada después de que el primer traspaso haya tenido lugar. Si finalmente se consigue, ocurren dos traspasos que podrían haberse ahorrado. Estos traspasos innecesarios provocan un aumento inútil de la carga de señalización en la red.

Los proveedores de red definen diferentes temporizadores y umbrales para clasificar los traspasos como LHO, EHO o PP [125]. En este trabajo, se supone que el equipo de usuario detecta un RLF cuando la potencia recibida de la señal de referencia (*Reference Signal Received Power*, RSRP) cae por debajo de -100 dBm durante 200 ms (temporizador T310 [125]). Del mismo modo, la ventana temporal para considerar dos traspasos consecutivos como PP se establece en 5000 ms (temporizador T311 [125]). LHO, EHO y PP son mutuamente excluyentes. Si se produce un traspaso y no ocurre

ninguno de los eventos explicados, se considera que el traspaso ha tenido éxito (*Successful HO*, SHO). En este trabajo sólo se consideran traspasos basados en el evento A3, en el que se detecta que una celda vecina pasa a ser mejor que la celda servidora [126].

Los algoritmos MRO modifican los valores de HOM y TTT para maximizar las tasas de SHO (o, complementariamente, minimizar LHO, EHO y PP). Idealmente, la configuración óptima debería asegurar que el traspaso ocurre en el punto en que los valores de RSRP recibidos de las celdas origen y destino son comparables y considerablemente buenos. Esta configuración se consigue con valores bajos de HOM y TTT. En la práctica, se suele forzar un valor de histéresis para evitar inestabilidades debidas a rápidas fluctuaciones de las condiciones de propagación. A la hora de elegir los valores de HOM y TTT, existe un compromiso entre LHO, EHO y PP. Cuanto más bajos sean HOM y TTT, más rápido se dispara el traspaso, asegurando que el usuario está siempre conectado a la mejor celda desde el punto de vista radio, pero es más probable que ocurra un EHO o un PP. Por el contrario, cuanto más altos sean HOM y TTT, más se retrasa el traspaso, evitando EHO y PP, pero el usuario permanece conectado a la celda origen recibiendo peor nivel de señal que el recibido por la celda destino, lo que puede degradar temporalmente la calidad del enlace radio, pudiendo terminar en un LHO.

Las degradaciones de señal experimentadas en el borde de celda por una mala configuración del traspaso pueden reducir el caudal de usuario y aumentar el retardo de paquetes, provocando un impacto negativo en la QoE de los usuarios. También es posible que esta degradación de caudal o retardo no se traslade, o no lo haga con toda la intensidad, a la QoE, y, por tanto, no sea percibida por el usuario, ya que no existe una relación directa entre el rendimiento del enlace radio y la QoE [115]. Por lo tanto, minimizar los fallos de traspaso y PP no conduce necesariamente a aumentar la QoE de los usuarios involucrados. Por la misma razón, la configuración de los parámetros de traspaso que consiguen el rendimiento óptimo del traspaso puede no llevar a la mejor QoE de los usuarios de borde de celda que son traspasados. Además, no todos los servicios se ven afectados del mismo modo cuando se degrada el rendimiento del enlace radio, como refleja la Figura 2.4. Esta diferencia de comportamiento recomienda una configuración distinta de los parámetros de traspaso para cada servicio.

Con estas hipótesis de partida, en este capítulo se propone un algoritmo capaz de aprender cuál es la configuración óptima de los parámetros de traspaso no solo para garantizar la robustez del enlace radio, sino también para maximizar la QoE de los usuarios antes y después del traspaso.

4.3. Algoritmo de MRO basado en QoE

En esta sección, se presenta un nuevo algoritmo de MRO basado en QoE para una red LTE. El objetivo del algoritmo es mejorar la QoE de los usuarios que experimentan el peor rendimiento del servicio, que son los usuarios de borde de celda. El algoritmo propuesto, denominado como MRO de Experiencia (*Experience MRO*, E-MRO), está inspirado en el algoritmo clásico de MRO descrito en [28] (de aquí en adelante, *Quality-MRO*, Q-MRO). Al igual que Q-MRO, la implementación de E-MRO está basada en un algoritmo QL que decide los mejores valores de los parámetros de traspaso HOM y TTT por adyacencia. Por claridad, se explica primero el mecanismo de aprendizaje por QL, para pasar después a detallar el algoritmo Q-MRO que se toma como referencia, y, por último, presentar el algoritmo E-MRO propuesto como evolución de Q-MRO.

4.3.1. Algoritmo Q-learning

Entre todas las técnicas de aprendizaje automático, se selecciona QL por su capacidad de aprender y mejorar el rendimiento de la red a través de la experiencia. QL es una técnica de aprendizaje por refuerzo, en la que el objetivo es aprender las reglas óptimas de funcionamiento de un sistema sin disponer de un modelo del entorno.

Un problema de QL se formula definiendo el trío (X, A, r), donde, X es el conjunto de estados del sistema, A es el conjunto de posibles acciones de control, y $r : X \cdot A \rightarrow R$ es la función de recompensa, que representa la mejora del rendimiento obtenida al ejecutar una acción en un estado. Un problema QL también involucra necesariamente a un agente, que puede definirse como un ente con características que usualmente se asocian con los seres humanos, pues se pretende que el agente imite su aprendizaje y comportamiento.

En este trabajo, un estado se define como una combinación de las variables que determinan el entorno radio de cada traspaso (p. ej., la velocidad del usuario, la carga de la celda, la interferencia ...). Del mismo modo, una acción es una configuración concreta de los parámetros de traspaso. Tanto los estados como las acciones pueden tomar únicamente valores discretos.

Cada vez que tiene lugar un evento n en el instante de tiempo t_n, el agente comprueba el beneficio de ejecutar una acción $a_{t_n} \in A$ cuando el entorno se encuentra en el estado $x_{t_n} \in X$ mediante la obtención de su recompensa asociada, $r(x_{t_n}, a_{t_n})$. A partir de estas observaciones, el agente elige aquellas acciones que maximicen la recompensa acumulada a lo largo del tiempo. Para ello, el agente usa una política voraz (del inglés, *greedy*) π que explora todas las acciones posibles y

encuentra la mejor acción a tomar en cada momento para maximizar las recompensas con el tiempo. Para alcanzar esta maximización, se define una función valor, $Q(x, a)$, que describe la recompensa esperada de una acción a cuando el entorno se encuentra en el estado x, como

$$Q(x, a) = E_\pi \left[r(x_{t_n}, a_{t_n}) \right] . \tag{4.2}$$

Para calcular $Q(x, a)$, se construye la conocida tabla Q (*Q-table*) con tantas filas como estados y tantas columnas como acciones. El objetivo de dicha tabla Q es almacenar y actualizar los valores Q, mostrando el rendimiento del sistema para cada par estado-acción (x, a) experimentado a lo largo del tiempo (en todos los eventos n). Los valores de recompensa se actualizan en la tabla Q cada vez que tiene lugar un evento n (un traspaso) en t_n usando la ecuación de Bellman:

$$Q^{(n+1)}(x, a) = (1 - \alpha) Q^{(n)}(x, a) + \alpha r(x_{t_n}, a_{t_n}) , \tag{4.3}$$

donde α es la tasa de aprendizaje. Con α se controla la rapidez con la que se actualizan los valores de la tabla Q. Un valor alto de α da mucha importancia a las recompensas instantáneas (es decir, al rendimiento de traspasos recientes), lo que puede provocar la divergencia del algoritmo. Por el contrario, un valor bajo de α hace que el algoritmo QL confíe más en las recompensas previas acumuladas (es decir, rendimiento de traspasos en el pasado) que en las recompensas instantáneas, lo que mejora la convergencia del sistema, aunque necesite más tiempo hasta alcanzar dicha convergencia. El superíndice n denota las versiones sucesivas de la tabla Q tras cada nuevo evento. Al final del proceso de convergencia, la tabla Q se contempla como la función Q definida en 4.2, que refleja la mejor acción que tomar en cada estado, $a_{max}(x)$, determinada como

$$a_{max}(x) = \max_a \left(Q^{(n_{end})}(x, a) \right) \quad , \tag{4.4}$$

objetivo del algoritmo QL. El superíndice n_{end} refleja el último evento, definido como el evento en que el proceso de actualización finaliza, pues no se consideran más eventos.

El algoritmo de optimización QL sigue un esquema iterativo en el que acciones, estados y recompensas se recogen durante un periodo de tiempo, y a partir de un momento, se eligen las mejores acciones para cada estado del entorno. Este proceso se repite hasta que se alcanza algún criterio de convergencia de la mejor recompensa. Este esquema iterativo requiere dividir el tiempo, para lo que se definen lapsos de tiempo, conocidos como intervalos de acción (IA). Durante un IA

se almacenan los valores (a, x, r) por evento, con los que se actualiza la tabla Q. Cuando finaliza el IA, se selecciona la mejor acción para cada estado, $a_{max}(x)$, a usar en el siguiente IA. El índice de IA, n_{IA}, denota el índice de iteración del algoritmo. En este trabajo, cada IA tiene una duración de 30 segundos de tiempo de red. Cabe recordar que el índice n en (4.3) se refiere a eventos, mientras que el índice n_{AI} indica el lapso de tiempo en que se obtienen las mejores acciones. Por tanto, la mejor acción por estado se puede definir para cada IA, $a_{max}(x, n_{IA})$.

No obstante, el rendimiento de la red en el último intervalo de acción puede no ser representativo del comportamiento global de la red, sobre todo al principio del proceso de optimización, cuando todavía no se han explorado en profundidad todas las acciones en todos los estados. Por esta razón, la trayectoria de los valores $\underset{a}{max}(Q(x_{nIA}, a_{nIA}))$ puede llevar a valores no óptimos de los parámetros de traspaso al final del proceso de optimización. Para asegurar una buena trayectoria de búsqueda, el agente de aprendizaje ha de explorar otras acciones, $a(x, n_{IA})$, incluso aunque dichas acciones no consigan el máximo valor Q (es decir, $a(x, n_{IA}) \neq a_{max}(x, n_{IA})$). El dilema de cuánto esfuerzo poner en probar nuevas acciones (es decir, configuraciones de parámetros de traspaso no óptimas) o tomar las acciones ya probadas (y que proporcionan altas recompensas) es el conocido como compromiso exploración-explotación [127]. Este compromiso se controla con una estrategia voraz de Epsilon decayente (del inglés, ε-*greedy*) que indica al agente cuánto explorar y cuánto explotar, y que evoluciona con el tiempo, haciendo decrecer progresivamente la proporción de exploración de acciones. La Figura 4.1 detalla la evolución temporal típica del referido compromiso exploración-explotación.

En la estrategia voraz de ε decayente, el valor $\varepsilon = 1$ indica que el agente siempre selecciona acciones nuevas en el próximo IA (es decir, no se tiene en cuenta el valor $a_{max}(x)$ de la tabla Q). Este valor $\varepsilon = 1$ se usa al principio del proceso de optimización durante un número de IAs determinado, correspondiente a un número de intervalos de acción $N_{explora}$. Tras este periodo de exploración, ε comienza a decrecer con el tiempo, de forma que las mejores acciones (aquellas con mayor valor Q) se seleccionan aleatoriamente con probabilidad ε. En caso contrario, se seleccionan las mejores acciones, $a_{max}(x, n_{AI})$. Esta fase en la que el valor de ε decrece dura $N_{entreno}$ IAs. Finalmente, en la fase de explotación, $\varepsilon = 0$, de forma que el agente únicamente selecciona las mejores acciones $a_{max}(x, n_{AI})$ calculadas en la tabla Q. Esta explotación de las mejores acciones, durante la que no se prueban nuevas acciones, solamente tiene sentido cuando las condiciones de red no cambian. En un escenario real, ε nunca debería valer 0, sino que debería mantenerse en un valor relativamente bajo para ser capaz de reaccionar ante cambios en la red, y así explorar nuevas acciones. En este trabajo, la duración de la fase de explotación, $N_{explota}$, se controla para asegurar que se recogen suficientes medidas para evaluar el rendimiento de los algoritmos analizados.

La estructura básica de un esquema QL se resume en el Algoritmo B.1. El bucle principal re-

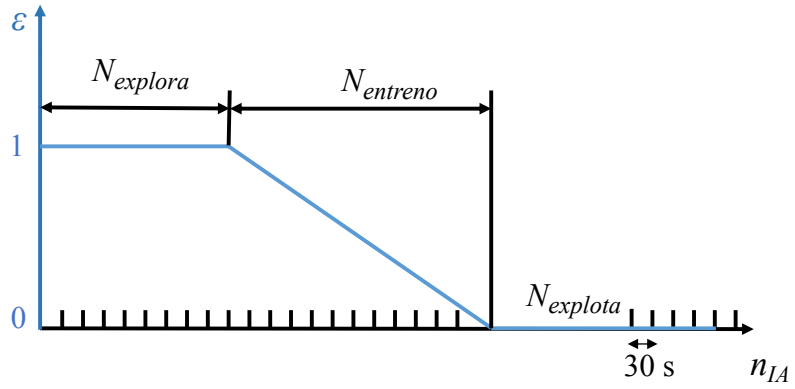


Figura 4.1: Estructura temporal del algoritmo QL MRO.

presenta las iteraciones a lo largo de los IAs. En cada iteración, en primero se actualiza el valor de ε, siguiendo la estrategia presentada en la Figura 4.1. En segundo lugar, se almacena información relativa a todos los eventos que tienen lugar en el IA (acciones y estados), y se calculan sus recompensas $r(x_{t_n}, a_{t_n})$. La tabla Q se actualiza con dichas recompensas usando (4.3). Finalmente, dependiendo del valor de ε, se deciden las acciones a tomar en la siguiente iteración. Todos los algoritmos MRO considerados en este trabajo siguen esta estructura. La diferencia entre los distintos algoritmos MRO se encuentra en la definición de la función recompensa, r, que se presentará en las siguientes secciones.

4.3.2. Algoritmo Q-MRO

Q-MRO es una implementación clásica de un algoritmo MRO usando un esquema Q-learning como el explicado en el apartado 4.3.1. El objetivo de Q-MRO es reducir el número de EHO, LHO y PP ajustando los parámetros de traspaso HOM y TTT. Este algoritmo propuesto en [28], se usa como punto de referencia. La parte esencial del algoritmo es la definición de la función de recompensa, r, con la que evaluar la bondad del rendimiento de un traspaso en particular [28], como

$$r(x_{t_n}, a_{t_n}) = -w_{RLF} X_{EHO}(n) - w_{RLF} X_{LHO}(n) - w_{PP} X_{PP}(n)\,, \tag{4.5}$$

donde $X_{EHO}(n)$, $X_{LHO}(n)$ y $X_{PP}(n)$ son variables binarias que toman los valores 0 o 1 dependiendo de la ocurrencia de EHO, LHO o PP para cada evento n. Dichas variables se multiplican por los pesos w_{RLF} y w_{PP}. En la mayoría de los casos, $w_{RLF} > w_{PP}$, pues se prefiere tener menos pérdidas

Algoritmo 4.1 Estructura del algoritmo MRO.

Entrada: estados x_{t_n} y acciones a_{t_n}, $\varepsilon(n_{AI})$
Salida: acciones a tomar en la siguiente iteración $a(x, n_{AI}+1)$

PARA $n_{AI} = 1, 2, ...$

calcula $\varepsilon(n_{AI})$
PARA cada evento n que tiene lugar en $t_n \in n_{AI}$

calcula $r(x_{t_n}, a_{t_n})$ con (4.5)
actualiza valor en la tabla Q, $Q^{(n+1)}(x_{t_n}, a_{t_n})$, con (4.3)

FIN

PARA cada $x \in X$

SI rand () $> \varepsilon(n_{AI})$
$a(x, n_{AI}+1) = a_{max}(x, n_{AI}+1) = \max_a \left(Q^{(n)}(x, a)\right)$, con (4.4)
SINO
selecciona una acción a aleatoria para $a(x, n_{AI}+1)$, con (4.4)
FIN

FIN

$n_{AI} = n_{AI} + 1$

FIN

de conexión aun a costa de tener más traspasos de ida y vuelta. En este trabajo, $w_{RLF} = 1$ y $w_{PP} = 0.5$. Si no ocurre ninguno de estos eventos, $r(x_{t_n}, a_{t_n}) = 0$.

4.3.3. Algoritmo E-MRO

A diferencia Q-MRO, E-MRO reduce RLF y PP, al mismo tiempo que se optimiza la QoE de los usuarios de borde de celda que realizan traspaso. Para ello, se ajustan los parámetros HOM y TTT por adyacencia para aumentar la QoE media de un usuario de borde de celda. En este trabajo, la ventana temporal en la que se evalúa la QoE de un usuario comprende 1 segundo antes y 1 segundo después del instante de disparo del traspaso. Esta corta ventana temporal pretende calcular la QoE del usuario únicamente en el borde de celda, aislando el efecto del traspaso en la QoE. El valor de QoE resultante, denotado por $QoE(n)$ (n por el enésimo evento de traspaso). El evento n se asocia al usuario u que realiza el enésimo evento de traspaso, con lo que los índices n y u_n pueden intercambiarse.

Con el objetivo descrito, la función de recompensa del algoritmo E-MRO, a diferencia de Q - MRO, se calcula como la suma de tres términos, según

$$r(x_{t_n}, a_{t_n}) = r_{radio}(x_{t_n}, a_{t_n}) + r_{QoE_{step}}(x_{t_n}, a_{t_n}) + r_{QoE_{prev}}(x_{t_n}, a_{t_n}) \ , \tag{4.6}$$

donde r_{radio} es la recompensa relativa a la robustez del enlace radio, $r_{QoE_{step}}$ es la recompensa debida al cambio de QoE que se produce en el evento de traspaso por el hecho de cambiar de celda, y $r_{QoE_{prev}}$ es una recompensa negativa (una penalización) debido a una QoE baja antes del traspaso, definidas como

$$\begin{aligned} r_{radio}(x_{t_n}, a_{t_n}) &= -w_{RLF} X_{EHO}(n) - w_{RLF} X_{LHO}(n) - w_{PP} X_{PP}(n) \ , \\ r_{QoE_{step}}(x_{t_n}, a_{t_n}) &= \max(-1, \min(1, (QoE^D(u_n) - QoE^A(u_n)))) \ , \\ r_{QoE_{prev}}(x_{t_n}, a_{t_n}) &= \max(-1, \min(0, (\frac{QoE^A(u_n)}{\overline{QoE^A(w,x)}} - \frac{\overline{QoE^A(w,x)}}{\overline{QoE^A(w,x)}}))) \ . \end{aligned} \tag{4.7}$$

El primer término, r_{radio}, que refleja el rendimiento del traspaso desde el punto de vista radio, es idéntico a la recompensa r definida en (4.5) para Q-MRO. Los otros dos términos, $r_{QoE_{step}}$ and $r_{QoE_{prev}}$, son propios de E-MRO. $r_{QoE_{step}}$ permite maximizar la mejora de QoE introducida por el traspaso, calculada como la diferencia de las QoE después y antes del traspaso, $QoE^D(u_n)$ y $QoE^A(u_n)$, respectivamente. Por tanto, $r_{QoE_{step}}$ premia los valores de HOM y TTT que hacen que $QoE(u_n)$ en la celda destino sea mejor que $QoE(u_n)$ en la celda servidora. La diferencia de QoE se limita al intervalo $[-1, 1]$, para garantizar que la recompensa $r_{QoE_{step}}$ resultante sea comparable a r_{radio}.

Si solamente se tiene en cuenta r_{radio} y $r_{QoE_{step}}$ en (4.6), E-MRO podría llevar a configurar los parámetros de traspaso de forma que se consiga un valor alto de recompensa r por una diferencia $QoE^D(u)$-$QoE^A(u)$ alta, pero provocada por una QoE de usuario muy baja en la celda origen (es decir, una reducción de $QoE^A(u)$ y no por un aumento de $QoE^D(u_n)$). El tercer término $r_{QoE_{prev}}$ de (4.6) busca limitar estas configuraciones erróneas de los parámetros de traspaso, penalizando aquellas en las que la QoE de usuario antes del traspaso sea menor que un valor medio, $\overline{QoE^A(w,x)}$. El valor medio $\overline{QoE^A(w,x)}$ incluye la QoE antes del traspaso de cualquier usuario w que efectúe un traspaso entre cualquier par de celdas del escenario siempre que se encuentren dentro del mismo estado x que la adyacencia del usuario u_n. En dicho valor medio, solamente se consideran los traspasos dentro de una ventana temporal que comprende el periodo inicial y final del proceso de optimización. De esta forma, $r_{QoE_{prev}}$ penaliza las configuraciones de parámetros de traspaso que degradan la QoE del usuario en la celda origen (antes del traspaso) comparado con el rendimiento

medio de los usuarios antes del traspaso bajo condiciones simulares (en el mismo estado).

Se diseñan dos variantes del algoritmo E-MRO: EQ-MRO y EN-MRO. EQ-MRO sigue la descripción anterior usando una tabla Q para almacenar y actualizar los valores Q, con un valor muy conservador de $\alpha = 0.005$. De esta forma, los cambios en la tabla Q son muy lentos. Con este valor tan bajo, se pretende maximizar los valores Q de cada par estado-acción, $Q(x, a)$, a largo plazo.

Como alternativa, EN-MRO reemplaza la tabla Q por una ANN para aproximar la relación entre estado-acción-recompensa [128]. Sustituir la tabla Q por una red neuronal artificial permite estimar $Q(x, a, \theta)$ para pares estado-acción no explorados, donde θ representa los pesos entrenables de la ANN [128]. Este trabajo usa una ANN poco profunda [1] con dos capas ocultas. Esta ANN sencilla se usa para reducir la necesidad de disponer de grandes conjuntos de datos, evitando el problema del sobreajuste con juegos de datos pequeños. Concretamente, la ANN se diseña como un perceptrón multicapa (*Multi-Layer Perceptron*, MLP) de dos capas ocultas, entrenado con un algoritmo de propagación hacia atrás basado en el algoritmo de optimización de Levenberg-Marquardt, como sustituto de cada fila de la tabla [129]. Aunque esta técnica permite la exploración de estados continuos (es decir, cualquier valor de los parámetros HOM y TTT), se mantienen estados y acciones discretos para reducir la complejidad del algoritmo. Este número limitado de estados está en consonancia con las redes LTE reales en las que el equipamiento no acepta valores continuos de HOM y TTT.

Aunque tanto en Q-MRO como en E-MRO se podrían utilizar funciones recompensa más sencillas, basadas en indicadores de calidad de señal (como la SINR media o la tasa de pérdida de paquetes), de esa forma se perdería información relevante, al no considerar las tasas de EHO, LHO o PP, que en último término son las cifras de mérito que deben guiar el proceso de optimización.

4.3.4. Complejidad del algoritmo

La complejidad temporal del algoritmo Q-learning en EQ-MRO es proporcional al tamaño de la tabla Q, que viene dada por el producto del número de estados, N_s, y el de acciones, N_a. El número de estados crece linealmente con el número de servicios. Por tanto, el caso peor de la complejidad temporal de EQ-MRO es $O(N_s, N_a)$. Para EN-MRO, la complejidad no sólo depende del número de estados y acciones, sino también del tamaño de las ANN. El caso peor teórico de la complejidad temporal para el algoritmo de propagación hacia atrás usado para entrenar la ANN en EN-MRO con N_i entradas, 1 salida y N_{hl} capas ocultas es $O(N_i, N_{hl}, N_{s-hl}, N_{it})$, donde N_{s-hl}

[1] Una red neuronal artificial poco profunda tiene hasta dos capas ocultas (al contrario de las redes neuronales artificiales profundas que tienen múltiples capas ocultas).

es el tamaño de las capas ocultas y N_{it} es el número de iteraciones en las que se reentrena la ANN. De la explicación anterior, se deduce que el número de servicios considerado en el escenario para evaluar el rendimiento del algoritmo, tiene un impacto directo en la complejidad del mismo (mediante N_s) y en el tiempo de convergencia (ya que un número N_s mayor requiere de mayores tiempos de simulación para acumular un número de eventos por acción suficientemente alto).

Existen varias evidencias de que Q-learning converge a la función Q óptima, siempre y cuando se seleccionen la política de exploración y la tasa de aprendizaje correctas [130]. La exploración requiere asegurar que cada acción se toma un número de veces lo suficientemente alto. Del mismo modo, la suma de las tasas de aprendizaje ha de tender a infinito (para que pueda alcanzarse cualquier valor), mientras que la suma de los cuadrados de las tasas de aprendizaje ha de ser finita (para asegurar la convergencia) [131]. Ambas condiciones se garantizan en el esquema de aprendizaje propuesto, que comienza con una tasa de aprendizaje alta para permitir cambios rápidos y gradualmente, se reduce con el tiempo.

La convergencia de QL en la práctica se ha estudiado en profundidad en [79]. Un aspecto clave para favorecer la convergencia es usar valores Q iniciales optimistas. En este trabajo, el valor Q inicial para cada par estado-acción es cero, que supone un valor optimista, pues los valores Q finales son negativos, como se muestra más adelante. Además, cuanto mayor es el número de eventos almacenados por estado y acción, más rápida es la convergencia, ya que el comportamiento de la red se conoce mejor y, por tanto, pueden seleccionarse las mejores acciones para el siguiente IA. Por ello, es necesario elegir una duración adecuada para cada IA (30 segundos en este trabajo). Para aumentar el número de eventos por acción, se sigue una estrategia de eliminación de acciones que progresivamente reduce el espacio de posibles valores estado-acción, como se explica en la Sección 4.4.3.

4.4. Análisis del rendimiento

Los algoritmos descritos anteriormente se prueban en el simulador dinámico LTE de nivel de sistema ya descrito [101]. En esta sección, se presenta primero la metodología experimental y después los resultados.

4.4.1. Metodología experimental

A continuación, se describe el espacio de estados y acciones en primer lugar, para después presentar la configuración de los experimentos.

4.4.1.1. Definición de estados y acciones

Tanto Q-MRO y E-MRO comparten el mismo conjunto de estados del sistema y acciones de control. Como se ilustra en la Tabla 4.1, cada estado se modela por un cuarteto $\{s,v,d,l\}$, determinado por la combinación de servicio, s, velocidad de usuario, v, distancia entre emplazamientos, d, y carga de celda destino, l.

a) Servicio, s

Con este índice, se identifica al servicio asociado al estado en cuestión. Inicialmente, s podría tener cuatro valores correspondientes a los servicios de VoIP, FTP, WEB y VIDEO. Sin embargo, como se explicó en el capítulo anterior, en el escenario considerado, inspirado en una red LTE comercial, el tráfico VoIP es muy escaso y se concentra en unas pocas celdas de la red [115]. Como esta situación puede causar que las estadísticas de RLF y QoE sean poco fiables, no se permite a Q-MRO y E-MRO cambiar los parámetros de traspaso para este servicio, es decir, $HOM(i,j,VoIP) =$ 0 dB y $TTT(i,j,VoIP) = 40$ ms en Q-MRO y E-MRO, donde i y j representan a las celdas origen y destino, respectivamente. Por consiguiente, finalmente solo se usan tres servicios para la definición del espacio de estados.

b) Velocidad de usuario, v

La velocidad del usuario se denota por v, que en este trabajo puede tomar los valores 30 o 70 km/h. Estas dos velocidades modelan usuarios que se mueven en calles de ciudad o en autovías. Las velocidades más bajas, asociadas a los peatones no se consideran en este trabajo, porque esos usuarios raramente efectúan traspasos, especialmente en los servicios de transmisión de datos, que presentan duraciones pequeñas de sesión. Así, las técnicas MRO no tienen un gran impacto sobre este tipo de usuarios.

Parámetro	Valores posibles	Índice
Servicio s	{FTP,VIDEO,WEB}	{1,2,3}
Velocidad de usuario v **[km/h]**	{30,70}	{1,2}
Distancia entre emplazamientos d **[km]**	{≤1.25,>1.25}	{1,2}
Carga de celda destino l **[%]**	{≤70,>70}	{1,2}

Tabla 4.1: Índices de parámetros que definen el espacio de estados.

c) Distancia entre emplazamientos, d

Las adyacencias de la red se clasifican según la distancia entre emplazamientos (*Inter-Site Distance*, ISD). Se distinguen 2 grandes grupos de adyacencias, dependiendo de si el ISD es mayor o menor que 1.25 km, para diferenciar entre entornos urbanos y suburbanos-rurales. De esta forma, del proceso de aprendizaje se obtiene una acción distinta dependiendo del entorno.

d) Carga media de celda destino, l

Las adyacencias también se clasifican según el valor de carga media de la celda destino, para diferenciar entre celdas sobrecargadas y no sobrecargadas. Para ello, se establece un umbral del 70 % de carga para considerar una celda como sobrecargada.

Con la configuración descrita, el número de estados es $3 \cdot 2 \cdot 2 \cdot 2 = 24$ estados. Cabe recordar que el objetivo del algoritmo de optimización es encontrar la configuración óptima de los parámetros de traspaso para cada estado, para lo que se necesita efectuar muchos traspasos por estado. Un número mayor de índices por cada uno de los parámetros que definen un estado (s, v, d o l) mejoraría la precisión al caracterizar el escenario. Sin embargo, un número de estados más elevado también implica un aumento considerable del número de eventos de traspaso necesarios para aprender el comportamiento óptimo del sistema, lo que supondría un aumento significativo del periodo de evaluación. Para simplificar el análisis, la Tabla 4.1 presenta los índices de los parámetros usados para diferenciar los estados y la Tabla 4.2 enumera los estados (desde el 1 hasta el 24) con sus índices por cada parámetro.

Se considera un total de 45 acciones posibles por estado, correspondientes a las combinaciones de los 15 valores posibles de HOM (desde -7 dB hasta +7 dB en pasos de 1 dB) y 3 valores de TTT (40, 100 y 256 ms). Por razones de espacio, la Tabla 4.3 presenta únicamente un subconjunto de 15 acciones, a, respecto al total de las 45 acciones, considerando sólo los valores $HOM \in \{-2, -1, 0, 1, 2\}$ dB y $TTT \in \{40, 100, 256\}$ ms, que se usarán más adelante.

Estado x	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
s	1	1	1	1	1	1	1	1
v	1	2	1	2	1	2	1	2
d	1	1	2	2	1	1	2	2
l	1	1	1	1	2	2	2	2
Estado x	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
s	2	2	2	2	2	2	2	2
v	1	2	1	2	1	2	1	2
d	1	1	2	2	1	1	2	2
l	1	1	1	1	2	2	2	2
Estado x	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**
s	3	3	3	3	3	3	3	3
v	1	2	1	2	1	2	1	2
d	1	1	2	2	1	1	2	2
l	1	1	1	1	2	2	2	2

Tabla 4.2: Enumeración de los estados.

Tabla 4.3: Enumeración del espacio de acciones.

Acción a	**1**	**2**	**3**	**4**	**5**
HOM [dB]	-2	-1	0	1	2
TTT [ms]	40	40	40	40	40
Acción a	**6**	**7**	**8**	**9**	**10**
HOM [dB]	-2	-1	0	1	2
TTT [ms]	100	100	100	100	100
Acción a	**11**	**12**	**13**	**14**	**15**
HOM [dB]	-2	-1	0	1	2
TTT [ms]	256	256	256	256	256

4.4.2. Experimentos

Para evaluar el rendimiento de las dos variantes del algoritmo E-MRO, se llevan a cabo tres experimentos en el mismo escenario que en el capítulo anterior, ilustrado en la Figura 3.7, que consiste en 108-macroceldas (36 emplazamientos con 3 antenas tri-sectoriales por emplazamiento). Los valores de los parámetros de simulación son los mostrados en la Tabla 3.2, excepto tres. En este caso, la resolución temporal se configura en 20 ms, para reducir la carga computacional de las simulaciones, ya que los periodos de evaluación del algoritmo E-MRO son muy largos. La velocidad de los usuarios se reparte entre 30 y 70 km/h atendiendo a los grupos establecidos al definir el espacio de soluciones. Por último, el ancho de banda del sistema es de 5 MHz (25 PRB) de nuevo, para reducir la carga computacional de las simulaciones. Por el mismo motivo, en todos los experimentos se simula únicamente el enlace descendente (*DownLink*, DL).

Experimento 1: Limitaciones de Q-MRO

El primero de los experimentos busca descubrir las limitaciones de Q-MRO, demostrando que ignorar las restricciones de QoE en el ajuste de parámetros lleva a bajos rendimientos de QoE. Para ello, se define un experimento sencillo en el que solo hay un estado para toda la red. Este estado está caracterizado por un solo servicio (FTP) y una velocidad de usuario fija (30 km/h). La distancia entre emplazamientos y la carga de la celda destino no se clasifican. Por su parte, el espacio de acciones A sólo incluye una dimensión, que cubre los valores de HOM en el rango entre -7 dB y 7 dB en pasos de 1 dB. Esto supone 15 acciones posibles, por lo que la tabla Q es un vector de 1x15. Se elige esta horquilla de valores de HOM porque, para la mayoría de los proveedores de red, los usuarios con una SINR por debajo de -7 dB no reciben recursos radio en el proceso de planificación. Recuérdese que el valor de HOM puede usarse como aproximación del valor de SINR experimentado por el usuario tras el traspaso [14]. TTT se fija a un valor de 40 ms, el menor valor no nulo posible según el 3GPP [125]. Seleccionando un valor bajo de TTT, que no restringe el proceso de traspaso, se consigue que el sistema muestre el impacto de las distintas configuraciones de HOM. Para simplificar el análisis, los usuarios se distribuyen de forma uniforme dentro de cada celda.

Como método de ajuste, se utiliza el algoritmo Q-MRO clásico con las recompensas definidas en (4.5). El periodo inicial se establece en $N_{explora} = 100$ (50 minutos), seguido de una fase de entrenamiento de 3 horas, $N_{entreno} = 360$, y de una fase de explotación de 2 horas, $N_{explota} = 240$, para asegurar que el algoritmo es capaz de encontrar y explotar la configuración óptima. Se monitorizan cinco cifras de mérito durante todo el proceso de optimización.

La primera es la QoE media de usuario de borde de celda, definida como

$$\overline{QoE_{borde}} = \frac{1}{2N_u}\sum_u (QoE^D(u) + QoE^A(u)) \ , \tag{4.8}$$

donde N_u es el número de usuarios involucrados en un evento de traspaso (EHO, LHO, PP o SHO) durante un intervalo de acción en todo el escenario.

La QoE para el servicio FTP se calcula como se indica en (2.3), lo que requiere estimar caudal de usuario. El caudal de usuario, $T(u)$, durante la ventana temporal en la que se evalúa la QoE (es decir, 1 segundo) se calcula mediante un filtro auto-regresivo (AR) con un único coeficiente $\beta = 0.98$ como

$$T_t(u) = (1-\beta)\ T_{t-1}(u) + \beta\ T_t(u) \ , \tag{4.9}$$

donde $T_t(u)$ es el *throughput* en el paso de simulación t.

Las otras cuatro métricas son indicadores tradicionales de rendimiento de traspaso HO que reflejan las tasas de LHO, EHO, PP y SHO, calculadas como

$$LHO\ [\%] = 100\frac{N_{LHO}}{N_{eventos}} \ , \tag{4.10}$$

$$EHO\ [\%] = 100\frac{N_{EHO}}{N_{eventos}} \ , \tag{4.11}$$

$$PP\ [\%] = 100\frac{N_{PP}}{N_{eventos}} \ , \tag{4.12}$$

$$SHO\ [\%] = 100\frac{N_{SHO}}{N_{eventos}} = 100\frac{N_{events} - N_{LHO} - N_{EHO} - N_{PP}}{N_{eventos}} \ , \tag{4.13}$$

donde N_{LHO}, N_{EHO}, N_{PP} y N_{SHO} son el número de LHO, EHO, PP y HOs realizados con éxito (SHO) en cada intervalo de acción, y $N_{eventos}$ es el número total de eventos de traspaso, con $N_{eventos} = N_{LHO} + N_{EHO} + N_{PP} + N_{SHO}$.

Tras analizar estas cinco cifras de mérito, se seleccionan las mejores acciones (valores de HOM) elegidas por Q-MRO para formar el espacio de acciones del segundo experimento.

Experimento 2: comparación de Q-MRO y E-MRO en escenario de 1 servicio

En el segundo experimento, se comparan el algoritmo E-MRO propuesto y el esquema tradicional Q-MRO mediante dos simulaciones (dos fases) de complejidad creciente en lo que se refiere al número de estados y acciones. Con el fin de comparar los distintos algoritmos, se define otro indicador importante: la QoE media de todos los usuarios de la red $\overline{QoE}_u$, considerando la sesión completa al final del proceso de optimización como

$$\overline{QoE}_u = \frac{1}{N_u}\sum_u QoE(u) \ . \tag{4.14}$$

Adicionalmente, también se usa como indicador de rendimiento el *throughput* medio de usuario, $\overline{T}$, definido como

$$\overline{T} = \frac{1}{N_u}\sum_{\forall u} T(u) \ , \tag{4.15}$$

donde $T(u)$ es el *throughput* medio del usuario u. Cabe recordar que el índice u en (4.14) y (4.15) denota todos los usuarios del escenario, no solamente aquellos que realizan traspaso, a diferencia de (4.8). Por tanto, $\overline{T}$ y $\overline{QoE}$ son indicadores de rendimiento globales, mientras que $\overline{QoE_{borde}}$ evalúa la experiencia de los usuarios de borde de celda. También es importante destacar que las cifras intermedias de *throughput* se presentan para comprender mejor el rendimiento de los distintos algoritmos. Por el contrario, los indicadores de QoE se definen para cuantificar el rendimiento final de las técnicas MRO.

El objetivo de la primera fase del experimento 2 es encontrar el mejor valor de HOM, en términos de QoE, entre todos los disponibles en el espacio de acciones. Para ello, se utiliza la misma configuración del espacio de estados que en el primer experimento, con un único estado para toda la red. Por su parte, el espacio de acciones está formado solo por 5 valores de HOM diferentes (es decir, TTT se considera fijo a un valor por defecto). Esos 5 valores de HOM son los que obtienen mejores rendimientos en el primer experimento. En este caso, $N_{explora} = 50$ (25 minutos) y $N_{entreno} = 300$ (2.5 horas), debido al menor número de valores de HOM en el espacio de acciones a probar. Como en el primer experimento, se simulan 6 horas de tiempo de red. EQ-MRO usa una tabla Q de 1x5

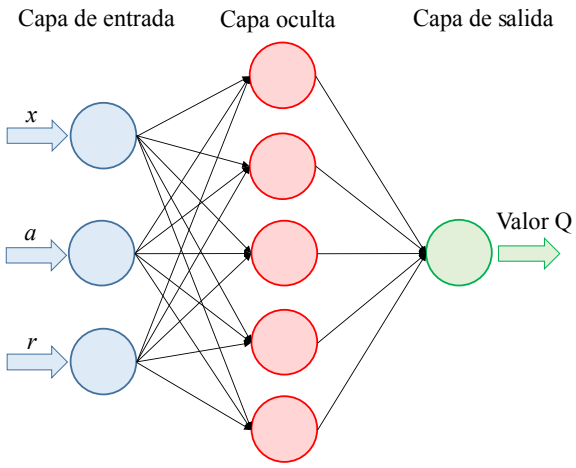


Figura 4.2: Red neuronal artificial (1ª fase, Experimento 2).

para guardar el valor Q de cada par estado-acción, mientras que EN-MRO sustituye esa tabla Q por una ANN con una única capa oculta de 5 neuronas (sólo para este experimento), como la que se ilustra en la Figura (4.2).

La ANN se entrena por primera vez tras $N_{entreno}$ intervalos de acción, y se reentrena cada 20 intervalos de acción (es decir, cada 10 minutos), tiempo suficiente para que se generen nuevos datos de los que aprender. Conforme pasa el tiempo, se prueban menos acciones, pues ε se va reduciendo para asegurar que se toma la mejor acción, siguiendo la evolución mostrada en la Figura 4.1. Cabe destacar que, aunque este esquema sea efectivo, la idoneidad del resultado de esta primera fase del experimento 2 es parcial, ya que se ha configurado un valor fijo para el TTT y se implementa el mismo valor de HOM para todas las adyacencias de la red, independientemente de su naturaleza (es decir, la velocidad del usuario que efectúa el traspaso, la distancia entre la celda servidora y la adyacente y la carga de la celda destino).

En una segunda fase del experimento 2, se comparan tres algoritmos MRO: Q-MRO, EQ-MRO y EN-MRO, tomando Q-MRO como referencia. El objetivo de esta segunda fase es encontrar las mejores configuraciones de HOM y TTT por adyacencia para un servicio concreto (FTP). Para ello, se consideran diferentes velocidades de usuario (30 y 70 km/h) con igual probabilidad dentro de cada celda; también se consideran diferentes ISD y cargas de celda vecina, representadas por los parámetros d y l. Las combinaciones de estos parámetros dan lugar a los estados 1-8 de la Tabla 4.2, que corresponden todos al servicio FTP ($s = 1$). Los estados 2, 4, 6 y 8 cubren los casos en que los usuarios se mueven a gran velocidad, los estados 3, 4, 7 y 8 hacen lo propio para grandes distancias entre emplazamientos, y los estados 5, 6, 7 y 8 para altas cargas de celda adyacente.

En esta segunda fase del experimento 2, el espacio de acciones se completa añadiendo el parámetro de traspaso TTT a las 5 mejores acciones seleccionadas en el primer experimento (es decir, los 5 mejores valores de HOM), $TTT = 40$, 100 y 256 ms, como se definió en la Tabla 4.3. En este punto, todavía se simula únicamente el servicio FTP. En esta segunda fase, $N_{explora} = 150$ (1.25 horas) y $N_{entreno} = 480$ (4 horas). La duración de la simulación es de 8 horas de tiempo de red. Se necesitan periodos más largos, ya que se considera un mayor número de acciones a probar (aumento de 5 a 15).

Para esta segunda fase, EQ-MRO usa una tabla Q de dimensiones 8x15 (estados·acciones), mientras que EN-MRO sustituye la tabla Q por una ANN poco profunda con dos capas ocultas de 4 neuronas cada una. La ANN se entrena por primera vez tras $N_{explora}$ intervalos de acción y se vuelve a entrenar cada 40 intervalos de acción (es decir, cada 20 minutos), pues el sistema necesita más tiempo para recoger datos nuevos de los que aprender. Para acelerar el proceso de aprendizaje, tras el periodo inicial $N_{explora}$ (75 minutos) y un primer entrenamiento (20 minutos), en sucesivos entrenamientos se elimina del proceso de aprendizaje una acción cada 20 minutos, aquella con menor valor Q. Este mecanismo se repite 10 veces para eliminar las 10 peores acciones durante la fase de aprendizaje. Al final de dicha fase de aprendizaje, solamente quedan las 5 mejores acciones (en este caso, combinaciones de valores de HOM y TTT).

Como resultado de esta segunda fase, se obtienen los mejores parámetros de traspaso, HOM y TTT, por adyacencia, teniendo en cuenta sus particularidades (p. ej., la velocidad de usuario, la distancia entre emplazamientos, la carga de la celda destino ...) cuando solamente hay usuarios del servicio FTP.

Experimento 3: comparación de Q-MRO y E-MRO en escenario multiservicio

En el tercer experimento, se considera el sistema completo con todos los servicios (FTP, VIDEO y WEB). Los usuarios WEB alternan periodos de descarga de información (páginas web) con otros periodos de lectura. Si tiene lugar un evento de traspaso cuando un usuario WEB está leyendo (es decir, no se está descargando información), no se tiene en cuenta la QoE de ese usuario para la optimización del traspaso. La QoE de los usuarios del servicio VIDEO no depende directamente del *throughput* de usuario (2.2), sino de la frecuencia y duración media de las interrupciones de la reproducción (*stalling*). Como el algoritmo de optimización necesita una estimación de la QoE antes y después de un evento de traspaso, se ha supuesto que, para cada usuario del servicio VIDEO, se produce como máximo un *stalling* en la ventana temporal considerada en torno al traspaso (es decir, 1 segundo). Como se hizo para el *throughput* de usuario, para realizar el cálculo de la duración media de stalling (DS) en la ventana temporal en la que se evalúa la QoE se utiliza un filtro AR

Tabla 4.4: Indicadores de tráfico de celda.

Indicador	**Mínimo**	**Media**	**Máximo**
$N_u(i, VIDEO)/N_u(i)$ [%]	4.3	33.03	50.7
$N_u(i, FTP)/N_u(i)$ [%]	16.9	29.46	72.25
$N_u(i, WEB)/N_u(i)$ [%]	22.1	37.51	47.62
$U(i)$ [%]	4.72	58.18	95.27

de un único coeficiente como sigue

$$DS_t(u) = \beta \; DS_{t-1}(u) + \Delta t \; , \tag{4.16}$$

donde DS_t es la duración media de *stalling* en el paso de simulación t, $\beta = 0.98$ y Δt es la duración del paso de simulación (20 ms, según la Tabla 3.1).

En este tercer experimento, se considera un espacio de 24 estados y 15 acciones posibles por estado, lo que precisa mayores tiempos de simulación. Concretamente, $N_{explora} = 300$ (2.5 horas), $N_{entreno} = 960$ (8 horas) y la duración de la simulación completa es de 11 horas. EQ-MRO usa una tabla Q de dimensiones 24x15 para almacenar los valores Q por cada par estado-acción, mientras que EN-MRO sustituye la tabla Q por la misma ANN poco profunda que en el segundo experimento. La ANN se entrena por primera vez tras $N_{explora}$ intervalos de acción, y se vuelve a entrenar cada 75 intervalos de acción (es decir, 37.5 minutos) mientras que $\varepsilon > 0.75$, o cada 50 intervalos de acción (es decir, 25 minutos) en el resto del tiempo. Para acelerar el proceso de aprendizaje, se elimina una acción cada 37.5 minutos si $\varepsilon > 0.75$, o cada 25 minutos en caso contrario. Este mecanismo se repite 10 veces.

La Tabla 4.4 presenta algunos indicadores de la distribución de tráfico en el escenario configurado en este experimento. Estos indicadores se recogen por celda de la red real. Las tres primeras filas muestran la mezcla de servicios presentando la tasa de usuarios de cada servicio. $N_u(i, s)$ denota el número de usuarios de cada servicio s en la celda i y $N_u(i)$ es el número total de usuarios en la celda i. La última fila muestra la carga media de celda, $U(i)$, medida como la utilización media de PRB en una celda. Las columnas representan los valores promedio y extremos de estos indicadores.

4.4.3. Resultados

A continuación, se presentan los resultados de cada uno de los experimentos descritos anteriormente.

Experimento 1: Limitaciones de Q-MRO

La Figura 4.3 muestra la evolución las distintas cifras de mérito a lo largo del tiempo, conforme se suceden los intervalos de acción, n_{IA}. Recuérdese que el periodo inicial comprende los 100 primeros intervalos de acción (en este experimento, 50 IAs más que la fase de exploración) y, el periodo de aprendizaje, otros 360 intervalos de acción. Por tanto, el rendimiento final se puede observar tras el intervalo de acción 460, aproximadamente. En la figura se superponen 2 curvas: una continua, con alta variabilidad, que representa el valor para cada intervalo de acción, y otra discontinua, que representa una media móvil de 100 muestras de la curva continua. Para mayor claridad, para la evaluación del algoritmo, se toma el primer y último valor de la media móvil, que representan el valor medio de los periodos inicial y final (de 100 intervalos de acción) de todo el proceso de optimización. Es importante recordar que los valores de rendimiento iniciales son los mismos para todos los métodos.

La Tabla 4.5 presenta los valores inicial y final de las distintas cifras de mérito para el primer experimento. Como se muestra en la tabla, los LHO quedan prácticamente eliminados al final del proceso de optimización ($\overline{LHO} = 2\,\%$, comparado con $\overline{LHO} = 18\,\%$ al principio). Por el contrario, EHO ya presentaba un valor bajo al principio del proceso (1 %), debido a la forma en que está diseñado el mecanismo de traspaso en la herramienta de simulación. No obstante, los EHO también se reducen al final del proceso de optimización (0.1 %). Con relación a los traspasos PP, desde el principio se obtienen valores relativamente altos, que no se reducen significativamente al final del proceso (21.6 y 19 %, respectivamente). Este resultado se debe a que la métrica de PP tiene un menor peso en la función de recompensa (4.5). Por último, aumenta la tasa de SHO como resultado de todas las mejoras (reducciones) en el resto de eventos de traspaso.

La Figura 4.4 ilustra la evolución del valor Q para las acciones $HOM(i,j) =$ -2 dB, -1 dB, 0 dB, 1 dB y 2 dB. Éstas son las 5 mejores acciones seleccionadas por Q-MRO de las 15 acciones probadas en el experimento. La mejor acción seleccionada por Q-MRO es $HOM(i,j) = 0$ dB, que sitúa el traspaso en el punto en que el usuario recibe la señal de ambas celdas con la misma potencia, lo que consigue maximizar $\overline{SHO}$. Se observa que los valores positivos de HOMs se comportan mejor que los valores negativos, ya que estos últimos aumentan notablemente los eventos PP, que es el evento

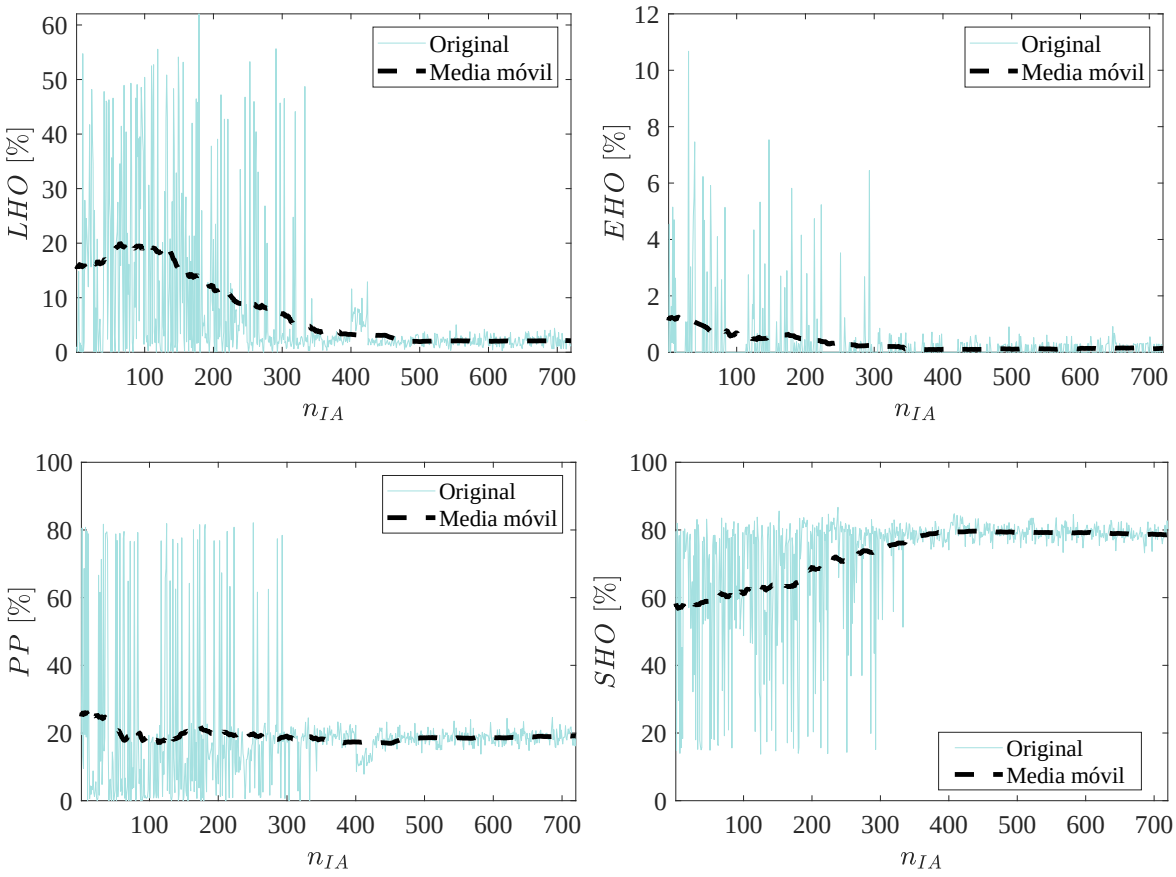


Figura 4.3: Rendimiento de Q-MRO (Experimento 1).

Tabla 4.5: Rendimiento de Q-MRO (Experimento 1).

	Inicial	**Final**
$\overline{EHO}$ [%]	1	0.1
$\overline{LHO}$ [%]	18.52	2
$\overline{PP}$ [%]	21.6	19
$\overline{SHO}$ [%]	58.88	78.9
$\overline{QoE_{borde}}$ [MOS]	1.48	1.28

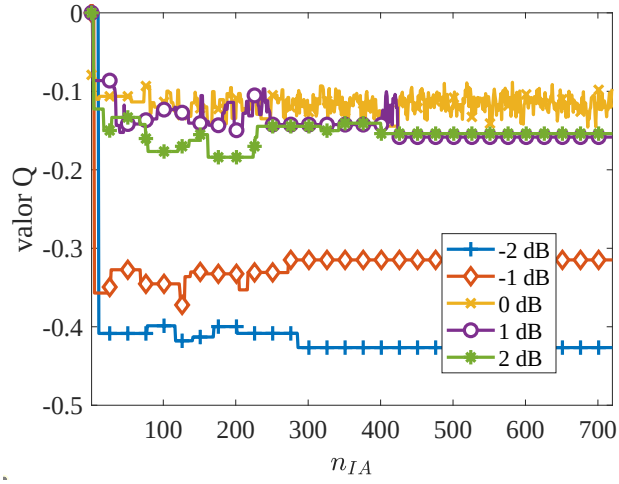


Figura 4.4: Evolución del valor Q para las 5 mejores acciones (Experimento 1).

de traspaso más frecuente. Estos 5 valores de HOM se seleccionan para los siguientes experimentos.

Finalmente, la Figura 4.5 muestra la evolución de la QoE de borde de celda durante el proceso de optimización. Cabe recordar que en este experimento no se ha tenido en cuenta la QoE de forma explícita en la función de recompensa. Concretamente, los valores inicial y final de $\overline{QoE_{borde}}$ son, respectivamente, $\overline{QoE_{borde}} = 1.48$ y $\overline{QoE_{borde}} = 1.28$. De estas cifras, se concluye que la mejora del rendimiento obtenida por Q-MRO es a costa de deteriorar la QoE de los usuarios de borde de celda en 0.2 puntos de MOS. Este resultado era previsible al no haber tenido en cuenta la QoE en la función de recompensa (4.5).

Experimento 2: comparación de Q-MRO y E-MRO en escenario de 1 servicio

La Tabla 4.7 recoge el rendimiento de la red con los distintos enfoques MRO al principio y al final del proceso de optimización. Es importante recordar que en esta primera fase del experimento 2, solamente están disponibles los 5 mejores valores de HOM seleccionados en el Experimento 1. Para mayor claridad, se resalta en gris el mejor resultado para cada cifra de mérito. Se observa que Q-MRO, EQ-MRO y EN-MRO alcanzan valores similares de traspasos exitosos al final del proceso de optimización ($\overline{SHO} \approx 80\,\%$), pero eliminando diferentes eventos de traspaso indeseados. Mientras que Q-MRO reduce LHO en un 4.78 % (desde 6.63 hasta 1.97 %), PP sólo se reduce en un 0.54 % y EHO en un 0.09 %. Por contra, EQ-MRO reduce PP en un 12.3 % (desde 20.25 hasta 7.89 %), mientras que LHO aumenta en un 5.52 %. EN-MRO consigue rendimientos similares a los de EQ-MRO, reduciendo PP en un 13 % (desde 20.99 hasta 7.98 %) y LHO aumenta en un 6.17 %.

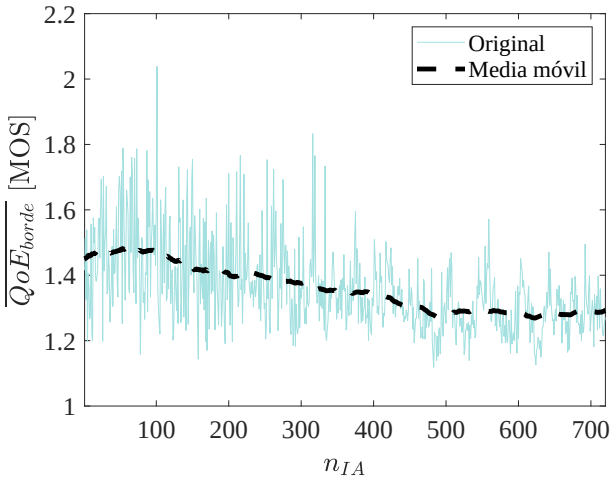


Figura 4.5: Evolución de QoE de usuarios de borde celda.

Tabla 4.6: Rendimiento de métodos (1ª fase, Experimento 2).

	Q-MRO		**EQ-MRO**		**EN-MRO**	
	Inicial	**Final**	**Inicial**	**Final**	**Inicial**	**Final**
$\overline{EHO}$ [%]	0.23	0.14	0.26	0.14	0.27	0.14
$\overline{LHO}$ [%]	6.75	1.97	6.63	12.15	6.09	12.26
$\overline{PP}$ [%]	19.22	18.68	20.24	7.89	20.99	7.98
$\overline{SHO}$ [%]	73.8	79.20	72.87	79.95	72.65	79.75
$\overline{QoE_{borde}}$[MOS]	1.34	1.28	1.34	1.41	1.34	1.43
$\overline{QoE}_u$[MOS]	-	4.05	-	4.19	-	4.22

Por lo tanto, el algoritmo Q-MRO elige reducir LHO, mientras que EN-MRO y EQ-MRO reducen PP a costa de aumentar LHO, ya que es la única forma de aumentar $\overline{QoE_{borde}}$ (desde 1.34 hasta 1.41 con EQ-MRO y hasta 1.43 con EN-MRO). Como era previsible, Q-MRO degrada la QoE de borde de celda desde 1.34 hasta 1.28 puntos de MOS. Finalmente, la Tabla 4.7 también incluye la QoE promedio de los usuarios de la red, $\overline{QoE}_u$ definida en (4.14). Se aprecia cómo $\overline{QoE_{borde}}$ también aumenta desde 4.05 hasta 4.19 con EQ-MRO, y hasta 4.22 con EN-MRO.

La Figura 4.6 ilustra la evolución de $\overline{QoE_{borde}}$ mediante una media móvil de 100 muestras. Como cabía esperar, la QoE de borde de celda se degrada con Q-MRO, mientras que EQ-MRO y EN-MRO eliminan dicha degradación. El rendimiento de EN-MRO es ligeramente mejor que el de EQ-MRO durante todo el proceso de optimización, lo que sugiere que la ANN que usa EN-MRO aproxima mejor la relación estado-acción-recompensa.

Como en el primer experimento, Q-MRO elige $HOM = 0$ dB como mejor acción, mientras que

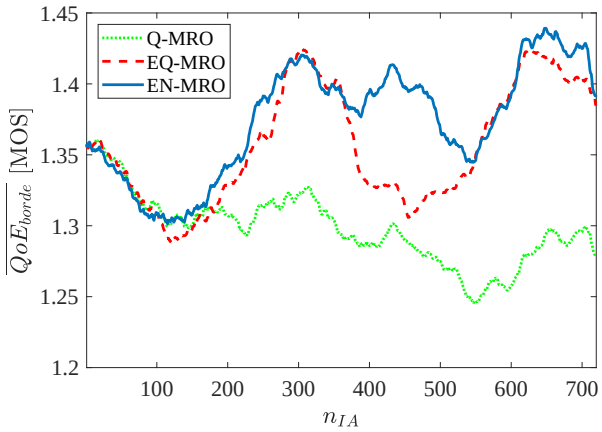


Figura 4.6: Evolución de la QoE de borde de celda (1ª fase, Experimento 2).

EQ-MRO y EN-MRO eligen retrasar el traspaso usando $HOM = 2$ dB, lo que permite alcanzar mejores rendimientos de QoE de borde de celda. Al retrasar 2 dB la ubicación en la que tiene lugar el traspaso, sobre todo cuando las celdas servidora y destino están cerca, el nivel de señal recibida de la celda destino aumenta significativamente, permitiendo el uso de esquemas de modulación y codificación en la celda destino (después del traspaso) que aumentan el *throughput* de usuario y la QoE, sin degradar demasiado el nivel de la señal recibida de la celda servidora antes del traspaso. Por tanto, el *throughput* de usuario y la QoE antes del traspaso en la celda servidora se mantienen aproximadamente constante, alcanzando mejores rendimientos de QoE.

La Tabla 4.7 muestra los resultados de la segunda fase del experimento 2, donde se completan el espacio de estados y de acciones. El espacio de estados considera diferentes velocidades de usuario, v, distancia entre emplazamientos, d, y cargas de celda vecina, l. El espacio de acciones se completa añadiendo el parámetro TTT al HOM (ya considerado en la 1ª fase). En la tabla se añade una métrica objetiva, como es el caudal medio de las conexiones $\overline{T}$, medido en Mbps, calculado según (4.15).

De acuerdo con los resultados, Q-MRO consigue reducir LHO (desde 18.7 hasta 4.05 %), pero la QoE no varía ($\overline{QoE_{borde}} = 1.45$). Por el contrario, EN-MRO reduce PP a la mitad (desde 20.65 hasta 11.44 %), mientras que LHO aumenta (desde 18.7 hasta 21.35 %). Por su parte, EQ-MRO consigue el mejor valor de SHO mejorando todas las cifras de mérito (LHO de 18.7 a 9.85 % y PP de 20.65 a 17.84 %). En esta segunda fase del experimento 2, la QoE de borde de celda aumenta levemente con ambos esquemas E-MRO ($\overline{QoE_{borde}}$ pasa de 1.45 a 1.48 con EQ-MRO, y hasta 1.52 con EN-MRO). La Tabla 4.7 incluye también el promedio $\overline{QoE}_u$ de todos los usuarios de la red,

Tabla 4.7: Rendimiento de los métodos con servicio FTP (2^{oa} fase, Experimento 2).

	Inicial	**Q-MRO**	**EQ-MRO**	**EN-MRO**
$\overline{EHO}$ [%]	0.46	1.12	0.47	0.53
$\overline{LHO}$ [%]	18.7	4.05	9.85	21.35
$\overline{PP}$ [%]	20.65	25.14	17.84	11.44
$\overline{SHO}$ [%]	60.19	69.7	71.84	66.68
$\overline{T}$ [Mbps]	-	4	4.19	4.29
$\overline{QoE_u}$[MOS]	-	4.09	4.17	4.17
$\overline{QoE_{borde}}$ [MOS]	1.45	1.45	1.48	1.52

según (4.14).

Como ocurre con la QoE de borde de celda, $\overline{QoE_u}$ también mejora con EQ-MRO y EN-MRO si se compara con Q-MRO (desde 4.09 hasta 4.17 para EQ-MRO y EN-MRO).

El análisis puede extenderse con un nuevo indicador que muestra el impacto global de los traspasos en la QoE, calculado comparando la QoE de usuario antes y después del traspaso,

$$\overline{QoE_{dif}} = \frac{1}{N_u}\sum_{u}(QoE^D(u) - QoE^A(u)) \ . \tag{4.17}$$

Este indicador es, en esencia, el término $r_{QoE_{step}}$ en la ecuación de recompensa (4.6). La Figura 4.7 muestra la evolución de $\overline{QoE_{dif}}$ en esta segunda fase del experimento 2. Se observa que, para todos los algoritmos, $\overline{QoE_{dif}}$ siempre es positivo (es decir, la QoE después del traspaso es mejor que la QoE antes del traspaso). Sin embargo, mientras Q-MRO reduce la QoE de los usuarios traspasados, EQ-MRO mantiene la misma QoE a lo largo del proceso de optimización y EN-MRO consigue aumentar la QoE de los usuarios que efectúan traspaso. Aunque ningún algoritmo empeora la QoE de los usuarios que efectúan traspaso, Q-MRO no optimiza la QoE al no incluirla en su recompensa, mientras que EQ-MRO y EN-MRO sí lo hacen.

La Tabla 4.8 detalla las mejores acciones seleccionadas por estado. Como era de esperar, y de acuerdo con los resultados obtenidos en el Experimento 1, Q-MRO selecciona los parámetros de traspaso que disparan el traspaso cuando el usuario se encuentra a la misma distancia radio de la celda servidora y de la celda destino (es decir, estados con $HOM = 0$ dB, y $TTT = 40$ ms), para 6 de los 8 estados considerados en esta segunda fase del experimento 2. Sólo en los estados 1 y 2 el traspaso se retrasa con valores de HOM más altos ($HOM = 1$ y 2 dB). EQ-MRO selecciona como mejores acciones aquellas que retrasan el disparo del traspaso (acción 5 con $HOM = 2$ dB y $TTT = 40$ ms, y acción 15 con $HOM = 2$ dB y $TTT = 256$ ms). EN-MRO retrasa el disparo del

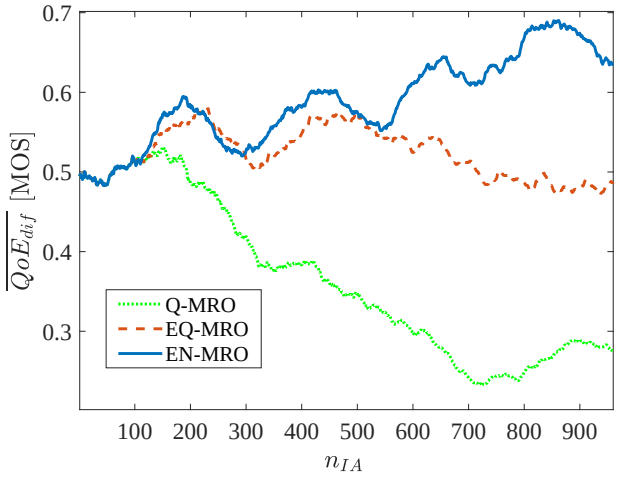


Figura 4.7: Evolución de la diferencia de QoE de borde de celda (2ª fase, Experimento 2).

Tabla 4.8: Mejores acciones seleccionadas por estado (Experimento 2).

Método/Estado	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Q-MRO	4	5	3	3	3	3	3	3
EQ-MRO	15	5	5	8	5	5	4	3
EN-MRO	14	10	9	4	15	13	4	3

traspaso aún más, ya que 5 de los 8 estados seleccionan como mejor acción aquellas con $TTT \geq 100$ ms. De estos resultados, se deduce que EQ-MRO y EN-MRO tienden a retrasar el traspaso, al igual que en la primera fase del experimento 2, de forma que se consiga una mejora significativa del *throughput* y con ella, de la QoE.

Mediante un análisis más profundo pueden compararse las mejores acciones seleccionadas por EQ-MRO y EN-MRO para los estados con menor ISD ($x \in 1, 2, 5, 6$) y con mayor ISD ($x \in 3, 4, 7, 8$). El primer grupo de estados selecciona acciones que retrasan el disparo del traspaso (aumentando HOM, TTT o ambos) en mayor medida que lo hace el segundo grupo (que eligen menores valores de HOM y/o TTT). Este comportamiento diferenciado en dos grupos refuerza la importancia considerar ISD a la hora de seleccionar los parámetros óptimos de traspaso. Un análisis similar (no presentado aquí) muestra que EN-MRO tiende a sugerir acciones con valores de TTT altos a los usuarios que se mueven a 30 km/h, la menor de las velocidades consideradas, (estados $x \in 1, 3, 5, 7$).

Tabla 4.9: Rendimiento de los métodos con distintos servicios (Experimento 3).

	Inicial	**Q-MRO**	**EQ-MRO**	**EN-MRO**
$\overline{EHO}$ [%]	1.27	0.64	0.43	0.43
$\overline{LHO}$ [%]	19.92	5.08	12.78	12.8
$\overline{PP}$ [%]	25.87	27.22	13.28	14.75
$\overline{SHO}$ [%]	52.94	67.06	73.51	72.02
$\overline{T}$ [Mbps]	-	3.59	3.64	3.69
$\overline{QoE}_u$[MOS]	-	4.08	4.12	4.12
$\overline{QoE_{borde}}$ [MOS]	2.04	2.07	2.09	2.14
$\overline{QoE_{borde}^{(VIDEO)}}$ [MOS]	2.05	2.11	2.1	2.11
$\overline{QoE_{borde}^{(FTP)}}$ [MOS]	1.58	1.55	1.59	1.66
$\overline{QoE_{borde}^{(WEB)}}$ [MOS]	2.38	2.43	2.46	2.56
$\overline{QoE}_u$ [MOS]	-	4.08	4.12	4.12
$\overline{QoE_{dif}}$ [MOS]	0.6	0.4	0.75	0.75

Experimento 3: Comparación de Q-MRO y E-MRO en escenario multiservicio

La Tabla 4.9 muestra las cifras de mérito principales obtenidas por cada método al final del proceso de optimización. Para analizar los resultados con más detalle, los valores de QoE se segregan por servicio.

Al igual que en el Experimento 2, Q-MRO consigue el mejor rendimiento en términos de LHO ($\overline{LHO} = 5.08\,\%$) en comparación con los demás algoritmos. No obstante, EQ-MRO y EN-MRO consiguen mayores tasas de traspasos realizados con éxito (67.06 % con Q-MRO, 73.51 % con EQ-MRO y 72.02 % con EN-MRO) y mejor QoE de borde de celda. Específicamente, EN-MRO obtiene los mejores indicadores de QoE, agregado y segregado por servicio ($\overline{QoE_{borde}}$, $\overline{QoE_{borde}^{(VIDEO)}}$, ...). Del mismo modo, EQ-MRO y EN-MRO superan la cifra de QoE global que consigue Q-MRO en prácticamente la misma medida ($\overline{QoE}_u = 4.08$ con Q-MRO, $\overline{QoE}_u = 4.12$ con EQ-MRO y EN-MRO). Finalmente, EQ-MRO y EN-MRO también mejoran el *throughput* medio de usuario de la red ($\overline{T} = 3.59$ Mbps con Q-MRO, mientras que $\overline{T} = 3.64$ Mbps con EQ-MRO y 3.69 Mbps con EN-MRO). Como se esperaba, Q-MRO degrada $\overline{QoE_{dif}}$, mientras que EQ-MRO y EN-MRO consiguen mejoras similares de QoE de usuario de borde de celda ($\Delta\overline{QoE_{dif}} \approx 0.35$).

Para analizar con más detalle la QoE de borde de celda, la Figura 4.8 muestra la función de distribución (abreviada por FD) de la QoE de usuarios individuales de borde de celda al final del proceso de optimización. Los usuarios se ordenan de peor a mejor QoE (de 1 a 5). La Figura 4.9 muestra una zona ampliada de la función de distribución completa dibujada en la Figura 4.8. En esta ampliación se observa que las curvas de QoE de borde de celda de Q-MRO y EQ-MRO

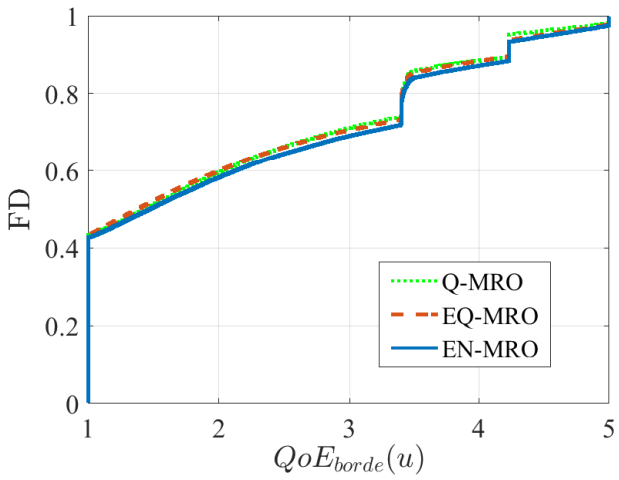


Figura 4.8: Distribución de la QoE de borde de celda al final del proceso de optimización.

quedan por encima de la curva de EN-MRO. Por tanto, ambos métodos E-MRO no sólo mejoran la QoE de borde de celda, sino también la QoE de los usuarios con valores de QoE medios/altos. Específicamente, al comparar el percentil del 70 % , Q-MRO obtiene 2.87 puntos de MOS, mientras que EQ-MRO and EN-MRO obtienen 2.93 y 3.13.

La Tabla 4.10 segrega las mejores acciones por estado y algoritmo al final del proceso de optimización. Cabe recordar que los estados 1-8 se refieren a usuarios del servicio FTP, los estados 9-16 a usuarios del servicio VIDEO y los estados 17-24 corresponden a usuarios del servicio WEB, como se muestra en la Tabla 4.2. En consecuencia, los estados $x = 1$, 9 y 17 comparten el mismo estado exceptuando el servicio (FTP, VIDEO y WEB). Un análisis detallado (no presentado aquí) muestra que la mejor configuración de los parámetros HOM y TTT por estado es similar a la obtenida en el Experimento 2, con valores positivos bajos de ambos parámetros.

Para comprobar el beneficio de seleccionar una configuración de parámetros de traspaso diferente por servicio, se analiza el valor Q final por acción y servicio que alcanza EN-MRO, obteniéndose 24 valores Q por acción (tantos como estados diferentes). Para facilitar el análisis, se promedian los valores Q de los 8 estados con el mismo servicio, con lo que se obtiene un único valor Q por acción y servicio como

$$valor\ Q^{(s)}(a) = \frac{1}{8}\sum_{x/s} valor\ Q(a,x) \tag{4.18}$$

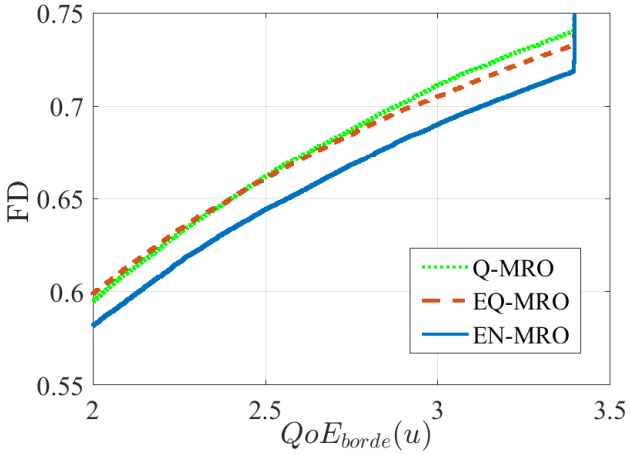


Figura 4.9: Ampliación de la QoE de borde de celda al final del proceso de optimización.

Tabla 4.10: Mejores acciones seleccionadas por estado (Experimento 3).

Método/Estado	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Q-MRO	3	3	3	2	4	3	3	3
EQ-MRO	5	10	4	3	9	10	4	4
EN-MRO	5	4	4	3	10	11	4	4
Método/Estado	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Q-MRO	3	5	3	3	3	4	3	3
EQ-MRO	5	5	3	3	5	12	3	3
EN-MRO	5	5	4	4	4	9	4	3
Método/Estado	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**
Q-MRO	3	4	3	4	3	3	3	3
EQ-MRO	5	5	5	4	10	5	5	5
EN-MRO	5	4	4	4	10	5	5	5

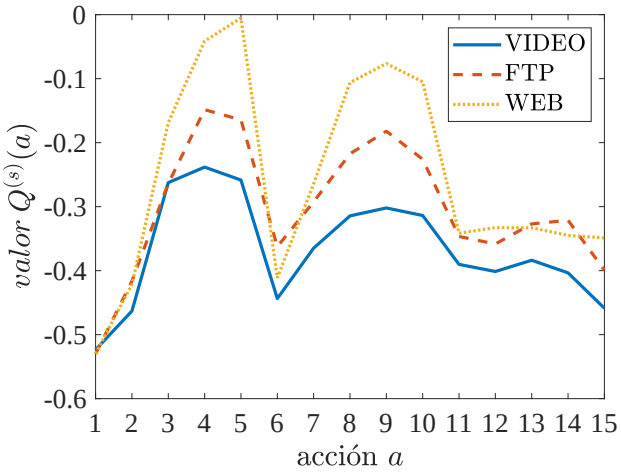


Figura 4.10: Sensibilidad del valor Q por servicio para cada acción.

donde $s \in \{FTP, VIDEO, WEB\}$. Los promedios resultantes (3 por acción) se muestran en la Figura 4.10. A primera vista, se observa que los 3 servicios alcanzan el mejor rendimiento (en media) con las acciones 3-5 (valores no negativos de HOM y el valor mínimo de TTT, como se presenta en la Tabla 4.3). Sin embargo, la mejor acción para el servicio WEB es $a = 5$ ($HOM = 2$ dB), mientras que la mejor acción para VIDEO y FTP es $a = 4$ ($HOM = 1$ dB). Además, el desplazamiento vertical entre las tres curvas (servicios) muestra que las recompensas son diferentes para diferentes servicios, lo que evidencia que las mejoras son diferentes dependiendo del servicio. Específicamente, el servicio WEB experimenta mayores recompensas que los otros dos servicios. Esto está en consonancia con los valores de QoE de borde de celda por servicio presentados en la Tabla 4.9, donde se observa que el servicio WEB es el que obtiene la mayor mejora con EQ-MRO (de 2.38 a 2.56 puntos de MOS).

4.4.4. Consideraciones de implementación

Los algoritmos EQ/EN-MRO propuestos se ejecutan periódicamente (concretamente, cada 30 s). La complejidad temporal del algoritmo Q-learning en EQ-MRO depende del tamaño de la tabla Q, determinado por el producto del número de estados y acciones, y, por tanto, es $\mathcal{O}(|XA|)$. El operador || hace referencia a la cardinalidad del conjunto en su interior. En el caso de EN-MRO, la complejidad asintótica del algoritmo de *backpropagation* para entrenar la ANN con N_m entradas, 1 salida y N_l capas ocultas es $\mathcal{O}(N_m N_l N_n N_i)$, donde N_n es el tamaño de las capas ocultas y N_i

es el número de iteraciones. Sin embargo, en un entorno real, el factor más limitante es el tiempo necesario para recoger datos de eventos de traspaso suficientes con los que entrenar la ANN, que crece linealmente con el número de estados.

En este trabajo, todas las simulaciones se llevan a cabo en un ordenador personal con un procesador de ocho núcleos a 3.6-GHz y 24 GB de memoria RAM. El algoritmo EN-MRO se implementa con el paquete de *Deep Learning* de Matlab. Con esta herramienta, entrenar una ANN con 2 capas ocultas de 4 neuronas lleva más tiempo que actualizar una matriz 2-D (tabla Q), aunque ésta sea poco profunda (0.07 segundos por entrenamiento). El tiempo medio de ejecución para una ejecución de EQ-MRO y EN-MRO es de 0.12 y 0.19 segundos, respectivamente. Específicamente, en los experimentos realizados con 24 estados y 11 horas de tiempo de red simulado (1320 intervalos de acción), EQ-MRO y EN-MRO tardan 2.6 y 4.2 minutos, respectivamente.

4.5. Conclusiones

En este capítulo, se ha propuesto un algoritmo basado en QoE que modifica adaptativamente los parámetros de traspaso para optimizar el traspaso de movilidad en una red LTE. El objetivo del algoritmo es mejorar la QoE de borde de celda y aumentar el porcentaje de traspasos realizados con éxito. El algoritmo de aprendizaje propuesto cambia el punto donde se dispara el traspaso entre adyacencias periódicamente (cada 30 segundos). Se han presentado dos variantes del algoritmo, dependiendo de la forma en que se obtiene el valor Q esperado para cada par estado-acción: bien mediante una tabla Q, bien entrenando una red neuronal artificial poco profunda de tipo perceptrón multicapa. La evaluación del algoritmo se realiza en un simulador dinámico a nivel de sistema de una red LTE que implementa un escenario macrocelular realista.

Los resultados muestran que el esquema MRO basado en QoE que utiliza la red neuronal artificial no solamente mejora la QoE de los usuarios de borde de celda en mayor medida que la variante que utiliza la tabla Q y que el algoritmo MRO clásico basado en el rendimiento del traspaso ($\overline{QoE_{borde}} = 2.14$ con EN-MRO, 2.09 y 2.07 con EQ-MRO y Q-MRO, respectivamente), sino que también supera a Q-MRO en la tasa de traspasos realizados con éxito ($\overline{SHO} = 72.02\,\%$ con EN-MRO y $\overline{SHO} = 67.6\,\%$ con Q-MRO).

Capítulo 5

Conclusiones finales

En este capítulo final, se presentan las principales conclusiones de esta tesis. En primer lugar, se resumen las contribuciones más importantes del trabajo. A continuación, se presentan las líneas de extensión. Por último, se muestra la lista de publicaciones generadas durante el desarrollo de esta tesis.

5.1. Principales contribuciones

La generalización del uso de técnicas de automatización de redes celulares se debe, en gran parte, al crecimiento exponencial de la demanda de tráfico asociada a la aparición de nuevos servicios diferentes a las tradicionales llamadas de voz. Las técnicas de gestión de redes más modernas se centran en la opinión subjetiva del usuario (QoE). Cuando las técnicas automáticas de optimización tienen como objetivo aliviar los problemas de congestión persistentes, puede ocurrir que no se estén aliviando, o incluso se estén empeorando, los problemas de QoE. Todo ello hace necesario la búsqueda de nuevas soluciones a dichos problemas de QoE. En esta tesis se ha realizado un profundo trabajo de búsqueda, análisis y experimentación de soluciones para problemas de QoE en redes LTE.

El trabajo se ha iniciado con un análisis de los cambios sufridos en los últimos años en la gestión de las redes celulares, presentados en el Capítulo 2. Se ha estudiado la diversidad de servicios en las redes celulares, precisando los modelos de servicio utilizados en esta tesis. En segundo lugar,

se han revisado las técnicas de reparto de carga usadas en redes 2G/3G, que son las precursoras de las técnicas de optimización propuestas en este trabajo. En tercer lugar, se han presentado las técnicas SON, con el caso de uso de autooptimización, donde se engloba este trabajo. En cuarto lugar, se han presentado las técnicas de inteligencia artificial aplicadas en la gestión de las redes celulares, prestando especial atención a las utilizadas en este trabajo, como son la lógica difusa, el aprendizaje por refuerzo y las redes neuronales. En quinto lugar, se ha explicado cómo se realiza la gestión de la experiencia de usuario, mediante el modelado, la medida y el control de la QoE. Por último, se han revisado las principales técnicas de verificación del rendimiento en redes celulares, que se emplean para validar los métodos propuestos en esta tesis.

Las dos primeras contribuciones de este trabajo son los métodos de reparto de tráfico con criterios de QoE mediante el traspaso, presentados en el Capítulo 3. En la formulación del problema, se ha puesto de manifiesto que los algoritmos clásicos de balance de carga no siempre equilibran la QoE. Para contrarrestar estas limitaciones, se han diseñado dos algoritmos de reparto de tráfico para redes LTE macrocelulares. Los algoritmos propuestos alivian los problemas de equilibrio y degradación de QoE distribuyendo la demanda de tráfico entre celdas vecinas. Para ello, los algoritmos descritos modifican las áreas de servicio de las celdas ajustando los márgenes de traspaso por celda, o por celda y servicio, de forma que parte del tráfico de las celdas con peor QoE se desplace hacia celdas vecinas con mejor QoE. Por simplicidad, el primer método de equilibrio de la QoE de celda se ha diseñado mediante controladores de lógica difusa que replican el conocimiento de un experto. Dentro de este primer método, la estrategia que ajusta los ajustes de los márgenes de traspaso por celda y servicio ha presentado los mejores resultados si se desean minimizar los desequilibrios de QoE de celda con diferentes mezclas de servicios. El segundo algoritmo, presentado como alternativa, implementa un algoritmo de ascenso de gradiente sobre un modelo analítico de rendimiento del sistema calibrado con en medidas, que asegura que los cambios de márgenes de traspaso siempre aumenten la QoE global del sistema. Ambos métodos de reparto de tráfico se han validado en un simulador de una red de macroceldas LTE modificado específicamente para esta tesis. Mediante simulaciones, se ha analizado el impacto de los algoritmos diseñados sobre la QoE ofrecida por el sistema. Los resultados han demostrado la robustez de las técnicas propuestas, que equilibran o mejoran la QoE de la red para diferentes condiciones de tráfico.

Los métodos anteriores utilizan el mecanismo de traspaso para redistribuir el tráfico en la red de forma que se equilibre, o se mejore, la QoE en situaciones de sobrecarga. En la tercera contribución de esta tesis, presentada en el Capítulo 4, se plantea la modificación de los parámetros de traspaso para redefinir el borde de celda, mejorando el proceso de traspaso en situaciones sin sobrecarga. Los primeros experimentos han puesto de manifiesto que los algoritmos clásicos de mejora de la robustez de la movilidad tienden a adelantar el traspaso para evitar los traspasos realizados demasiado tarde,

haciendo que se degrade la QoE de los usuarios de borde de celda. Para solventar estas limitaciones, se propone por primera vez la inclusión de criterios de QoE, junto a la tasa de traspasos con éxito, para controlar el traspaso. Con este fin, se presenta un algoritmo de traspaso adaptativo que encuentra la configuración óptima de los márgenes y temporizadores de traspaso por adyacencia sin disponer de un modelo del sistema, utilizando aprendizaje autónomo. Como principal novedad, el algoritmo propuesto incluye la evaluación de la QoE en una ventana temporal alrededor del evento de traspaso, para aislar el efecto del traspaso de otros posibles efectos en la QoE. En una primera variante, se extiende un algoritmo de Q-learning clásico para considerar también la QoE de los usuarios que realizan traspaso. En la segunda variante, se mejora la técnica de Q-learning con una red neuronal no profunda para asegurar una aprendizaje más eficaz. Las simulaciones han demostrado que se puede mejorar la QoE de los usuarios que realizan traspaso al mismo tiempo que se mejora la tasa de traspasos con éxito. De las dos variantes del algoritmo, aquella que combina Q-learning con una ANN ofrece mejores resultados de QoE para los usuarios de borde de celda, sobre todo para los usuarios de servicios de datos con características de búfer completo (p. ej., transferencia de archivos).

Los métodos de optimización de redes celulares propuestos en esta tesis utilizan información disponible en el sistema de gestión de red, y, por ello, están concebidos como soluciones centralizadas. Además, por su baja carga computacional, pueden integrarse en las herramientas de planificación y optimización automática que ofrecen los fabricantes para el sistema de soporte a las operaciones, sin plantear problemas de escalabilidad.

5.2. Líneas futuras

Tras completar este trabajo, se plantean las siguientes líneas futuras de investigación. Algunas de ellas son mejoras de los algoritmos propuestos en esta tesis, mientras que otras son extensiones a otros problemas que plantean las redes celulares.

- *Desarrollo de un algoritmo de balance de carga proactivo.* Uno de los principales problemas del SON en la actualidad es la falta de proactividad de sus métodos. La optimización proactiva de la calidad de experiencia requiere establecer un modelo de predicción de la QoE de los servicios de una celda a medio y largo plazo con medidas históricas de tráfico y rendimiento por servicio. Una vez identificadas las medidas que mejor reflejan la QoE de cada servicio en la interfaz radio, se ha de construir un modelo de predicción de la QoE colectiva a largo plazo (meses) o a medio plazo (minutos) e integrarlo en el algoritmo de balance de QoE. Con dichos modelos, se persigue mejorar la capacidad de reacción de los métodos planificación y optimización

de red. Para la predicción a largo plazo, se suelen utilizar técnicas clásicas de análisis de series temporales, que han venido utilizándose con éxito para predecir la demanda agregada de tráfico en las interfaces de la red troncal. Las técnicas tradicionales de análisis de series temporales suelen convertir primero la serie de datos de la variable a predecir (en este caso, el descriptor de la QoE de celda) en estacionaria, detectando y eliminando las componentes de tendencia y estacionalidad. Para la componente residual, que modela la dependencia a corto plazo, se suele ajustar un modelo lineal autoregresivo de media móvil (ARMA). Ejemplos de estas técnicas son los métodos de predicción de *Holt-Winters* y *Auto Regressive Integrative Moving Average* (ARIMA), para series con varianza constante, o *Generalized AutoRegressive Conditional Heteroscedasticity* (GARCH), para series con cambios de varianza [132]. Más recientemente, se ha propuesto el uso de técnicas de aprendizaje autónomo para la predicción de rendimiento, formulando el problema como un problema de regresión [133].

- *Extensión a los nuevos tipos de servicios propios de los sistemas 5G*: Los servicios considerados en los experimentos llevados a cabo en este tesis se engloban dentro de la categoría (*Enhanced Mobile Broadband*, eMBB). En los futuros sistemas 5G, este grupo de servicios coexistirá con servicios de comunicaciones entre máquinas (*Massive Machine Type Communications*, mMTC) y servicios de misión crítica (*Ultra Reliable and Low Latency Communications*, URLLC). Los servicios mMTC tienen la peculiaridad de necesitar un ancho de banda muy pequeño, ya que se transmiten mensajes cortos y muy espaciados en el tiempo. Los servicios URLLC tampoco requieren anchos de banda muy elevados, pero, en cambio, tienen altas exigencias de retardo y de fiabilidad. Estas nuevas características requerirá cambios sustanciales en la concepción, diseño y desarrollo de los algoritmos propuestos en esta tesis. Como punto de partida, deben desarrollarse los modelos de tráfico y QoE de estos servicios.

- *Adaptación de los algoritmos de ajuste de parámetros para su funcionamiento en tiempo real.* Los algoritmos de reparto de tráfico propuestos en esta tesis se han centrado en la resolución de problemas de QoE persistentes, solventados de forma progresiva ejecutando lazos de optimización cada hora. Sin embargo, en la práctica, si se dispone de la información necesaria en intervalos más cortos de tiempo (p. ej., minutos) se puede reducir la periodicidad de estos algoritmos para resolver problemas de QoE momentáneos. Se trata de llevar a cabo un ajuste de los parámetros en tiempo real, de modo que se adapten rápidamente a cualquier cambio en las condiciones de red. La principal dificultad es la pérdida de fiabilidad de los ajustes al disponer de un menor número de muestras a partir de las que tomar las decisiones. Para ello, habría que asegurar la robustez de las estadísticas de QoE estableciendo intervalos de confianza en los indicadores o directamente no llevar a cabo el proceso de optimización cuando el número de medidas sea insuficiente.

- *Mejora del proceso de aprendizaje mediante aprendizaje profundo.* La segunda parte de esta

tesis se ha centrado en optimizar el traspaso de movilidad aumentando la tasa de traspasos realizados con éxito y la QoE de los usuarios de borde de celda. Para ello, se ha utilizado aprendizaje por refuerzo mediante Q-learning, manteniéndose el juego de estados y acciones discretas. Una posible mejora de este trabajo es el uso de la técnica de aprendizaje por refuerzo *Deep-Q-Networks* (DQN), basada en Q-learning y redes neuronales profundas, para hacer continuo el espacio de estados y acciones, y ser capaz de aprender cuál es la mejor acción para cada estado sin haberlas probado todas.

- *Extensión del análisis a otros escenarios típicos de los sistemas 5G.* Se plantea aplicar las técnicas de reparto de tráfico y de optimización del traspaso de movilidad a otros escenarios que combinen celdas pequeñas en las bandas de milimétricas. Las celdas pequeñas tienen un área de servicio mucho menor que las macroceldas, por lo tanto, es probable que necesiten configurar un valor de histéresis menor y que sea necesario limitar la desviación máxima de los márgenes de traspaso o imponer un valor máximo del TTT entre celdas pequeñas y entre una celda pequeña y una macrocelda. Además, el hecho de incluir celdas pequeñas en un escenario de macroceldas aumenta considerablemente el número de adyacencias en la red, lo que va a suponer un aumento del coste computacional de los algoritmos. En este contexto, adquieren importancia aspectos prácticos como la necesidad de restringir el proceso de ajuste a las adyacencias más relevante de cada celda. Sería interesante evaluar la pérdida de eficacia que implica este tipo de restricciones en un entorno de red heterogénea real. Igualmente, queda por estudiar si el diferente comportamiento de los fenómenos de desvanecimiento del canal radio en la banda de milimétricas requiere una configuración de parámetros de traspaso distinta respecto a la configuración en las bandas actuales.

5.3. Publicaciones

El trabajo presentado en esta tesis ha generado los siguientes resultados de investigación:

Artículos

I M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, and C. Gijón, "A QoE-Driven Traffic Steering Algorithm for LTE Networks ", IEEE Transactions on Vehicular Technology, vol. 68, no. 11, pp. 11271-11282, Nov. 2019.

II M. L. Marí-Altozano, M. Toril, S. Luna-Ramírez and C. Gijón, "A Self-Tuning Algorithm for Optimal QoE-Driven Traffic Steering in LTE," IEEE Access, vol. 8, pp. 156707-156717, Sep.

2020.

III M. L. Marí-Altozano, S. Mwanje, S. Luna-Ramírez, M. Toril, H. Sanneck and C. Gijón, "A service-centric Q-learning algorithm for mobility robustness optimization in LTE ", IEEE Transactions on Network and Service Management, Apr. 2021.

Aportaciones a congresos y reuniones científicas

IV M.L. Marí-Altozano, S. Luna-Ramírez, M. Toril, "Limitaciones del equilibrio de carga para la mejora de la calidad de experiencia en redes LTE", XXXII Simposio de la Unión Científica Internacional de Radio (URSI) 2017, Cartagena(España), Sep. 2017.

V M.L. Marí-Altozano, S. Luna-Ramírez, M. Toril, C. Gijón, "Algoritmo de control de carga basado en calidad de experiencia en redes LTE", XXXIII Simposio de la Unión Científica Internacional de Radio (URSI) 2018, Granada(España), Sep. 2018.

VI M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, "Load balance performance analysis with a quality of experience perspective in LTE networks", CA15104 IRACON, Graz(Austria), Sep. 2018.

VII M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, "A QoE-driven Traffic Steering algorithm for LTE networks", CA15104 IRACON, Dublin(Ireland), Jan. 2019.

VIII M.L. Marí-Altozano, S. Luna-Ramírez, M. Toril, C. Gijón, "Optimización de la calidad de experiencia en redes LTE mediante el reparto de tráfico", XXXIV Simposio de la Unión Científica Internacional de Radio (URSI) 2019, Sevilla(España), Sep. 2019.

IX M.L. Marí-Altozano, S. Mwanje, S. Luna-Ramírez, M. Toril, C. Gijón, "Una visión basada en QoE para algoritmo MRO en redes LTE", XXXV Simposio de la Unión Científica Internacional de Radio (URSI) 2020, Málaga(España), Sep. 2020.

El estudio de los métodos de reparto de tráfico con QoE presentado en el Capítulo 3, se describe en [I, IV, V, VI, VII]. En [IV] y [VI] se describe el análisis preliminar de las limitaciones del balance de carga en LTE en términos de QoE. En [I, V y VII], se presentan los algoritmos de reparto de tráfico para equilibrar la QoE media de celda en redes macrocelulares LTE. En [II, VIII], se presenta el algoritmo analítico de optimización de la QoE de usuario para un servicio *full-buffer*. Finalmente, en [III, IX] se presenta el algoritmo MRO para la optimización de la QoE de los usuarios de borde de celda y la tasa de traspasos exitosos descrito en el Capítulo 4.

Estas contribuciones se han llevado a cabo en el marco de diversos proyectos de investigación:

- *Métodos de planificación y optimización de la calidad de experiencia en redes B4G* (TEC2015-69982-R), financiado por el Ministerio de Economía y Competitividad, 2015 [I, IV, V, VI, VII, VIII].
- *E2E-aware Optimizations and advancements for the Network Edge of 5G New Radio* (ONE5G), financiado por la Comisión Europea bajo la iniciativa de la UE "Horizonte 2020", 2018 [I, VII].
- *Métodos de planificación automática de redes 5G virtualizadas* (RTI2018-099148-B-I00), financiado por el Ministerio de Ciencia, Innovación y Universidades [III, II,IX].
- *Predicción de la calidad de experiencia en redes 5G* (UMA18-FEDERJA-256), financiado por la Consejería de Economía, Conocimiento, Empresas y Universidad de la Junta de Andalucía [III, IX].

Además, el experimento de equilibrio de QoE realizado en la red piloto del apartado 3.4.1, resultado de la investigación en I, se presentó en una reunión científica presencial del proyecto ONE5G en el Instituto Fraunhofer Heinrich Hertz de Berlín.

Otras aportaciones

Las herramientas, habilidades y conocimientos desarrollados en esta tesis han sentado la base de otros estudios, que han dado lugar a las siguientes publicaciones:

X C.Gijón, M. Toril, S. Luna-Ramírez, M.L. Marí-Altozano, "Un nuevo criterio basado en calidad de experiencia para el balance de carga en redes LTE ", URSI 2018, Granada(España), Sep. 2018.

XI C.Gijón, M. Toril, S. Luna-Ramírez, M.L. Marí-Altozano, "A data-driven user steering algorithm for optimizing user experience in multi-tier LTE networks", Cost Action CA15104 IRACON, Dublin(Ireland), Jan. 2019.

XII C.Gijón, M. Toril, S. Luna-Ramírez, M.L. Marí-Altozano, "Mejora de la calidad de experiencia en redes LTE multi-portadora", URSI 2019, Sevilla(España), Sep. 2019.

XIII C. Gijón, M. Toril, S. Luna-Ramírez, M.L. Marí-Altozano, "A data-driven traffic steering algorithm for optimizing user experience in multi-tier LTE networks", IEEE Transactions on Vehicular Technology, vol. 68, no. 10, pp. 9414-9424, Oct. 2019.

XIV C.Gijón, M. Toril, S. Luna-Ramírez, J. L. Bejarano-Luque, M.L. Marí-Altozano, "Estimación de capacidad en redes LTE mediante aprendizaje supervisado", URSI 2020, Málaga(España), Sep. 2020.

XV C. Gijón, M. Toril, S. Luna-Ramírez, J. L. Bejarano-Luque, M.L. Marí-Altozano, "Estimating Pole Capacity from Radio Network Performance Statistics by Supervised Learning", IEEE Transactions on Network and Service Management, vol. 17, no. 4, pp. 2090-2101, Dec. 2020.

XVI C. Gijón, M. Toril, S. Luna-Ramírez, J. M. Ruiz-Avilés, M.L. Marí-Altozano, A Comparative Study of Long-Term Traffic Forecasting on a Cell Basis in a Live Cellular Network, IEEE Access, en revisión.

En X se propone un nuevo criterio para efectuar el traspaso entre celdas vecinas basado en QoE. La validación del criterio propuesto se lleva a cabo en la misma red piloto que la utilizada en el apartado 3.4.1. Las referencias XI, XII y XIII , que presentan un indicador para dirigir los traspasos intersistema en una red LTE multiportadora para optimizar la QoE, están desarrolladas y validadas en el mismo simulador dinámico de red LTE a nivel de sistema que el utilizado y desarrollado en esta tesis. En XIV, XV y XVI se utilizan distintas técnicas de aprendizaje autónomo para predecir el tráfico a largo plazo en una red celular y para estimar su capacidad máxima.

Apéndice A

Anexo: Herramienta de simulación

En este anexo, se presenta la herramienta de simulación usada para validar las contribuciones de este trabajo. Se trata de un simulador de red LTE a nivel de sistema, implementado en Matlab R2011a por su eficiencia en el manejo de matrices [101]. Este simulador permite al usuario configurar los parámetros básicos del escenario simulado a través de un fichero de configuración, sin necesidad de conocer exhaustivamente los procesos y métodos internos implementados para modelar las principales funcionalidades de red.

En las siguientes secciones, se describen las funcionalidades más importantes del simulador. En la primera sección, se presenta la estructura básica del simulador. En las siguientes, se explican las funcionalidades de las capas física, de enlace y de red.

A.1. Estructura básica

Esta sección presenta la estructura de bloques básica de la que consta la herramienta de simulación, para, después, repasar brevemente los principales bloques del simulador. Estos bloques principales son el escenario de simulación, el modelo de movilidad, el modelo de distribución espacial de tráfico y el modelo de servicio.

Como se observa en la Figura A.1, tras la configuración del escenario, la primera fase de la simulación es la inicialización de los parámetros de simulación principales, definiendo el comportamiento de las funciones de simulador. En este punto, se generan el escenario a simular y la distribución

Figura A.1: Diagrama de bloques del simulador[101].

inicial de usuarios (*warm-up*), que permite obtener estadísticas de rendimiento fiables desde las primeras iteraciones de la simulación. La siguiente función calcula las matrices de pérdidas de propagación. Esta función calcula la potencia que recibe cada punto del escenario desde todas las estaciones base del escenario. Para el cálculo de estas pérdidas, el simulador incluye un modelo de pérdidas de trayecto (*PathLoss*, PL), un modelo de desvanecimiento lento y otro de desvanecimiento rápido. Esta información se utiliza para calcular la interferencia que experimenta cada usuario. A partir de esta información, se obtiene el valor de SINR (*Signal to Interference and Noise Ratio*) para cada usuario.

Completados los cálculos de propagación, se ejecutan las funciones de gestión de recursos radio. En el nivel de enlace, el simulador incluye las funciones de planificación de recursos (*scheduling*) y adaptación del enlace (modulación y codificación adaptativa). La función de planificación de recursos asigna los recursos radio disponibles a los usuarios dependiendo del servicio solicitado y las condiciones de canal experimentadas, deducidas a partir del valor del indicador CQI (*Channel Quality Indicator*). También basándose en el CQI, la función de adaptación al enlace selecciona el esquema de modulación y codificación más apropiado dependiendo de las condiciones de propagación de cada usuario. A nivel de red, el simulador incluye funciones de control de admisión y traspaso.

El escenario de simulación utilizado en este trabajo, desarrollado como parte del mismo, se explica en la Sección 3.4.1. Dicho escenario consta de 108 macroceldas (36 emplazamientos con

Tabla A.1: Parámetros de simulación.

Resolución temporal	10 TTI (10 ms)
Modelo de propagación	Pérdidas de trayecto COST 231 Hata, desvanecimiento lento (lognormal σ = 8 dB, d_{corr} = 20 m), desvanecimiento rápido (modelo ETU)
Modelo de estación base	Antenas Tri-sectoriales, MIMO 2x2, BW = 5 MHz (25 PRB), $f_{portadora}$ = 2 GHz, PIRE$_{max}$= 68 dBm.
Planificador	Exponencial clásico/proportional fair [105]
Adaptación al enlace	Basado en CQI

celdas trisectoriales) que cubren un área de 60 km^2 cuyos principales parámetros se presentaron en el apartado (3.4.1), y se repiten en la Tabla A.1.

Como puede observarse en la tabla, los usuarios (que simulan ser peatones) se mueven a una velocidad fija de 3 km/h y siguen un trayecto rectilíneo con dirección aleatoria. El modelo de distribución espacial del tráfico original en [101] se ha modificado para hacer que los usuarios se distribuyan de forma uniforme dentro del área de servicio de cada celda. Por último, el simulador implementa 4 servicios diferentes: un servicio de voz sobre IP (VoIP), un servicio de transmisión progresiva de vídeo (VIDEO), un servicio de descarga de ficheros (FTP) y un servicio de descarga de varias páginas web con un tiempo aleatorio de lectura entre descargas (WEB). Los detalles del modelo de cada uno de los servicios se han presentado en la Sección 2.1.1.

A.2. Capa física

En esta sección, se explican los aspectos más importantes del modelo de capa física del sistema de red LTE, que incluye el modelo de propagación de la señal radio, el modelo de ruido y el modelo de interferencia.

A.2.1. Modelo de propagación

Modelo de pérdidas de trayecto

Para reducir el coste computacional, los cálculos de propagación se realizan a partir de un conjunto de matrices pregeneradas, que integran las diferentes componentes de las pérdidas de

propagación (trayecto y desvanecimiento lento y rápido). Para definir la matriz de propagación pregenerada, se divide el escenario en una rejilla, cuya resolución es de 50 m x 50 m. Para conocer los valores de las pérdidas de propagación que está experimentando un usuario, solo es necesario leer la posición de la matriz que corresponde a la posición ocupada por el usuario en el escenario con relación a cada estación base e interpolar el valor de pérdida en dicha posición a partir de otros valores de la matriz precalculada en posiciones cercanas. La interpolación se realiza dependiendo de la posición relativa del usuario en la rejilla con respecto a la posición de los valores conocidos de pérdidas de propagación.

Las matrices de propagación incluyen el cálculo de las pérdidas del trayecto y de desvanecimiento lento. El modelo de propagación radio es la extensión COST 231 del modelo Hata [134]. Este modelo es aplicable a frecuencias dentro del rango 1500-2000 MHz. Las alturas de las estaciones base se toman de la red real en la que está inspirado el escenario, oscilando entre los 11 y los 64 m, mientras que la altura efectiva de los terminales móviles se ha establecido en 1.5 m. Con estas suposiciones y configurando la frecuencia de portadora a 2 GHz, la expresión para el cálculo de las pérdidas de propagación en función de la distancia es

$$L\ [dB] = 134.79 + 34.22log(d)\ , \tag{A.1}$$

donde d es la distancia en km entre el terminal móvil y la estación base a la que está conectado.

Modelo de desvanecimiento lento.

La matriz de propagación incluye también un modelo de desvanecimiento lento basado en el hecho de que la media local de la envolvente de la señal radio se puede modelar mediante una distribución lognormal en la que la media local en dB es una variable aleatoria Gaussiana. La desviación estándar de la distribución depende del entorno considerado. Un valor típico para una red urbana de macroceldas es de 8 dB [135].

Modelo de canal radio

El canal radio se modela como un filtro lineal que varía con el tiempo [136]. Como todo filtro, se puede caracterizar en el dominio del tiempo mediante su respuesta al impulso $h(\tau, t)$, donde τ representa el retardo de cada trayecto en h, y la amplitud de cada trayecto varía con el tiempo t.

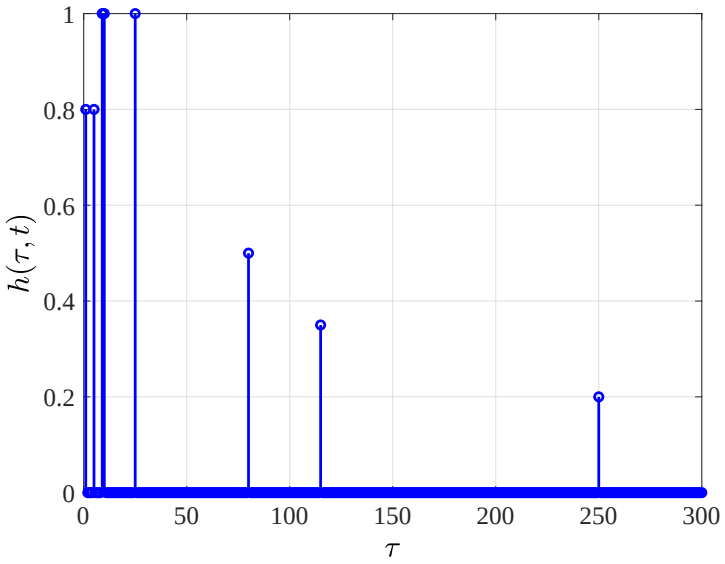


Figura A.2: Respuesta al impulso [138]

Un ejemplo de respuesta al impulso para un canal ETU (*Extended Typical Urban*) [137] se muestra en la Figura A.2

Complementariamente, el canal se puede caracterizar por la función de transferencia variante en el tiempo, $H(f,t)$, que se relaciona con la respuesta al impulso del filtro mediante la transformada de Fourier respecto de la variable de retardo. Cuando el comportamiento del canal es variable con el tiempo de forma aleatoria, las funciones del canal explicadas se convierten en procesos estocásticos.

Para simular móviles con distintas velocidades, resulta adecuado sustituir la realización temporal de la atenuación causada por el desvanecimiento rápido por variables espaciales que modelen cómo el canal varía dependiendo de la posición actual del usuario en cada iteración de la simulación. Para ello, se genera una rejilla espacial de desvanecimiento de canal, que modela la variación del desvanecimiento rápido en el espacio. Esta rejilla ofrece respuestas de canal para cada posición física en el escenario simulado independientemente de la velocidad del móvil. De la misma forma, permite que los usuarios pueden permanecer estáticos en su posición durante un periodo de tiempo, y después comenzar a moverse a cualquier velocidad. Todo ello permite simular patrones de movilidad urbanos, en los que los vehículos se mueven por la ciudad, o peatones que andan por la ciudad o dentro de un edificio.

Para modelar el desvanecimiento rápido de la señal radio, también se utiliza una matriz prege-

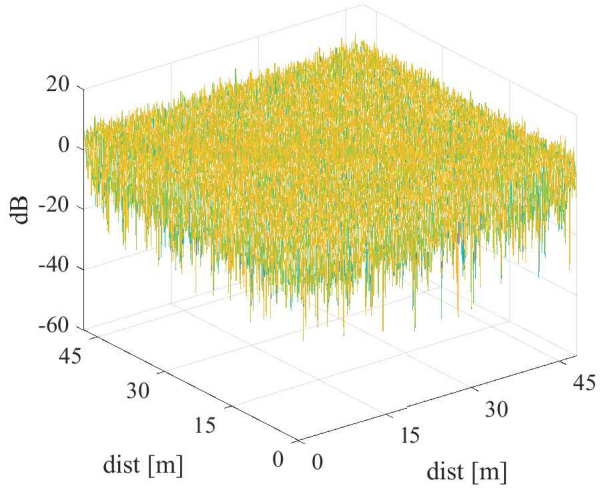


Figura A.3: Respuesta al impulso bidimensional para un modelo de canal ETU (rejilla de 48 m x 48 m).

nerada. A diferencia del caso de la matriz de propagación, la resolución de la rejilla ha de ser mucho menor (15 cm) para poder apreciar el fenómeno del desvanecimiento rápido, lo que hace inviable, por razones de memoria, construir una rejilla que abarque todo el escenario. De forma alternativa, se pregenera una matriz que cubre 48 m x 48 m, ilustrada en Figura A.3, que se reutiliza a lo largo y ancho de todo el escenario hasta cubrirlo por completo. Para simular el desvanecimiento rápido se utiliza el modelo descrito por el estándar *Extended Typical Urban* [137].

A.2.2. Modelo de ruido

El modelo de ruido asume una temperatura de referencia de 293 K, una densidad espectral de potencia de ruido de -174 dBm/Hz, un ancho de banda equivalente de ruido del receptor de 1 PRB igual a 180 kHz y una cifra de ruido de los receptores del terminal móvil y la estación base de 9 y 5 dB, respectivamente. El resultado es un nivel de ruido de -112.44 dBm por PRB.

A.2.3. Modelo de interferencia

En el cálculo del interferencia, se supone que la versión básica de LTE (Release 8) la interferencia intracelda es despreciable, porque el planificador asigna diferentes frecuencias y ranuras temporales

a cada usuario. Por tanto, el cálculo de interferencia intercelda requiere conocer el nivel de señal recibido por cada usuario desde cada una de las celdas interferentes. En dicho cálculo, no se tiene en cuenta la respuesta del canal, sino únicamente las pérdidas de trayecto y el desvanecimiento lento. De esta forma, la interferencia cocanal se calcula mediante la fórmula

$$I = \sum P(j)L(j)$$

donde, en el enlace descendente, $P(i)$ y $P(j)$ son los niveles de señal recibidos de las estaciones base servidora i y vecina j, calculados a partir de las pérdidas de trayecto y el desvanecimiento lento, el diagrama de radiación de la antena y la posición del terminal móvil, y $L(j)$ es la carga media de celda vecina j durante el intervalo de resolución temporal configurado en las simulaciones (10 ms).

A.3. Capa de enlace

En esta sección se presentan las principales funcionalidades de la capa de enlace en el simulador, que son el cálculo de la relación señal a interferente (*Signal to Interference plus Noise Ratio*, SINR), el esquema HARQ (*Hybrid Automatic Repeat reQuest*), la adaptación del enlace y la asignación de recursos.

A.3.1. Cálculo de la SINR

La SINR es una medida que representa la calidad del enlace que experimenta el usuario. Para calcular la SINR en el simulador, en primer lugar, es necesario calcular la interferencia que experimenta cada usuario. Se supone que en LTE la interferencia intracelda es despreciable porque el planificador asigna las diferentes frecuencias y ranuras temporales a cada usuario. El cálculo de interferencia intercelda requiere conocer la señal recibida por cada usuario desde cada una de las celdas interferentes. Para calcular la interferencia de cada estación base recibida por el terminal, no se tiene en cuenta la respuesta del canal, sino únicamente las pérdidas de propagación y el desvanecimiento lento. En el simulador, la SINR se calcula en cada paso de simulación para cada usuario en su ubicación concreta dentro del escenario. Este cálculo se realiza a partir de las medidas de potencia de la señal recibida de la celda servidora y de las medidas de interferencia intercelda que ofrece el simulador.

A.3.2. Esquema HARQ

El esquema HARQ es una función de nivel de enlace que permite efectuar retransmisiones directamente en la capa física o la capa MAC (*Medium Access Control*) en LTE. De [139], se extrae un modelo sencillo capaz de predecir con precisión la ganancia de HARQ en la capa física. Cuando tiene lugar una retransmisión HARQ, se espera una mejora de la BLER (*Block Error Rate*). El resultado es que las curvas de BLER basadas en el modelo de canal AWGN (*Additive White Gaussian Noise*) se desplazan ofreciendo ganancia de relación señal al ruido (*Signal to Noise Ratio*, SNR) gracias al uso de HARQ. Por lo tanto, la nueva SINR se puede calcular como sigue:

$$SINR(i) = SINR + SINR_{gain}(i) \ , \tag{A.2}$$

donde i denota el índice de la retransmisión. El valor de $SINR_{gain}$, obtenida de tablas recogidas en [139], depende del valor de redundancia para el índice de retransmisión i y del MCS (*Mobile and Coding Scheme*) utilizado.

Obtenido el valor de BLER y conocido el MCS utilizado en la transmisión, se puede calcular la tasa de transmisión, Ti, como sigue

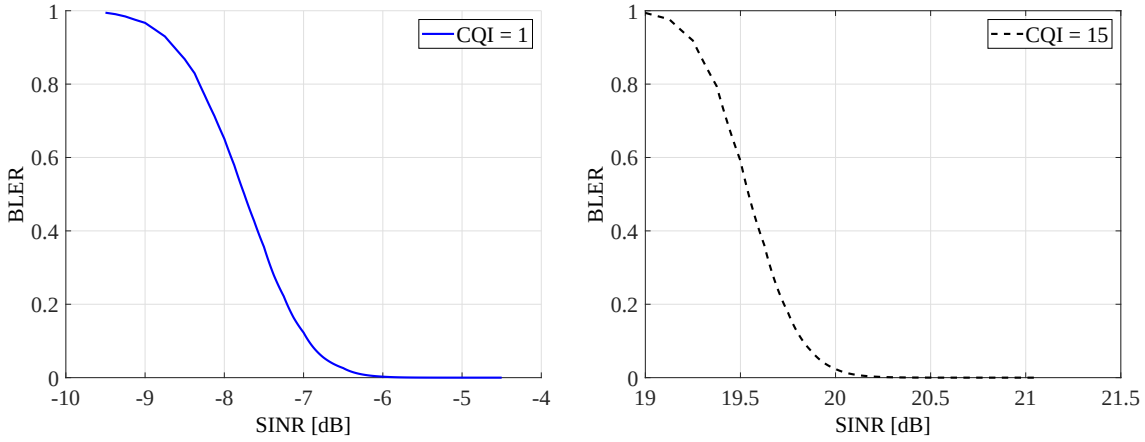
$$Ti = (1 - BLER(SINR_i))\frac{D_i}{TTI}, \tag{A.3}$$

donde TTI es el intervalo de tiempo de transmisión, Di es la carga útil por bloque en bits [140], cuyo valor depende del MCS utilizado por el usuario en ese intervalo de tiempo, y $BLER(SINR_i)$ es el valor de BLER obtenido para la SINR efectiva.

La Figura A.4 muestra la relación SINR-BLER para el mínimo y máximo valor de CQI posible, esto es, 1 y 15. Dichas curvas se han obtenido mediante el simulador de nivel de enlace Vienna [141]para un sistema LTE de ancho de banda 1.4 MHz. El modelo de canal es AWGN y la velocidad del móvil es de 3 km/h.

A.3.3. Adaptación del enlace

Antes de explicar la adaptación al enlace, es necesario explicar el parámetro CQI estandarizado por el 3GPP. Este indicador representa la calidad de la conexión en una subbanda del espectro. La



(a) SINR-BLER para CQI 1.

(b) SINR-BLER para CQI 15.

Figura A.4: Curvas SINR-BLER para LTE con 1.4 MHz.

resolución del CQI es de 4 bits, pero se puede transmitir un valor diferencial de CQI para reducir la sobrecarga de señalización. Por tanto solamente hay un subconjunto posible de MCS que se corresponden con un valor de CQI [142]. En el simulador, se recoge el valor del CQI de cada usuario cada paso de simulación (10 ms). Según de los valores de CQI, el módulo de adaptación al enlace selecciona la modulación y esquema de codificación más apropiado para transmitir la información en el *Physical Downlink Shared Channel* (PDSCH) dependiendo de las condiciones de propagación del entorno. Para cuantificar la calidad del enlace de cada usuario en cada subbanda del espectro se usa el valor de CQI. Si se necesita que el valor de BLER experimentado esté por debajo de un umbral específico de cada servicio, se puede establecer una relación SINR-CQI que permite seleccionar el MCS más apropiado a partir de un valor de SINR [143]. El 3GPP define un campo para el MCS de 5 bits en el enlace descendente de control. Esto ofrece una gran variedad de posibles MCS. Por simplicidad, el simulador sólo incluye un conjunto de MCS similar al de valores de CQI. Por lo tanto, a partir del valor de SINR efectivo se calcula el valor de CQI y así determinar el MCS para el siguiente paso de simulación. La relación SINR-CQI-MCS que implementa el simulador se muestra en la Tabla A.2.

A.3.4. Asignación de recursos

La planificación dinámica de recursos la realiza el subsistema comúnmente denominado *scheduler*. Por simplicidad, dicha planificación suele descomponerse en una planificación en el dominio del

Tabla A.2: Relación SINR-CQI-MCS en el simulador.

SIR [dB]	**CQI**	**MCS**
-15	1	1
-14	1	1
-13	1	1
-12	1	1
-11	1	1
-10	1	1
-9	1	1
-8	1	1
-7	1	1
-6	1	1
-5	2	1
-4	2	1
-3	3	2
-2	3	3
-1	4	3
0	4	4
1	5	4
2	5	5
3	6	5
4	6	6
5	7	6
6	7	7
7	8	8
8	8	8
9	9	8
10	10	8
11	10	9
12	11	10
13	11	10
14	12	11
15	12	11
16	13	12
17	13	13
18	14	13
19	14	13
20	15	13
21	15	13
22	15	13
23	15	13
24	15	13
25	15	13

tiempo y otra en el dominio de la frecuencia. Por una parte, es necesario determinar qué usuario transmitirá en el siguiente paso de simulación. Por otra parte, el planificador en el dominio de la frecuencia selecciona las subportadoras dentro del ancho de banda del sistema cuya respuesta del canal es más adecuada para la transmisión del usuario. Para ello, se ha de estimar la respuesta del canal para cada usuario y para cada subportadora del ancho de banda del sistema. Esta información se deduce de las distintas realizaciones del canal que se generan en la fase de inicialización de la simulación, suponiendo una estimación perfecta de la respuesta del canal. Para seleccionar la frecuencia de la subbanda más apropiada, se usa el valor de CQI. El simulador incluye varias estrategias de planificación de recursos radio [144]:

1 Mejor Canal (*Best Channel*, BC): como su nombre indica, esta estrategia de planificación de recursos radio asigna los PRBs a aquellos usuarios con mejores condiciones del enlace radio. Para llevar a cabo dicha planificación de recursos radio, los terminales envían su CQI a la estación base. Un valor mayor de CQI significa mejor condición de canal. Esta estrategia aumenta la capacidad de la celda, pero no es justa entre usuarios. Los usuarios ubicados en el borde de celda con peores condiciones de canal raramente serán planificados, lo que puede llevar a sufrir grandes retardos de transmisión o, incluso, a la caída de la conexión por falta de recursos.

2 *Round Robin to Best Channel* (RR-BC): esta estrategia de planificación utiliza *Round Robin* en el dominio del tiempo y *Best Channel* en el dominio de la frecuencia. Así, los usuarios se ordenan de forma cíclica sin tener en cuenta sus condiciones de canal, para luego asignarles los PRBs en los que se dan las mejores condiciones de canal. De esta forma se consigue un reparto de recursos mucho más justa, maximizando la eficiencia del sistema.

3 *Large Delay First to Best Channel Scheduler* (LDF-BC): a diferencia de la estrategia anterior, ésta utiliza *Large Delay First* en el dominio del tiempo. Dicha estrategia ordena a los usuarios según el tiempo transcurrido sin transmitir información. Así, a aquellos usuarios que llevan mucho tiempo sin transmitir (por ejemplo, porque se encuentra en el borde de la celda) se les dará un puesto prioritario en la ordenación. En cada TTI se actualiza el tiempo que cada usuario lleva sin transmitir.

4 *Proportional Fair* (PF): esta estrategia de planificación se encuentra a medio camino entre BC y RR. El objetivo de esta estrategia es aumentar la capacidad del sistema, pero garantizando cierto nivel de justicia en el reparto de los recursos radio. Para conseguir dicho objetivo, a la hora de ordenar los usuarios no solamente se tiene en cuenta la tasa binaria potencial sino también la tasa media de transmisión del usuario en TTIs previos. El algoritmo sigue la

expresión

$$\widehat{u}[n] = argmax \left\{ \frac{r_{uk}[n]}{\overline{r_u}} \right\}, \tag{A.4}$$

donde $\widehat{u}$ es el orden del usuario seleccionado u, $r_{uk}[n]$ es la tasa de transmisión potencial del usuario u en el TTI n para el PRB k (obtenida del CQI) y $\overline{r_u}$ es la tasa de transmisión media experimentada por el usuario hasta el momento [101]. Así, se planificarán antes aquellos usuarios con una tasa de transmisión potencial relativamente mejor que su tasa media. De esta forma, se planifican los usuarios cuando su conexión se encuentra en las mejores condiciones.

5 Exponencial clásico/*proportional fair* [105]: esta estrategia de planificación de recursos radio es una modificación de PF que tiene en cuenta tanto prioridad del servicio o del usuario como la calidad instantánea del canal. La introducción de este nuevo factor modifica la planificación con respecto al caso en el que sólo se considera la prioridad de servicios y usuarios, ya que podría suceder que aquellos usuarios o servicios con prioridad más baja a priori reciban recursos antes, por disponer de mejores condiciones de transmisión. De este modo, se persigue maximizar la eficiencia de la asignación de recursos, manteniendo el objetivo de la estrategia PF: la justicia o *fairness*.

En todas ellas, el CQI aporta información de la calidad del canal que experimenta cada usuario. En todos los experimentos realizados a lo largo de esta tesis, se ha utilizado la última estrategia de planificación de recursos radio (Exponencial clásico/*proportional fair* [105]).

A.4. Capa de red

En las funciones principales de nivel de red, se encuentran los procesos de gestión de recurso radio (*Radio Resource Management*, RRM). En esta sección, se describen los procesos de control de admisión y traspaso implementados en el simulador.

A.4.1. Control de admisión

Una vez el usuario decide comenzar una conexión, la primera decisión a tomar es qué celda será su celda servidora. Esta decisión se toma en dos pasos. El primer paso es comprobar si el nivel de señal recibida es suficiente. Para ello, el usuario envía a la red el nivel de señal de referencia recibido (RSRP) de la celda de la red en la que está acampado y de todas sus celdas vecinas. Todas estas

celdas se ordenan en una lista de mayor a menor nivel de señal recibido, etiquetándose como celdas candidatas aquellas que cumplen que

$$RSRP(i) \geq RSRP(i)_{mín} \ , \tag{A.5}$$

donde $RSRP(i)$ es la medida del nivel de señal de referencia recibido de la celda i, y $RSRP_{min}(i)$ es el nivel mínimo de señal necesario para que la celda i sea aceptada como candidata a ser la celda servidora del usuario [101]. Este nivel mínimo de señal se define por celda. Finalmente, se selecciona la mejor celda i de la lista como la opción inicial para la celda servidora.

A.4.2. Algoritmos de traspaso

El algoritmo de traspaso es la principal funcionalidad para gestionar la movilidad de los usuarios. Los algoritmos de traspaso son específicos de cada proveedor de red. Los siguientes párrafos describen el algoritmo de traspaso propuesto para LTE implementado en el simulador.

A.4.2.1. Traspaso por balance de potencia

Un traspaso por balance de potencia se dispara cuando:

$$RSRP(i) - RSRP(j) \geq Margen_{PBGT}(i,j) \text{ durante } TTT^{PBGT} \text{ segundos.} \tag{A.6}$$

La ecuación (A.6) se evalúa cada N^{PBGT} segundos. El traspaso por balance de potencia no se considera como un traspaso urgente, sino como un algoritmo de optimización, puesto que busca que el usuario esté siempre conectado a la mejor celda, independientemente de que en su celda servidora actual pueda tener un buen nivel de señal. Al final del proceso de traspaso por balance de potencia, el usuario debería estar conectado a la mejor celda en términos de señal recibida (siempre que $Margen_{PBGT}(i,j)$ tenga un valor positivo).

Apéndice B

Summary in English

This final appendix summarizes this Ph.D. thesis in English to meet the requirements of international doctorate, as described in RD 99/2011. It is organized as follows. Section B.1 introduces this work and states its general objective. Section B.2 presents two algorithms for QoE balancing and optimizing, respectively. Section B.3 presents a QoE-driven MRO algorithm based on Artificial Intelligence. Finally, Section B.4 states the final conclusions of this work and future lines of work.

B.1. Motivation

Over the last years there has been an exponential growth in the traffic demand associated to mobility services [1]. Besides, the success of smart phones and tablets has changed traffic patterns in cellular networks, turning their management into a very complex task.

All these changes have generalized the use of automatic tools to manage mobile communications networks, known as Self-Organizing Networks (SON) [2]. These automatic techniques cover processes such as network dimensioning, radio planning, deployment, monitoring, troubleshooting and optimization. The techniques applied to these processes are often classified into three different groups:

- Self-configuration (a.k.a. self-planning), which refers to automatic configuration and integration of the network elements added to the global system (e.g., base stations).

- Self-healing, referring to immediate detection of network failures to take preventive actions to avoid performance degradation, and
- Self-optimization (or self-tuning): maintenance of the optimal network parameters configuration.
- This Ph.D. thesis is focused on the latter group of SON techniques, namely self-optimization.

This PhD thesis is focused on the last group of SON techniques, self-optimization.

Traditionally, the management of mobile communications networks has been based on Quality of Service (QoS) indicators such as performance counters and alarms. Nevertheless, the rising user expectations have forced network operators to change the way they manage their networks. Provided that nowadays the offer of different services is similar for all operators, Quality of Experience (QoE) has become the main differentiating factor, and, thus, network operators are changing management processes to be more modern by focusing on the user opinion and QoE.

QoE is defined as the overall acceptability of a service as it is subjectively perceived by the end user. QoE is usually measured using the *Mean Opinion Score* (MOS) scale, ranging from 1 (very bad experience) to 5 (excellent experience) [3].

Due to the different users' expectations for each service, it is common that different services show completely different behavior with regards to QoE, even under identical networks conditions [3]. Such a diversity, together with the difficulty of estimating the QoE on a per-user basis, has made it very difficult for operators to include QoE aspects in network management. However, with the advent of big-data processing techniques, operators can now leverage massive mobile data to add a QoE perspective to network management.

To the best of author's knowledge, QoE-driven cellular network management is a recent field of research. Yet, for this to become a reality, it is important to convince operators that traditional network management techniques (i.e., not considering this QoE perspective) show important performance limitations. Equally important, for a correct QoE-perspective in the design of optimization techniques, operators must make the most of all the information available in their networks to guarantee an efficient network operation. This will be especially true for upcoming 5G systems, where the combination of different radio access technologies and virtualization techniques will make the network extremely complex, diverse and dynamic.

The industry search for efficient network management in the above complex and changing environment has favored intense research on self-optimization processes [145]. Self-optimization (or

self-tuning) aims at maintaining the optimal configuration of network parameters regardless of changes in the environment. The most relevant use cases are capacity and coverage optimization (CCO), inter-cell interference coordination (ICIC), Load Balancing (LB), and Mobility Robustness Optimization (MRO) [4]. This thesis is focused on the latter two use cases.

Load balance aims at minimizing the negative effects of the uneven distribution of cellular traffic both in time and space, causing that some network elements are overloaded, while others are underutilized. Although these problems are common to all network segments, the radio access network has attracted most attention.

There exist several techniques to share load between neighbor cells in a mobile network, among which the modification of the cell service area stands out due to its efficiency. In the literature, different ways of changing the cell service area have been proposed. A first group of techniques tune physical parameters of the base stations, such as the pilot transmit power [5] or the antenna tilt angle [6] [7] [8]. In practice, these techniques are rarely used to balance traffic because they may generate coverage holes.

Alternatively, a second group of techniques modifies parameters of the radio resource management processes, as cell reselection [9] or handover [10]. The latter is usually the preferred option since the effect of a change in cell reselection parameters only impacts the IDLE mode UEs and they do not contribute to the network load as they do not have active connections. For this reason, most of the traffic steering algorithms for cellular networks use handover margins [11] [12]. To find the optimal value for these parameters, the problem of tuning handover margins can be formulated as a classical optimization problem [13]. However, operators usually solve this problem by means of heuristic rules, as the data and measurements needed to build the analytical model are hardly available. Tuning rules can be defined as simple control rules taken from expert knowledge [18] [146] or derived from interactions with the system by reinforcement learning techniques [16].

Depending on the speed of the traffic sharing process, the performance indicator to be balanced may be the average cell load [11] [12] or the blocking ratio [13]. As shown in [14], the latter option behaves better when congestion problems are persistent, due to its higher stability.

In this work, two novel QoE-aware algorithms are proposed for LTE systems. The first algorithm aims to minimize QoE differences across cells and services by adjusting handover parameters on a per-adjacency basis. The second algorithm aims to maximize the overall system QoE.

As aforementioned, traffic steering techniques use the handover mechanism to solve congestion problems by equalizing a previously selected traffic indicator. However, the handover itself might fail

as a consequence of a wrong setting of parameters defining triggering thresholds and margins. If this is the case, a finer tuning of handover parameters is needed, so that the user does not experience any failure as it traverses the network. Traditionally, handover parameter configuration has been done manually, which implies a large workload and, besides, has a negative impact on the robustness of the configuration. Should network conditions change, the old configuration might be wrong, causing that users may suffer from handover failures (i.e., handover is not completed) or interrupted connections (a.k.a. radio link failures, RLF). To address this problem, the Mobility Robustness Optimization (MRO) SON use case is devoted to the optimization of handover performance. In this context, handover optimization aims to detect RLFs caused by handovers performed too early or too late [22] with the goal of dynamically improving the handover performance in the network. This action seeks to improve end user experience, while reducing unnecessary signaling load in the network. This is achieved by tuning handover parameters to adapt cell borders depending on certain performance indicators. Automating these changes minimizes human intervention in network management and optimization tasks.

In the literature, many references address that problem of a correct handover parameters configuration [24] [24]. A first set of approaches [118] [119] [120] [121] [122] use an analytical model to find the optimal HandOver (HO) parameter settings by formulating the tuning problem as a multi-objective optimization problem, whose objective function includes the number of HOs per call, the outage probability, the cell-edge spectral efficiency or RLF rates. A second set of approaches use self-tuning methods during network operation to adjust the parameters of an existing HO scheme based on threshold crossing [23] [123]. Changes are made by iterative control algorithms driven by heuristic rules taken from expert knowledge, which can be adapted to outdoor [23] or indoor [123] environments. Such an approach can be improved by proactively changing HO parameters based on RLF prediction [124]. However, with heuristic rules, it is not guaranteed that optimal HO settings are reached in steady state. A third set of approaches redesign the HO scheme that processes instantaneous signal measurements to decide when to trigger a HO for each individual user [81] [28]. These adaptive HO schemes constantly improve by analyzing their past behavior, avoiding the need for an expert, and becoming a powerful tool for network optimization. Their main drawback is that they require updating vendor equipment, whereas the analytical and self-tuning approaches can still be used with existing infrastructure. However, none of the previous references take explicitly into account QoE as an indicator to drive the optimization process.

With recent advances in information technologies, big data analytics can be used for automated QoE management [147] [148]. In this context, Machine Learning (ML) algorithms can help to convert network data into actionable insights. Reinforcement Learning (RL) and Artificial Neural Networks (ANN) are two of the most popular ML tools used for self-optimization in mobile

networks [77]. With these techniques, legacy reactive self-tuning algorithms, based on threshold comparison and simple heuristic rules, can be substituted by intelligent schemes that autonomously find the best tuning policies and proactively change network settings to quickly adapt to changing environments [78]. To this end, ML techniques have already been applied to MRO [72] [28] [29] [81] [73].

In this Ph.D. thesis, several control and optimization algorithms are proposed to improve user QoE while increasing network operational efficiency. The proposed algorithms are based on information available in the Operational Support System (OSS). To this end, two sources of information are combined, namely perfomance counters aggregated at a cell/period basis and connection traces that give details of individual user performance. Thus, these algorithms have been conceived to be integrated into network optimization tools included in the OSS of a cellular network.

B.1.1. Research objectives

The general purpose of this thesis is the design and validation of several handover parameter optimization algorithms for LTE networks, based on artificial intelligence:

1 QoE-driven traffic steering algorithm based on Fuzzy Logic.

2 Self-tuning algorithm for optimal QoE-driven traffic steering.

3 Service-centric Q-learning algorithm for mobility robustness optimization based on reinforcement learning.

The work methodology is the same for all research objectives. Firstly, the problem is formulated, detailing what is to be solved and the review of the technique. Secondly, the simulation tool is updated to develop and validate the optimization algorithms. Finally, the optimization algorithms proposed are designed and validated using a dynamic system-level LTE simulator [101] implementing a realistic macrocell scenario, and a real LTE Pilot network.

B.1.2. Appendix structure

After this first section, Section B.2 presents the self-tuning algorithm for balancing QoE across cells or maximizing the overall system QoE. Section B.3 describes a QoE-aware MRO algorithm to

maximize cell edge users QoE, while improving handover performance. Finally, Section B.4 summarizes the main conclusions of this Ph.D. thesis and future work, including the list of publications resulted from the research.

B.2. QoE-driven Traffic Steering in LTE

This Section is organized as follows. Section B.2.1 reviews the technique used in this Section. Section B.2.2 discusses the limitations of classical load balancing schemes in terms of QoE. Section B.2.3 describes the system model used in this work. Section B.2.4 describes the proposed QoE balancing algorithm, whose assessment process is presented in Section B.2.4. Section B.2.4 describes the QoE optimization algorithm. Section B.2.4 presents the assessment process. Finally, Section B.2.6 summarizes the main conclusions.

B.2.1. Review of the technique

Schedulers are, perhaps, one of the first subsystems in mobile networks to include a QoE perspective in their management. To this end, QoS-aware schedulers implement certain resource sharing strategies. These scheduling processes dynamically assign radio resources to users depending on QoS criteria [103] [104]. Likewise, more sophisticated schedulers consider QoE aiming to optimize network average QoE while ensuring a minimum QoE for all users. These advanced schedulers are usually designed for particular services (e.g, web [106], progressive video streaming [107], or adaptive video streaming [108], [109], [110]). Implementing these schedulers implies updating current network equipment, which requires an important investment, something that network operators are usually reluctant to. Alternatively, some works propose a self-tuning algorithm for standard schedulers with the aim of balancing [37] or optimizing [38] the QoE of different services within a cell. Nevertheless, the main objective of most schedulers is to guarantee a minimum QoS or QoE value to those users with lower levels, or balancing average QoS or QoE within a cell, rather than balancing average service QoE across cells in the network.

Thus, user or service QoE balance is not guaranteed in the spatial domain. A QoE imbalanced network implies an unfair radio resource allocation: while fully satisfied users in underutilized cells waste available radio resources without increasing their QoE, there are absolutely unsatisfied users in congested cells who lack these resources. Ultimately, this unfair user satisfaction distribution may turn in higher churn rates and corresponding revenue losses.

Apart from QoS/QoE aware schedulers, classical traffic steering techniques aim to balance average cell load throughout the network. Even if they may potentially reduce QoE differences among cells, to the authors' best knowledge, in the literature there are no traffic steering schemes proposed that explicitly consider QoE.

In this Section, two novel QoE-aware traffic steering algorithms are proposed for LTE networks. In contrast to traditional techniques, the aim of the first algorithm is to minimize cell and service QoE differences by adjusting handover (HO) parameters on a per adjacency basis. This adjustment is carried out using Fuzzy Logic Controllers (FLC) driven by QoE estimates from key performance indicators. The second approach uses a gradient ascent algorithm to ensure that handover margin changes, performed on an adjacency basis, always improve the system global QoE. Both algorithms are validated in a dynamic system-level LTE simulator implementing a realistic macrocellular scenario. The main contributions of this Section are: a) to uncover the limitations of classical traffic steering techniques from a QoE perspective, b) a novel self-tuning algorithm based on fuzzy logic to balance QoE by modifying handover margins, c) another gradient ascent algorithm to optimize QoE for users demanding an FTP (File Transfer Protocol) service by modifying handover margins, and d) the validation of both algorithms employing simulations in a realistic macrocellular LTE scenario.

B.2.2. Problem formulation

In mobile networks, the handover process ensures seamless connections to users moving between two adjacent cells. Specifically, a handover takes places when the following condition is fulfilled

$$P_{rx}(j) - P_{rx}(i) \geq HOM(i,j)\,, \tag{B.1}$$

where $P_{rx}(j)$ is the pilot signal level received from the neighbor cell j, $P_{rx}(i)$ is the pilot signal level received from the serving cell i, and HOM is the HandOver Margin (HO Margin), defined at adjacency level (i.e., one value per cell pair and direction of the adjacency). In most cases, handover margins are established complementarily in both directions of the adjacency to avoid Ping-Pong effect, so that

$$HOM(i,j) + HOM(j,i) = H\,, \tag{B.2}$$

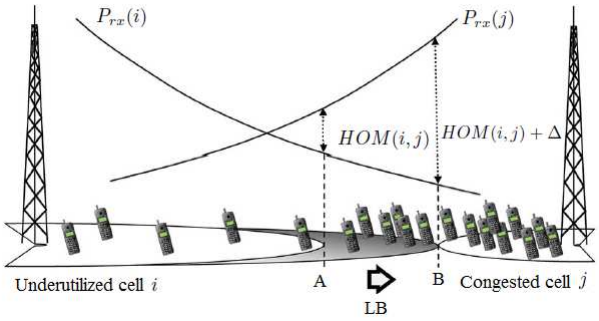


Figure B.1: Traffic sharing by changing handover margins [21].

where H represents the hysteresis value.

Figure B.1 illustrate how handover margins modifications can be used as a traffic steering technique [21]. An increment of Δ dB in $HOM(i,j)$ enlarges cell i service area and shrinks that of cell j. Thus, total traffic (and load) carried by cell i increases, while that of cell j decreases. On the contrary, when $HOM(i,j)$ decreases, cell i service area decreases, whereas that of cell j increases.

Traditional LB schemes modify handover margins for balancing cell load among cells in the network in the hope of improving global call blocking ratio (or any other QoS indicator) [12] [16]. But a cell load balanced network does not necessarily imply a QoE balanced network [115]. User satisfaction is highly dependent on the service demanded, causing that QoE of users of different services significantly differs even if they receive the same amount of radio resources. More importantly, balancing load among neighbor cells does not necessarily balancing QoE, since some services are more sensitive to load increment than others. As a consequence, balancing load between two adjacent cells usually does not reduce QoE differences if the service mix is not exactly the same in both cells, as it is generally the case.

The above considerations suggest that a traditional traffic steering scheme based on QoS results in an even network load, but may lead to a situation where the QoE distribution is unbalanced [115]. This is the main hypothesis to be tested in the first QoE balancing algorithm, whose aim is to improve the worst user QoE at the expense of degrading fully satisfied users QoE. This action does not necessarily improve the global system QoE, but it does balance QoE among cells.

As a result of the first algorithm, a QoE balanced network is obtained, which does not necessarily lead to an improvement of the global system QoE. Thus, a second traffic steering algorithm aims to optimize the overall system QoE, even when the optimization may lead to a less fair average cell

Table B.1: Main simulation parameters.

Time resolution	10 TTI (10 ms)
Propagation model	Pathloss COST 231 Hata, slow fading (log-normal σ = 8 dB, d_{corr} = 20 m), fast fading (ETU model)
Base station model	Tri-sectorized antennas, MIMO 2x2, BW = 5 MHz (25 PRB), $f_{carrier}$ = 2 GHz, EIRP$_{max}$= 68 dBm.
Scheduler	Classical exponential/proportional fair [105]
Link adaptation	CQI-based

Table B.2: Traffic model parameters.

Service	**Main features**
VoIP	Coding rate 16 kbps Session time: exponential distribution (avg. 60 s). Call dropped after 1 s without resources.
VIDEO	H.264/MPEG-4 AVC VBR (Variable bit rate) 720p resolution, 25 frames per second. Video duration: uniform distribution between 0 and 540 s. Frame size according to real traces (avg. 9.2 MB). Connection dropped when stalling lasts for twice the video duration.
FTP	File size: log-normal distribution (avg. 20 MB) [41].
WEB	Web page size: log-normal distribution (avg. 20 MB). No. pages per session: log-normal (avg. 4). Waiting time: exponential distribution (avg. 107 s) [41].

QoE distribution. This second algorithm is designed following a gradient ascent algorithm, making sure that any small handover margin modification generates a system QoE improvement, something much appreciated by network operators. For simplicity, the analysis is restricted to FTP services.

B.2.3. System model

This work has been carried out by using a dynamic system-level LTE simulator [101], where a realistic 108-macrocell scenario has been implemented. This scenario is illustrated in [115]. The main simulation parameters are shown in Table B.1. The key traffic model parameters for every service in the simulator (i.e., VoIP, VIDEO, FTP, and WEB) are described in [115], and briefly summarized in Table B.2.

B.2.3.1. QoE models

User QoE is calculated once the service session is finished [115]. QoE calculation is implemented in two steps: first, key service parameters are extracted from simulation data, and, secondly, user QoE is estimated by a service-based utility function, using a MOS scale. These utility functions quantitatively model the relationship between QoS indicators taken directly from the network and QoE measurements, showing the impact of QoS on the final user subjective perception.

For VoIP service, user QoE can be estimated as [96]:

$$QoE^{(VoIP)} = 1 + 0.035R + R(R-60)(100-R)7 \cdot 10^{-6} \,, \tag{B.3}$$

where R is a parameter representing the connection quality taking values from 0 (minimum) to 93 (maximum). R only depends on the delay experienced by VoIP packets (mouth-to-ear delay). Specifically, R is defined here as the 90^{th} percentile of the packet delay for the entire connection. Note that the maximum value of the QoE for this service is $QoE^{(VoIP)} = 4.405$. Dropped VoIP calls reach the minimum QoE value $QoE^{(VoIP)} = 1$.

Video utility function is defined as

$$QoE^{(VIDEO)} = 4.23 - 0.0672L_{ti} - 0.742L_{fr} - 0.106L_{tr} \,, \tag{B.4}$$

where L_{ti} denotes the initial buffering time (in seconds), L_{fr} is the average stalling frequency in Hertz (i.e., number of times per second that the video player is paused due to an empty client buffer), and L_{tr} is the average stalling duration (in seconds). The maximum QoE value for a video connection is upper limited to 4.23. As with VoIP, $QoE^{(VIDEO)} = 1$ if the connection is dropped.

The utility function for FTP service is

$$QoE^{(FTP)} = \max(1, \min(5, 6.5 \cdot T - 0.54)) \,, \tag{B.5}$$

where T denotes the average user throughput in Mbps.

Finally, the utility function for WEB service is

$$QoE^{(WEB)} = 5 - \frac{578}{1 + (\frac{T+541.1}{45.98})^2} , \tag{B.6}$$

where T is the average user throughput in kbps. Note that, $\max(QoE^{(WEB)}) = 5$.

B.2.4. QoE balancing algorithm

This section presents a self-tuning algorithm with the objective of balancing QoE among cells in an LTE network. The proposed algorithm, hereafter referred to as EB (for Experience Balancing), is based on the Mobility Load Balancing (MLB) algorithm described in [18], hereafter referred to as LB (for Load Balancing). As LB, EB is implemented using Fuzzy Logic Controllers, FLCs, which are in charge of deciding whether to increment or not the handover margin, HOM, on an adjacency basis. In contrast to LB whose objective is balancing cell average load throughout the network, EB aims to balance cell average QoE. For this purpose, users are handed over from a cell with a low average QoE to a neighbor cell with a higher average QoE.

EB defines two variants. The first one, EB-C (for cell), aims to balance average cell QoE, defined as

$$\overline{QoE}(i) = \frac{\sum_s \overline{QoE}(i,s)}{N_s(i)} , \tag{B.7}$$

where $\overline{QoE}(i,s)$ is the average QoE for users demanding service s in cell i, defined as

$$\overline{QoE}(i,s) = \frac{\sum_{\forall u \in i\,, S(u)=s} QoE^{(s)}(u)}{N_u(i,s)} , \tag{B.8}$$

where $QoE^{(s)}(u)$ is the quality of experience for any user u of a service s, calculated as indicated in (B.3)-(B.6), N_s is the number of services demanded in cell i and $N_u(i,s)$ is the number of users in cell i demanding service s. In (B.7), it is implicitly assumed that all services are equally important to the network operator. This assumption is in line with the objective of ensuring all users the same QoE regardless of the service demanded.

Finally, the global QoE is defined by averaging across cells, as

$$\overline{QoE} = \frac{\sum_{i=1}^{N_c} \overline{QoE}(i)}{N_C} , \tag{B.9}$$

where N_C is the number of cells in the network.

As previously mentioned, the goal of EB-C is to balance cell average QoE (B.7). Thus, a difference indicator is defined for EB-C as follows

$$QoE_{diff}(i,j) = \overline{QoE}(j) - \overline{QoE}(i) . \tag{B.10}$$

This indicator is used as an input to EB-C FLC. Figure B.2 shows the EB-C FLC structure. The output variable is the HOM increment/decrement between adjacent cells i and j, $\Delta HOM(i,j)$. As illustrated in the figure, FLC comprises three stages: fuzzification, inference, and defuzzification.

The first stage is the fuzzification stage, where membership functions, μ_x, translate the input value into a membership degree raging from 0 to 1 according to some linguistic labels (e.g., Very Negative, Negative, Zero, Positive, Very Positive...) [66]. The specific membership functions are shown in Figure B.3a. In the second stage, known as the inference stage, a set of «IF-THEN» rules define an input-to-output mapping using linguistic terms. User rules are described in Table B.3c. As inferred from the rules, the more negative/positive the value of $QoE_{diff}(i,j)$ (i.e., the higher QoE imbalance) the higher increment/decrement for $HOM(i,j)$. Finally, in the defuzzification stage, the final value of $HOM(i,j)$ is calculated by means of the output membership functions, shown in Figure B.3b. For simplicity, a Takagi-Sugeno approach [66] is used, where the output is obtained by aggregating the result of every rule using the *centre-of-gravity* method [111]. This method calculates $HOM(i,j)$ as a weighted average, where the weights are deduced from the degree of fulfillment of the activated rules.

FLCs are executed periodically with a fixed time period, called Reporting Output Period (ROP). The scheme followed per iteration is always the same: collecting key performance indicators, using their values in the corresponding FLC, and calculating the change of handover margin change for the next iteration. The $HOM(i,j)$ value for the next algorithm iteration, hereafter referred to as optimization loop, is calculated as follows

$$HOM^{(n+1)}(i,j) = \min(\max(\text{round}(HOM^{(n)}(i,j) + \Delta HOM^{(n)}(i,j)), -7), 13) , \tag{B.11}$$

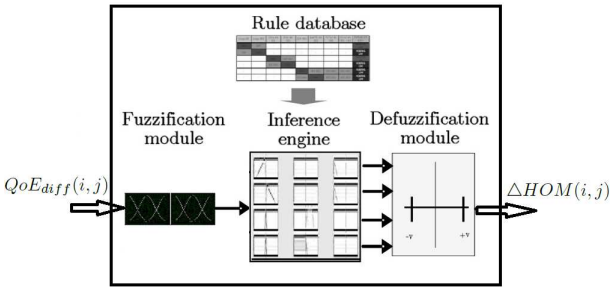
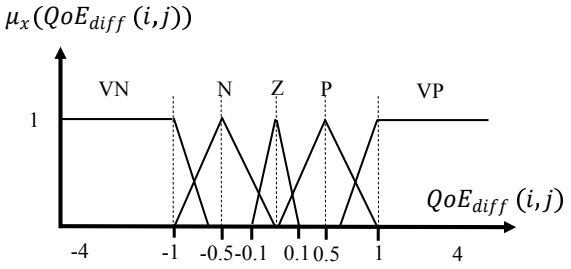
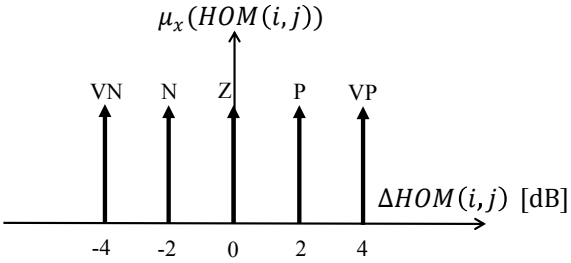


Figure B.2: FLC structure for EB-C[18].



(a) Input membership functions.



(b) Output membership functions.

$QoE_{diff}(i,j)$	$\Delta HOM(i,j)$
VP	VN
P	N
Z	Z
N	P
VN	VP

(c) Rules.

Figure B.3: Fuzzy logic controller for EB-C algorithm.

where superscripts n and $n+1$ denote the iteration number, with all parameters expressed in dB. Please note that HOM changes are limited to the range -7 to 13 dB. The lower limit is the minimum signal-to-interference-plus-noise ratio (SINR) configured by most vendors in their schedulers to allocate any radio resources for a connection. The upper limit is calculated with (3.2) to guarantee a hysteresis level of $H = 6$ dB. From (3.2), it is also deduced that any $HOM(i,j)$ change automatically leads to a change in the opposite direction of the adjacency, $HOM(j,i)$. As a result, EB-C only needs one FLC per adjacency. Consequently, the number of FLC needed equals the number of adjacencies in the system.

Due to their similarities, EB-C and LB should work the same way (i.e., propose the same HOM changes) if the service mix was exactly the same in all cells. Nevertheless, this is usually not the case.

It is worth noting that, even if EB-C ensures average cell QoE equalization among cells in the network (i.e., $\overline{QoE}(i) \simeq \overline{QoE}(j) \forall i,j$), some services may experience higher QoE than others within a cell (i.e., $\overline{QoE}(i,s_1) \neq \overline{QoE}(i,s_2)$), or different QoE to that experienced by the same service in other cells (i.e., $\overline{QoE}(i,s_1) \neq \overline{QoE}(j,s_1)$). To solve these problems, an alternative method, EB-CS, is proposed, whose objective is to balance the average service QoE in a cell with the average cell QoE in neighbor cells. EB-CS exploits the flexibility of establishing different HOM values per service within the same adjacency. Thus, HOM tuning is carried out on per-adjacency and per-service basis, driven by a new difference indicator

$$QoE_{diff}(i,j,s) = \overline{QoE}(j) - \overline{QoE}(i,s)\,. \tag{B.12}$$

As for EB-C, $QoE_{diff}(i,j,s)$ for EB-CS is defined per adjacency, but differently, it is also segregated per service. Thus, an FLC is needed per service and adjacency, one for each $HOM(i,j,s)$. EB-CS can be understood as four QoE balancing mechanisms per adjacency (i.e., one per service), which may work differently. For example, WEB users may be handed over from cell i to cell j, but VIDEO users may be forced to move from cell j to cell i. Input and output membership functions and inference rules are identical to that of EB-C, shown in Figure B.3. The only difference is that the input variable is $QoE_{diff}(i,j,s)$, instead of $QoE_{diff}(i,j)$.

Another important difference between EB-CS and EB-C is the loss of symmetry. The difference indicator in (B.12) is calculating by subtracting $\overline{QoE}(i,s)$ to $\overline{QoE}(j)$, so that $QoE_{diff}(i,j,s) \neq QoE_{diff}(j,i,s)$. Unlike EB-C, EB-CS needs to execute two FLCs per adjacency, one per each direction. These independent executions may lead to changes of different magnitude within the same adjacency, $\Delta HOM(i,j,s)$ and $\Delta HOM(j,i,s)$, not fulfilling (3.2). To maintain a constant

hysteresis level, EB-CS calculates a ΔHOM average per adjacency as

$$\overline{\Delta HOM}^{(n)}(i,j,s) = -\overline{\Delta HOM}^{(n)}(j,i,s) = \frac{\Delta HOM^{(n)}(i,j,s) - \Delta HOM^{(n)}(j,i,s)}{2} . \quad \text{(B.13)}$$

Assessment methodology

EB-C and EB-CS are compared with other self-tuning methods in a realistic scenario to assess their performance gain. In the experiments, only DL is simulated to reduce computational load. Two experiments are carried out to assess the performance of the proposed algorithm. The first experiment is performed in the LTE system-level simulator, the second experiment is performed in a operating LTE pilot network located in the University of Málaga. The experimental set-up is presented first, and results are shown later.

Experiment 1

Experiment set-up During the analysis, five self-tuning approaches are compared. The first two are EB-C and EB-CS whose aim is to equalize user experience across cells or cells and services, respectively. The other three schemes are proposed in the literature. The third scheme is the legacy MLB algorithm [18], LB, aiming to equalize the average PRB utilization between neighbor cells, $U(i)$. The aim of the fourth scheme, referred to as throughput-based balancing, TB [17], is to balance average cell throughput, $T(i)$, across cells in the network. Finally, the fifth scheme, referred to as QoE-based reprioritization, QR [37] aims to balance the QoE of users within a cell by reprioritizing services in a classical scheduler.

The scenario used in this experiment is illustrated in [115]. Simulation parameters for this experiment are those presented in Table B.1, with the only difference that system bandwidth is increased to 10 MHz (50 PRB).

The five self-tuning algorithms (LB, TB, QR, EB-C, and EB-CS) are tested along 15 optimization loops. The system reaches stability after 15 iterations in the five algorithms. The duration of every optimization loop (i.e., the ROP) is 1 hour. At the end of each loop, the indicators used as drivers ($U(i)$, $T(i)$, $\overline{QoE}(i,s)$, $\overline{QoE}(i)$ and $\overline{QoE}(i,s)$) are collected and algorithms are triggered. After each optimization loop, the system updates HOM or SPI values and a new optimization loop begins. Network performance with the default HOM/SPI settings is considered as a baseline.

EB-C and EB-CS reduce QoE differences between users of different cells and services by improving the worst users/services, at the expense of deteriorating that of the best users/services. For consistency, the main figure of merit is the 5^{th} percentile of the QoE distribution across cells and services in the network, $\overline{QoE}^{(5\%-th)}(i,s)$.

A secondary figure of merit is the overall system QoE, computed as the average of all services and cells in the scenario,

$$\overline{QoE} = \frac{1}{N_C}\sum_i \overline{QoE}(i) = \frac{1}{N_C}\sum_i \frac{\sum_s QoE(i,s)}{N_s(i)} \,. \tag{B.14}$$

Five additional key performance indicators are defined to check performance differences across cells and services in the network. An overall cell load imbalance indicator is defined as

$$\overline{U_{imb}} = \frac{1}{N_C}\sum_i |U_{imb}(i)| = \frac{1}{N_C}\sum_i \left| U(i) - \frac{\sum_{j\in A(i)} U(j)}{N_{adj}(i)} \right| \,, \tag{B.15}$$

where $U_{imb}(i)$ is the average PRB utilization imbalance of cell i, computed by comparing its average PRB utilization against that of its neighbors, $A(i)$ is the set of neighbor cells of cell i, $N_{adj}(i)$ is the number of neighbor cells of cell i and N_c is the number of cells in the scenario. An overall intra-cell QoE imbalance indicator is defined as

$$\overline{QoE_{imb,f}} = \frac{1}{N_C}\sum_i |QoE_{imb,f}(i)| = \frac{1}{N_C}\frac{1}{N_s(i)}\sum_i\sum_k \left|\triangle\overline{QoE}(i,s_k)\right| \,, \tag{B.16}$$

where

$$\triangle\overline{QoE}(i,s_k) = \overline{QoE}(i,s_k) - \frac{\sum_{s\neq s_k} \overline{QoE}(i,s)}{N_s(i)-1} \,, \tag{B.17}$$

and $QoE_{imb,f}(i)$ is the QoE imbalance among services in cell i, calculated as the mean value of the difference between the QoE of a service and the mean QoE for the rest of the services, as in (B.17).

Similarly, an overall throughput imbalance indicator is defined as

$$\overline{T_{imb}} = \frac{1}{N_C}\sum_i |T_{imb}(i)| = \frac{1}{N_C}\sum_i \left| \overline{T}(i) - \frac{\sum\limits_{j\in A(i)} \overline{T}(j)}{N_{adj}(i)} \right| , \tag{B.18}$$

where $T_{imb}(i)$ is the throughput imbalance indicator of cell i, computed by comparing its average throughput against that of its neighbors, $\overline{T}(j)$ is defined as

$$\overline{T}(i) = \frac{\sum\limits_s \overline{T}(i,s)}{N_s(i)} , \tag{B.19}$$

where

$$\overline{T}(i,s) = \frac{\sum\limits_{u\in(i,s)} T(u)}{N_u(i,s)} . \tag{B.20}$$

and $T(u)$ is the connection throughput of user u. Likewise, an overall QoE imbalance indicator across cells in the scenario is defined as

$$\overline{QoE_{imb,c}} = \frac{1}{N_C}\sum_i |QoE_{imb,c}(i)| = \frac{1}{N_C}\sum_i \left| \overline{QoE}(i) - \frac{\sum\limits_{j\in A(i)} \overline{QoE}(j)}{N_{adj}(i)} \right| , \tag{B.21}$$

where $QoE_{imb,c}(i)$ is the average cell imbalance indicator of cell i, computed by comparing its average QoE against that of its neighbors. Finally, an overall QoE imbalance indicator across services in the scenario is defined as

$$\overline{QoE_{imb,s}} = \frac{1}{N_C}\sum_i |QoE_{imb,s}(i)| = \frac{1}{N_C}\frac{1}{N_s(i)}\sum_i\sum_s \left| \overline{QoE}(i,s) - \frac{\sum\limits_{j\in A(i)} \overline{QoE}(j)}{N_{adj}(i)} \right| , \tag{B.22}$$

where $QoE_{imb,s}(i)$ is the average cell and service imbalance indicator of cell i, computed by comparing its average QoE per service against the average QoE of its neighbors.

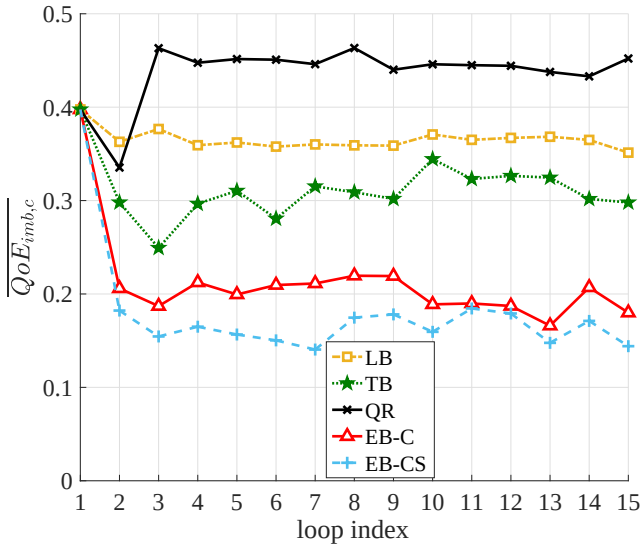


Figure B.4: Evolution of QoE imbalance.

Both spatial user distribution and service mix are taken from real statistics. Only the call arrival rate is artificially modified (i.e., increased) to generate a highly loaded scenario. The average cell load in the scenario is $U(i) = 75$ % and the average cell QoE is $QoE(i) = 3.02$. The default HOM value is 3 dB for all adjacencies and the initial SPI value in all cells is 7 for every service. It should be pointed out that, in the considered live scenario, VoIP traffic is extremely low and scattered in a few cells in the network. As this might cause unreliable QoE statistics, EB-CS is not allowed to change HOM settings for this service.

Results Figure B.4 shows the impact of LB, TB, QR and EB algorithms on QoE imbalance among cells along the 15 optimization loops. As illustrated, LB does not change $\overline{QoE_{imb,c}}$ significantly, QR increases $\overline{QoE_{imb,c}}$ and TB achieves a slight reduction of $\overline{QoE_{imb,c}}$. In contrast, both EB-C and EB-CS more than half the initial imbalance.

To spot the difference between EB-C and EB-CS, Figure B.5 shows the evolution of the imbalance between cells and services, $\overline{QoE_{imb,s}}$, across iterations in both schemes. As expected, EB-CS better equalizes QoE among cells and services due to its service-based design.

For comparison purposes, Table B.3 summarizes the main performance indicators at the beginning (column *Initial*) and the end of the tuning process (15^{th} optimization loop) for the different schemes. As expected, LB achieves the best load balance, $\overline{U_{imb}}$, TB achieves the best throug-

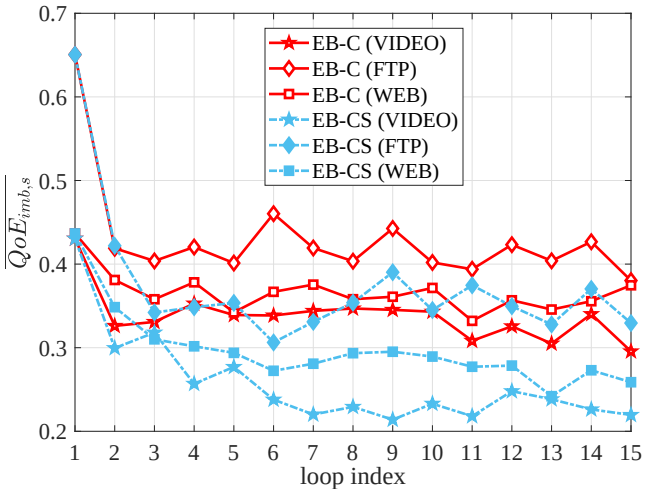


Figure B.5: Evolution of QoE imbalance per service.

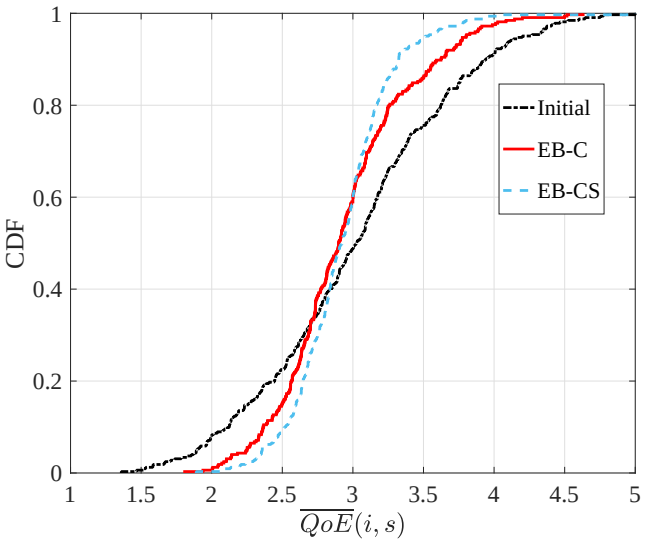


Figure B.6: QoE distribution for services across cells function.

Table B.3: Main performance indicators.

Indicator	Initial	LB	TB	QR	EB-C	EB-CS
$\overline{U_{imb}}[\%]$	15.2	6.3	15.2	14.9	17	16.6
$\overline{T_{imb}}$ $[Mbps]$	1.01	4.57	0.28	0.52	4.09	0.79
$\overline{QoE_{imb,f}}$	0.51	0.50	0.49	0.32	0.46	0.33
$\overline{QoE_{imb,c}}$	0.4	0.45	0.30	0.35	0.18	0.14
$\overline{QoE_{imb,s}}$	0.51	0.56	0.45	0.42	0.35	0.26
$\overline{QoE}$	3.02	2.94	2.96	2.99	2.94	2.93
$\overline{QoE}(i,s)$ 5^{th} tile	1.89	1.90	2.01	2.07	2.10	2.36

hput balance among services within a cell, $\overline{T_{imb}}$ (0.28 Mbps) and QR achieves the smallest QoE imbalance among services within a cell, $\overline{QoE_{imb,f}}$ (0.32). EB-C gets a better QoE balance than LB, TB or QR, $\overline{QoE_{imb,c}}$ (0.18). However, EB-CS achieves the smallest QoE imbalance across cells ($\overline{QoE_{imb,c}}$0.14) and services ($\overline{QoE_{imb,s}}$ = 0.26). This is the result of adjusting HOM on a per-adjacency and per-service basis.

Finally, Figure B.6 shows the cumulative distribution function of $\overline{QoE}(i,s)$ achieved by EB and EB-CS. It is observed that both balancing approaches deteriorate the QoE of the best cells/services to improve that of the worst cells/services. Focusing on the worst cells and services (lower left), it is observed that EB-CS achieves the best improvement for those cells and services experiencing the lowest QoE values. This is also shown in Table B.3, where EB-CS has the highest value for the QoE indicator representing the worst users (i.e., $\overline{QoE}^{(5\%-tile)}(i,s) = 2.36$).

Experiment 2

Experiment set-up The goal of this experiment is to assess EB-CS performance in a real LTE network located in the University of Málaga. The pilot network used in this experiment consists of 12 picocells (Huawei model BTS3911B). Network management is performed remotely using iManager U2000 application. Particularly, only three cells are used in this experiment. Cell 1 is located in laboratory 1.1.2 and cell 2 and cell 3 are located in laboratory 1.1.1. Picocell configuration is shown in Table B.4. Pilot signal strength level has been reduced to minimize the overlapping area between neighbor cells, thus limiting the zone in which HO takes place when changing HO margins.

This experiment pursues to have a heavily loaded cell next to two underutilized cells. For this

Table B.4: Picocells configuration.

Bandwidth	5 MHz (25 PRB)
Reference signal power	-20 dBm
eNB transmission power	23 dBm
Carrier frequency	2.516 GHz
Scheduling algorithm	Proportional Fair (PF)
Default HOM value	3 dB

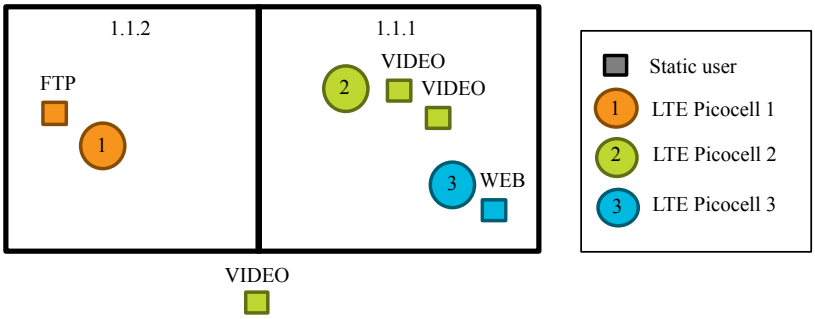


Figure B.7: Initial network layout.

purpose, three VIDEO users are spatially-wise irregularly distributed initially connected to cell 2 as depicted in B.7. One of these users is located in the corridor, right at the border between cell 2 and cell 1, another one is placed at the border between cell 2 and cell 3, and the third one at the cell center. An FTP user is connected to cell 1, and cell 3 has a WEB user within its service area. All users are static. To monitor network performance G-MON [112] application is used. An external in-house developed software is used to collect those Service KPIs (S-KPIs) needed to estimate QoE for each service (throughput, stallings). QoE models used are those described in B.4-B.6. The main traffic parameters are shown in Table B.5. With this initial network setup, cell 2 QoE is much lower than that of cells 1 and 3. By changing HO margins according to EB-C, QoE imbalance is reduced up to a QoE balance point. The experiment carried out lasts for a little more than ten minutes. During this period the video is reproduced once, the file is also downloaded once, and the web page is continuously downloaded 800 times.

Results Figure B.7 shows the network situation when handover margins take their default value (3 dB). From service performance indicators, QoE values are calculated per user. With this initial network setup, $\overline{QoE}(C1) = 4.25$, $\overline{QoE}(C2) = 1$, and $\overline{QoE}(C3) = 5$, thus $\overline{QoE_{imb,c}} = 3.625$.

Figure B.8 shows the network setup after two optimization loops of EB-C. At the end of the optimization process, handover margins from cell 2 towards cells 1 and 3 are reduced, decreasing

Table B.5: Traffic model parameters.

VIDEO	Sequence duration: 600 s. Resolution: 3840x2160 pixels. Average throughput: 14931 kbps. Frame rate: 25 frames/s.
FTP	Fixed file size: 900 MB.
WEB	800 continuous downloads of www.uma.es.

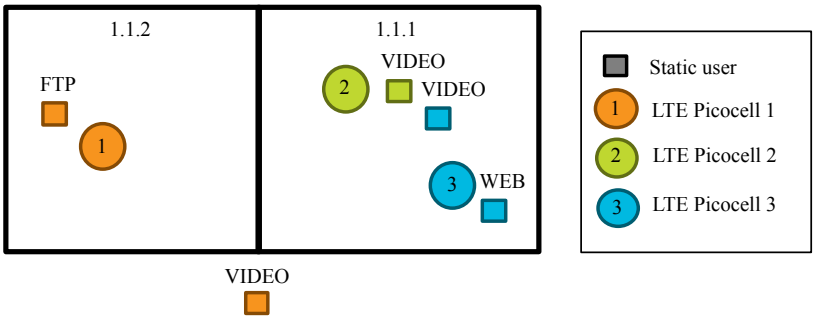


Figura B.8: Network layout after EB-C.

cell 2 service area. Particularly, $HOM(C2, C1) = -5$ dB and $HOM(C2, C3) = -5$ dB. In the opposite way, handover margins are adjusted according to (B.2), $HOM(C1, C2) = 11$ dB and $HOM(C3, C1) = 11$ dB, therefore cell 1 and cell 3 increase their service areas. Thus, one VIDEO user is handed over to cell 1 and another to cell 3. With this final HO parameters setting, $\overline{QoE}(C1) = 2.62$, $\overline{QoE}(C2) = 2.3$, and $\overline{QoE}(C3) = 3$, hence $\overline{QoE_{imb,c}} = 0.36$, proving the ability of EB-C to reduce QoE imbalance among adjacent cells.

B.2.5. QoE optimization algorithm

The above-described EB-C and EB-CS schemes are designed to achieve QoE equilibrium among cells in the network, which implies degrading QoE in cells and services experiencing better QoE to be able to improve those with lower QoE values. As a result, the overall system QoE is slightly degraded, which can be inferred from the shift of the median value to the left in Figure B.6. Alternatively, changes in HO margins can be driven by optimality criteria provided that a network performance model is approximated. Should the approximate model be able to find reasonable estimates of the gradient of the objective function, a gradient ascent algorithm can be used to progressively improve the overall figure of merit. Thus, a local maximum of the problem can be achieved. Equally important, the gradient rule ensures that no parameter change degrades system

performance if the magnitude of changes is small.

The proposed QoE-driven optimization algorithm, hereafter referred to as OE (for Optimizing Experience) [149], modifies the HO margin between neighbor cells i and j, $HOM(i,j)$, with the aim of maximizing overall system QoE, $\overline{QoE}_u$, defined as

$$\overline{QoE}_u = \frac{\sum_u QoE(u)}{N_u} , \tag{B.23}$$

where N_u is the number of users in the system and $QoE(u)$ is QoE for any user u.

For this purpose, OE follows a gradient ascent approach, where an iterative algorithm changes $HOM(i,j)$ based on estimates of the gradient of the objective function, $\overline{QoE}_u$, computed on an adjacency basis with an analytical model. In each iteration (optimization loop), HO margins in all adjacencies are updated as

$$HOM^{(n+1)}(i,j) = HOM^{(n)}(i,j) + \Delta HOM^{(n)}(i,j) = HOM^{(n)}(i,j) + f(\frac{\delta \overline{QoE}_u}{\delta HOM(i,j)}^{(n)}) , \tag{B.24}$$

where superscripts (n) and $(n+1)$ denote the optimization loop index, and $\frac{\delta \overline{QoE}_u}{\delta HOM(i,j)}$ is the gradient of the objective function in the direction of the decision variable $HOM(i,j)$, quantifying the impact of increasing HOM in an adjacency on the overall system QoE. $\Delta HOM^{(n)}(i,j)$ is the value to be estimated with the analytical model described next.

Analytical system model for optimization

The aim of OE is to maximize the overall system QoE. For simplicity, it is assumed here that all users demand FTP service, which can be modeled as a full buffer traffic source until session ends. Yet, it is considered that user experience depends on user context (indoors or outdoors), which can be inferred by analyzing connection traces in practice [113]. Thus, two utility functions are used, depending on user location [38]. For *outdoor* users, QoE is estimated as

$$QoE^{(FTP)}_{outdoor}(u) = \max(1, \min(5, 6.5T(u) - 0.54)) , \tag{B.25}$$

For *indoor* users, QoE is estimated as

$$QoE_{indoor}^{(FTP)}(u) = \max(1, \min(5, 6.5\frac{T(u)}{1.5} - 0.54)) \ . \tag{B.26}$$

By comparing (B.25) and (B.26), it is observed that throughput T is divided by 1.5 for the latter, reflecting that *indoor* users experience worse QoE for the same T value, since expectations of indoor users are higher.

The analytical model must only establish the relationship between HOM and T changes, which can then be translated into QoE changes. Specifically, the gradient of the objective function is computed on an adjacency basis by aggregating the impact of changes across users in the adjacency as

$$\begin{aligned} \frac{\delta\overline{QoE}_u}{\delta HOM(i,j)} &= \sum_u \frac{\delta QoE(u)}{\delta HOM(i,j)} \\ &= \sum_u [\frac{\delta QoE(u)}{\delta T(u)} \frac{\delta TH(u)}{\delta SE(u)} \frac{\delta SE(u)}{\delta SINR(u)} \frac{\delta SINR(u)}{\delta HOM(i,j)} \\ &+ \frac{\delta QoE(u)}{\delta T(u)} \frac{\delta TH(u)}{\delta BW(u)} \frac{\delta BW(u)}{\delta N_{su}(k)} \frac{\delta N_{su}(k)}{\delta A(k)} \frac{\delta A(k)}{\delta HOM(i,j)}] \ , \end{aligned} \tag{B.27}$$

where k is the serving cell for user u (i.e., either cell i or j), $BW(u)$ is the average system bandwidth assigned to the user u, $N_{su}(k)$ is the average number of simultaneous active users with user u in the cell serving (excluding inactive periods), $A(k)$ is the service area of cell k, and $SE(u)$ and $SINR(u)$ are the average spectral efficiency and signal quality of user u.

The chain rule in (B.27) reflects that any user throughput change achieved by traffic sharing is due to: a) a change in radio link conditions experienced, or b) a change in the number of available resources for the user caused by the new number of simultaneous users in the cell. To increase the robustness of the method, the gradient is approximated by estimating the impact of a large change in the HO margin of the adjacency under study (i.e., 3 dB) on the overall system QoE, $\Delta\overline{QoE}_u$.

Figure B.9 shows a flow diagram of the proposed algorithm, whose aim is to estimate the potential impact of traffic sharing on QoE at a connection level, which is then aggregated at a cell level to derive HO margin changes per adjacency. The inputs to the method are: a) user traces, including performance measurements at a connection level, b) cell traces, including instantaneous

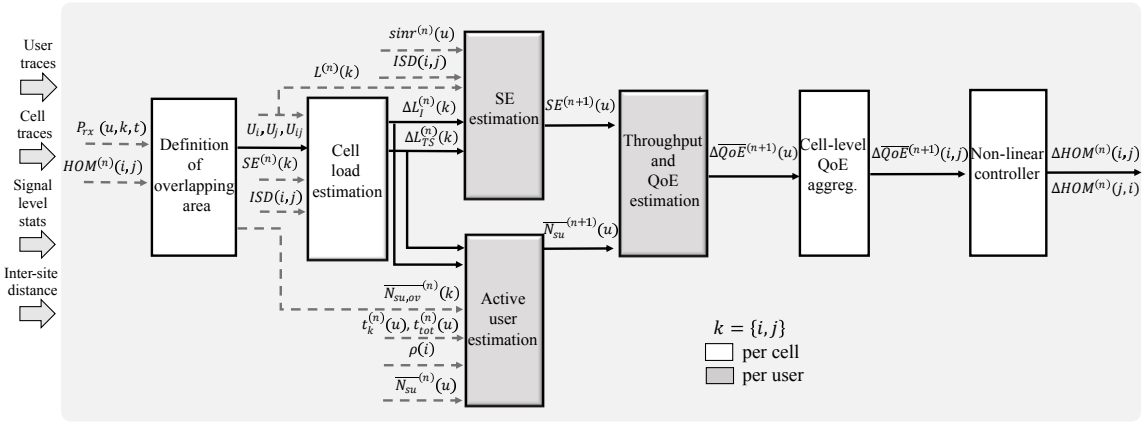


Figure B.9: Flow diagram of the computation of changes in handover margins per adjacency.

performance measurements at a cell level, c) signal level statistics, including reference signal measurements from serving and neighbor cells, and d) Inter-Site Distance (ISD), computed from site coordinates. The output is the change in HO margin in the adjacency. For clarity, variables directly taken from measurements are depicted with dashed lines, to isolate them from estimated variables, shown with solid lines. Likewise, stages (i.e., boxes in the figure) dealing with cell-level statistics are filled in white and stages dealing with connection-level stats are filled in gray. Hereafter, for brevity, k denotes both source and target cell in the adjacency, i and j. All stages in the figure are described next.

Definition of overlapping area

A first step is to estimate the amount of traffic re-distributed by changing the HO margin. To this end, users in cells i and j are classified into three sets, depending on whether they change serving cell due to traffic sharing. On the one hand, U_i and U_j denote the part of connections in cell i and j that keep served by i and j after traffic sharing. On the other hand, U_{ij} denotes the part of connections that would be re-allocated by the traffic sharing algorithm. As in [114], U_{ij} is identified precisely from pilot signal level statistics collected by base stations, as those users u fulfilling that

$$P_{rx}(u,j,t) \geq P_{rx}(u,i,t) + HOM(i,j) - 3 \quad , j \neq i \ , \tag{B.28}$$

where $P_{rx}(u,i,t)$ is the Reference Signal Received Power (RSRP) received by user u from cell i

at time t. By aggregating the time these users are in the overlapping area between adjacent cells, the method computes the average number of simultaneous connections removed by traffic steering in active periods of cell i, $\overline{N_{su,ov}}^{(n)}(i)$. The number of connections removed by traffic steering in active periods of cell j is computed in the same way but interchanging the index of cells i and j.

Cell load estimation

In this stage, changes in the average load in cell i and j load are estimated. The inputs to this stage are the sets of users, $U_x(x \in \{i, j, ij\})$, the average cell load and spectral efficiency before changes, $L^{(n)}(k)$ and $SE^{(n)}(k)$, and the ISD in the adjacency, $ISD(i, j)$. The main output of this stage are the new cell loads after traffic sharing.

If users in the overlapping area are handed over from cell i to cell j, the load of cell i decreases and that of cell j increases. However, at the same time, the interference from cell j received in cell i increases due to the load increase in the former. Thus, the spectral efficiency of cell i might decrease, causing an increase of cell load that might counteract the congestion relief effect of traffic steering. The contrary effect is observed in the spectral efficiency of cell j. To model both effects, load changes are broken down in two components as

$$\Delta L^{(n)}(k) = \Delta L_{TS}^{(n)}(k) + \Delta L_{I}^{(n)}(k) \ , \ k \in \{i, j\} \ , \tag{B.29}$$

where $\Delta L_{TS}^{(n)}(k)$ reflects the change due to the different number of users caused by the HO of users from cell i to cell j, and $\Delta L_{I}^{(n)}(k)$ reflects the change due to the different signal quality (spectral efficiency) caused by the new interference conditions.

Specifically, the load change in source cell i due to traffic steering is calculated as

$$\Delta L_{TS}^{(n)}(i) = -\sum_{u_{ij}} \Delta L^{(n)}(u_{ij}, i) \ , \tag{B.30}$$

where $L^{(n)}(u_{ij}, i)$ is the PRB utilization ratio removed from cell i by steering user u_{ij} to cell j, derived from the total connection time in the overlapping area observed in signal level measurements. Then, the load change in the target cell j due to traffic steering is estimated by rescaling the load removed from the source cell by considering the spectral efficiency in both cells as

$$\Delta L_{TS}^{(n)}(j) = \Delta L_{TS}^{(n)}(i) \frac{SE^{(n)}(i)}{SE^{(n)}(j)} \ . \tag{B.31}$$

Load changes due to interference are estimated depending on Inter-Site Distance (ISD) to differentiate between interference-limited and noise-limited scenarios. In the source cell, the interference-related term is estimated as

$$\Delta L_{I}^{(n)}(i) = \begin{cases} L^{(n)}(i) \left[\frac{SE^{(n)}(i)}{SE^{(n+1)}(i)} - 1 \right] & \text{if } ISD(i,j) \leq 1.25 \text{ km} \ , \\ 0 & \text{otherwise} \ , \end{cases} \tag{B.32}$$

where $SE^{(n)}(i)$ is the average spectral efficiency of cell i measured at loop n, and $SE^{(n+1)}(i)$ is the average spectral efficiency of cell i at loop $n+1$. In the latter, it is assumed that interference is much larger than noise (interference-limited scenario) and all interference received in cell i comes from cell j (single interferer), so that

$$SE^{(n+1)}(i) = SE^{(n)}(i) \frac{L^{(n)}(j)}{L^{(n)}(j) + \Delta L_{TS}^{(n)}(j)} \ . \tag{B.33}$$

Similarly, the interference-related term in the target cell is estimated as

$$\Delta L_{I}^{(n)}(j) = \begin{cases} L^{(n)}(j) \left[\frac{SE^{(n)}(j)}{SE^{(n+1)}(j)} - 1 \right] & \text{if } ISD(i,j) \leq 1.25 \text{ km} \ , \\ 0 & \text{otherwise} \ , \end{cases} \tag{B.34}$$

where

$$SE^{(n+1)}(j) = SE^{(n)}(j) \frac{L^{(n)}(i)}{L^{(n)}(i) + \Delta L_{TS}^{(n)}(i)} \ . \tag{B.35}$$

It should be pointed out that the single-interferer assumption in (B.32) and (B.34) is the worst case of interference-limited scenarios where traffic steering achieves the lowest congestion relief effect. Results presented later show that such an approximation has a negligible impact on method performance.

Spectral efficiency estimation

The next stage is to estimate changes in spectral efficiency per user. The input to this stage is the SINR per user $sinr^{(n)}(u)$, the ISD between cell i and cell j, $ISD(i,j)$, the average cell load before changes, $L^{(n)}(k)$, and the estimation of cell load changes $\Delta L_{TS}^{(n)}(k)$ and $\Delta L_{I}^{(n)}(k)$. The output is the estimated spectral efficiency for the next loop per user, $SE^{(n+1)}(u)$. According to an analysis carried out in the considered scenario, if the distance from cell i to cell j is less than 1.25 km, it can be assumed that interference I received from adjacent cell j is much greater than noise, so that $sinr^{(n)}(u) \simeq \frac{c}{i}^{(n)}(u)$, $\forall u \in \{U_i, U_j, U_{ij}\}$. Otherwise, spectral efficiency remains invariant. Specifically, for users served by cell i, spectral efficiency is estimated as

$$SE^{(n+1)}(u_i) \simeq \begin{cases} \log_2(1 + \frac{c}{i}^{(n)}(u_i)\frac{L^{(n)}(j)}{L^{(n+1)}(j)}) & \text{if } ISD(i,j) \leq 1.25 \text{ km} , \\ SE^{(n)}(u_i) & \text{otherwise} , \end{cases} \tag{B.36}$$

where the factor $\frac{L^{(n+1)}(j)}{L^{(n)}(j)}$ reflects the increase in interference due to neighbor cell load increase. Note that cell load estimate for next iteration, $L^{(n+1)}(k)$ can easily be calculated from the estimate of cell load changes as

$$L^{(n+1)}(k) = L^{(n)}(k) + \Delta L_{TS}^{(n)}(k) + \Delta L_{I}^{(n)}(k) . \tag{B.37}$$

Similarly, for users served by cell j, spectral efficiency is estimated as

$$SE^{(n+1)}(u_j) \simeq \begin{cases} \log_2(1 + \frac{c}{i}^{(n)}(u_j)\frac{L^{(n)}(i)}{L^{(n+1)}(i)})) & \text{if } ISD(i,j) \leq 1.25 \text{ km} , \\ SE^{(n)}(u_j) & \text{otherwise} . \end{cases} \tag{B.38}$$

Finally, for users in the overlapping area, spectral efficiency is estimated as

$$SE^{(n+1)}(u_{ij}) \simeq \begin{cases} SE^{(n)}(u_{ij}(j)) & \text{if } u_{ij}(j) \neq 0 , \\ \overline{SE^{(n)}(u_{ij}(j))} & \forall u_{ij}(j) \neq 0 \text{ otherwise} , \end{cases} \tag{B.39}$$

where users in the overlapping area are divided in two different groups: those who were handed over from cell i to cell j in the optimization loop n, $u_{ij}(j) \neq 0$, and those who were not $u_{ij}(j) = 0$.

Active user estimation

The fourth stage addresses the number of simultaneous users in the next loop. This variable has to be estimated on a per-user basis to obtain reliable estimates of user throughput and QoE. The input to this stage are the estimates of cell load changes, together with the average number of simultaneous users in the overlapping area during active periods, $\overline{N_{su,ov}}^{(n)}(i)$, the time spent by the user in cell $k = \{i, j\}$, $t_k^{(n)}(u)$, the total connection duration, $t_{tot}^{(n)}(u)$, the activity ratio of cell i (measured as the ratio of active TTIs), $\rho(i)$, and the average number of simultaneous users when each user is transmitting in the current loop, $\overline{N_{su}}^{(n)}(u)$. The output is the number of active users with each user, $\overline{N_{su}}^{(n+1)}(u)$.

The expected number of simultaneous users in source cell i in the next loop for a user u is calculated from the value in the past loop as

$$\overline{N_{su}}^{(n+1)}(u_i) = \overline{N_{su}}^{(n)}(u_i) - \overline{N_{su,ov}}^{(n)}(i) + \overline{\Delta N_{su,I}}^{(n)}(i) \ , \tag{B.40}$$

where $\overline{N_{su,ov}}^{(n)}(i)$ captures the decrease in the number of active users due to congestion relief achieved by handing over users, and $\overline{\Delta N_{su,I}}^{(n)}(i)$ reflects the increase in the number of active users due to the loss of spectral efficiency from a higher interference. To estimate $\overline{N_{su,I}}^{(n)}(i)$, a empirical regression analysis is performed to find the expression relating cell load L with the number of active users N_{su} in a cell, $N_{su}(L)$. In practice, such an analysis can easily be done with performance counters aggregated at a cell level stored in the network management system. In this work, this analysis is carried out by simulations in which cells carry different loads with different number of simultaneous users. The resulting regression curve is the exponential function

$$N_{su}(L) = 1.851e^{1.505L} \ . \tag{B.41}$$

The sensitivity of the number of active users due to increments of cell load is obtained by differentiating the above exponential function, as

$$\frac{\delta N_{su}}{\delta L}(L) = 1.851 \cdot 1.505e^{1.505L} \ . \tag{B.42}$$

The latter is used to quantify the increase in the number of active users due to interference from the increase of cell load due to traffic steering, $\Delta L_{TS}^{(n)}(i)$, and interference, $\Delta L_I^{(n)}(i)$, as

$$\overline{\Delta N_{su,I}}^{(n)}(i) = \Delta L_I^{(n)}(i)\frac{\Delta N_{su}}{\Delta L}(L^{(n)}(i) + \Delta L_{TS}^{(n)}(i)) \ . \tag{B.43}$$

Note that changes in the number of users in source cell i due to traffic steering can be directly taken from network measurements. In contrast, changes in target cell j have to be estimated. For this purpose, it must be taken into account that handed-over users might require a different amount of resources in the new cell, because of new radio link conditions. This can easily be taken into account by multiplying by the ratio of the average cell spectral efficiency in the old and target cell. To account for this effect, the new average number of active users in target cell j is estimated as

$$\overline{N_{su}}^{(n+1)}(u_j) = \overline{N_{su}}^{(n)}(u_j) - \overline{N_{su,ov}}^{(n)}(i)\rho(i)\frac{SE^{(n)}(i)}{SE^{(n)}(j)} + \overline{\Delta N_{su,I}}^{(n)}(j) \ , \tag{B.44}$$

following the same structure as (B.40). $\rho(i)$ is the activity ratio of cell i (measured as the ratio of active Time transmission Intervals, TTIs). Similarly to (B.43), $\overline{\Delta N_{su,I}}^{(n)}(j)$ is estimated as

$$\overline{\Delta N_{su,I}}^{(n)}(j) = \Delta L_I^{(n)}(j)\frac{\Delta N_{su}}{\Delta L}(L^{(n)}(j) + \Delta L_{TS}^{(n)}(j)) \ , \tag{B.45}$$

using the same derivative function as in (B.42). Finally, for users in the overlapping area u_{ij} that already performed HO from i to j in the previous loop (without traffic steering), the number of simultaneous users in loop $n+1$ is estimated as a weighted average of the measured number of simultaneous users during the segments of the connections in cell i and j in the previous loop, u_i and u_j, weighted by the time in each cell i and j, as

$$\overline{N_{su}}^{(n+1)}(u_{ij}) = \frac{t_i^{(n)}(u_{ij})}{t_{tot}^{(n)}(u_{ij})}\overline{N_{su}}^{(n+1)}(u_i) + \frac{t_j^{(n)}(u_{ij})}{t_{tot}^{(n)}(u_{ij})}\overline{N_{su}}^{(n+1)}(u_j) \ , \tag{B.46}$$

where $t_i^{(n)}(u_{ij})$ and $t_j^{(n)}(u_{ij})$ is the time that user u_{ij} spent in cells i and j, respectively, and $t_{tot}^{(n)}(u_{ij})$ is the total time user u is served by both cells.

Throughput and QoE estimation

Once the number of simultaneous users has been estimated per user for the next optimization loop $n+1$, $\overline{N_{su}}^{(n+1)}(u)$, as well as spectral efficiency, $SE^{(n+1)}(u)$, the output of this stage is the

best HOM change $\Delta QoE^{(n+1)}(u)$. For this purpose, user throughput variations are first estimated, so that, QoE changes can then be computed on a per-user basis.

The estimation of user throughput after a HOM change is calculated as

$$TH^{(n+1)}(u) = SE^{(n+1)}(u)BW^{(n+1)}(u) = SE^{(n+1)}(u)\frac{N_{PRB}}{\overline{N_{su}}^{(n+1)}(u)},\ \forall u \in U_i, U_j, U_{ij}\ , \tag{B.47}$$

where N_{PRB} is the system bandwidth and $SE^{(n+1)}(u)$ is considered as throughput per PRB.

Throughput values are easily translated into QoE with (B.25)-(B.26). Then, the change in user QoE due to HOM changes is estimated as

$$\Delta QoE^{(n+1)}(u) = QoE^{(n+1)}(u) - QoE^{(n)}(u),\ \forall u \in U_i, U_j, U_{ij}\,. \tag{B.48}$$

Cell level QoE aggregation

The network average QoE variation due to $HOM(i,j)$ change is calculated as

$$\Delta QoE^{(n+1)}(i,j) = \frac{\sum\limits_{u \in U_i, U_j, U_{ij}} \Delta QoE^{(n+1)}(u)}{N_u(i) + N_u(j)}\ , \tag{B.49}$$

i.e., the aggregation of every individual QoE modification divided by the number of users in cells i and j (which is maintained in iterations n and $n+1$ in this work).

The above analysis considers the case when $HOM(i,j)$ is decreased. The opposite case, when $HOM(i,j)$ is increased, can be evaluated by analyzing the opposite side of the adjacency, where $HOM(j,i)$ is decreased to satisfy (B.2). The whole estimation process explained above must be repeated for a similar decrease of $HOM(j,i)$ by 3 dB, resulting in $\Delta QoE^{(n+1)}(j,i)$. Therefore, two estimations of average QoE changes are carried out per adjacency: $\Delta QoE^{(n+1)}(i,j)$ and $\Delta QoE^{(n+1)}(j,i)$, corresponding to both HOM movements, i.e., reducing cell i or j service areas, respectively. OE algorithm discerns which option obtains the highest ΔQoE value (i.e., move traffic from cell i to j, or vice-versa). If both movements degrade the overall QoE in the adjacency, no HOM change is made (gradient ascent rule).

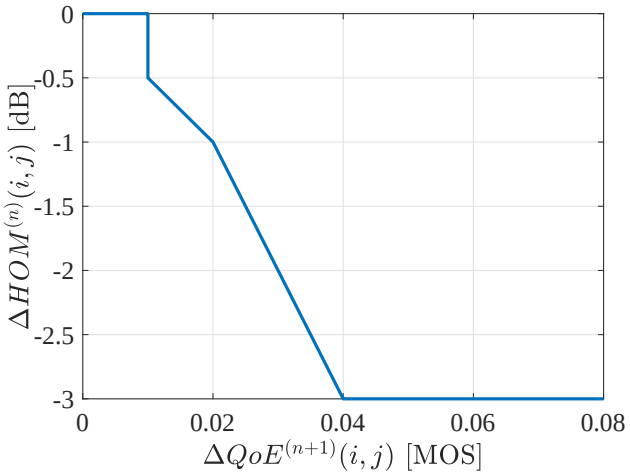


Figure B.10: Proportional controller per adjacency (i, j).

Non-linear controller

Finally, to define the magnitude of HOM changes, the incremental controller shown in Figure B.10 is used. It is observed that the controller includes a gain scheduling algorithm modifying the feedback loop gain to control the trade-off between convergence speed and system stability. A coring operation ensures that no changes are implemented when expected QoE benefits are below 0.01. Thus, the control system reaches equilibrium earlier. Beyond that value, a larger slope is used to favor adjacencies with larger expected QoE benefits. To avoid instabilities, the maximum HOM change per iteration is limited to 3 dB.

The figure shows the case when $\Delta QoE^{(n+1)}(i,j) > 0$ and $\Delta QoE^{(n+1)}(i,j) > \Delta QoE^{(n+1)}(j,i)$ (i.e., decreasing $HOM(i,j)$ is more beneficial than decreasing $HOM(j,i)$). In this case, the controller computes the HOM change in adjacency (i,j), $\Delta HOM^{(n)}(i,j)$, so that $HOM^{(n+1)}(i,j) = HOM^{(n)}(i,j) + \Delta HOM^{(n)}(i,j)$. To maintain hysteresis, $HOM^{(n+1)}(j,i) = HOM^{(n)}(j,i) - \Delta HOM^{(n)}(i,j)$. In the opposite scenario (i.e., decreasing $HOM(j,i)$ is more beneficial than decreasing $HOM(i,j)$), the controller works the same, just interchanging indexes i and j.

Assessment methodology

In this section, the proposed optimization algorithm is validated with simulations. For clarity, the experiment set-up is first presented, and results are shown later.

Experiment set-up

The traffic model implemented in the scenario is a FTP service reflecting the download of a file whose size follows a log-normal distribution of average 2 MB, standard deviation 0.25. In the simulated scenario, the percentage of indoor users per cell depends on cell location. For this purpose, 36 % of cells are categorized as urban, 46 % as suburban and 18 % as rural, based on the predominant land use in its service areas. Then, it is assumed that urban cells have 70 % of indoor users, suburban cells have 50 % of indoor users and rural cells have 10 % of indoor users. Spatial traffic distribution at a cell level follows the same profile as in the live network, taken from real traffic statistics. With default HOM settings, the average cell load in the network is $\overline{U(i)} = 61$ % and $\overline{QoE}_u = 3.08$.

To assess OE algorithm, its performance is compared with that of LB, EB-C and TB, already defined in Section B.2.4. For all algorithms, 14 optimization loops of 30 minutes of network time are simulated. For all methods, the main figure of merit is $\overline{QoE}_u$, defined in (B.23).

To verify the ability of methods to reduce each indicator imbalance, the figures of merit defined in (B.15), (B.21) and (B.18) are used.

Results

Figure B.11 illustrates $\overline{QoE}_u$ evolution for LB, TB, EB-C and OE along the 14 optimization loops. It is observed that the $\overline{QoE}_u$ trend for OE is absolutely different to that shown by LB, TB and EB-C. Initially, with the default value of HOM (3 dB) for all adjacencies in the network, $\overline{QoE}_u = 3.08$. At the end of the optimization process, OE reaches a value of $\overline{QoE}_u = 3.3$, while LB, TB and EB-C achieve 2.56, 2.48 and 2.63, respectively, that is, around 0.6 MOS points higher than the second best algorithm (EB-C). More importantly, OE increases $\overline{QoE}_u$ more than 0.2 MOS points compared to the initial state (3.3 versus 3.08), while the rest of the methods decrease $\overline{QoE}_u$ in 0.52, 0.6 and 0.43 MOS points, respectively. As expected, OE reaches the best $\overline{QoE}_u$ figures due to its analytical formulation that considers signal, congestion and interference issues, and considers optimality.

It is worth noting that the global user QoE improvement is not achieved at the expense of degrading the QoE for the worst users. To confirm this statement, Figure B.12 compares the cumulative distribution function of user QoE in the scenario obtained by each method against that with default settings (initial curve), used as a baseline. In the Initial situation, 35 % of the users have the lowest possible QoE figure ($QoE(u) = 1$), even when 40 % of them experience the highest

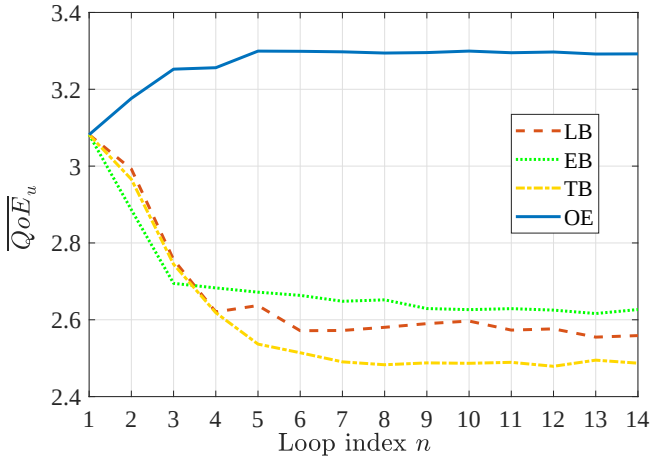


Figure B.11: Evolution of $\overline{QoE}_u$.

QoE value ($QoE(u) = 5$), due to the uneven spatial distribution of connections in the scenario.

Unexpectedly, all balancing schemes increase the ratio of users with low QoE values. LB and TB balance traffic indicators regardless of the traffic mix. Likewise, in EB-C, the average user QoE does not improve because balancing average cell QoE, $QoE(i)$, improves the QoE of the worst users in the worst cells at the expense of deteriorating the experience of the best users in neighbor cells. On the contrary, OE reduces the number of fully unsatisfied users from 35 to 30 %, while increasing the number of fully satisfied users from 41 to 42 %. Most percentiles maintain these differences (e.g., the 35^{th} percentile of $QoE(u)$ is 1.22 in the initial configuration, 1 for LB, EB-C and TB, and 1.65 for OE).

Table B.6 presents different performance indicators at the beginning and end of the optimization process for all the considered methods (columns LB, TB, EB-C and OE). For the sake of clarity, the best algorithm per figure of merits is highlighted in gray. As expected, $\overline{U_{imb}(i)}$, $\overline{T_{imb}}$ and $\overline{QoE}_u$ reach their best performance for the algorithms they were designed for (i.e., LB, TB and OE, respectively). In particular, LB reaches the lowest average cell load imbalance, $\overline{U_{imb}} = 7.43\,\%$, and TB reaches the lowest average throughput imbalance, $\overline{T_{imb}} = 0.22$ Mbps.

Nevertheless, surprisingly, the minimum value of $\overline{QoE_{imb,c}}$ is achieved by TB ($\overline{QoE_{imb,c}} = 0.3$), while for EB-C, $\overline{QoE_{imb,c}} = 0.53$, at the expense of a higher degradation in $\overline{QoE}$ (2.64 versus 2.88 for TB and EB-C, respectively). This unexpected behavior is due to the limits imposed by utility functions (3.22)-(3.23), which reach saturation levels (QoE = 1 or 5) with very different values of $T(u)$ (0.237 and 0.853 Mbps, respectively, for *outdoor* users). Different values for $T(i)$

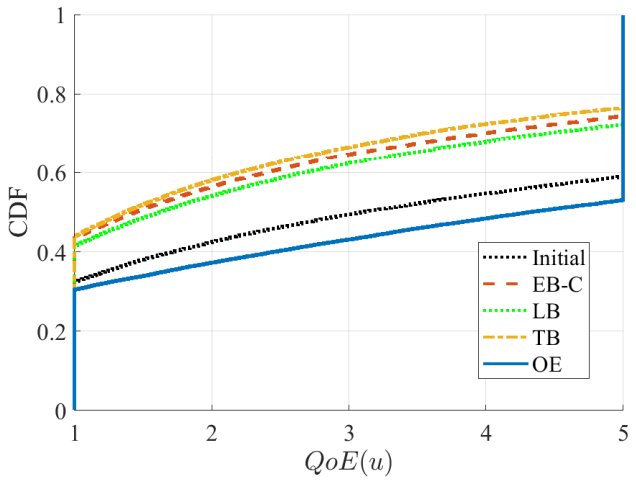


Figure B.12: Cumulative distribution function of user QoE.

Table B.6: Main performance indicators.

	Initial	**LB**	**TB**	**EB-C**	**OE**
$\overline{U_{imb}}$ [%]	18.84	7.43	10.3	11.22	15.14
$\overline{T_{imb}}$ $[Mbps]$	0.90	0.36	0.22	0.36	0.85
$\overline{QoE_{cell,imb}}$	0.71	0.55	0.3	0.53	0.71
$\overline{QoE}_u$	3.08	2.56	2.48	2.63	3.3
$\overline{QoE}$	3.73	2.62	2.64	2.88	3.84

and $T(j)$ may result in similar average cell QoE values, $QoE(i)$ and $QoE(j)$ (for example, two cells with $T(i) = 0.05$ Mbps and $T(j) = 0.26$ Mbps on average would experience $QoE(i) = 1$ and $QoE(j) = 1.1$, respectively). Under these circumstances, TB would continue degrading $QoE(j)$, while improving $\overline{QoE_{imb,c}}$ (up to 0.3 MOS points). Thus, $\overline{QoE}_u$ achieved by EB-C is 0.24 MOS points higher than that achieved by TB. When having different services mixes for different cells, different utility functions are used to calculate average cell QoE, the relationship between *throughput* and QoE is not as limited as it is in the particular scenario, and both objectives (balancing $T(i)$ and $QoE(i)$) would follow different trajectories [115].

B.2.6. Conclusions

In this section, two QoE-driven traffic steering algorithms have been proposed: EB and OE. The aim of EB is to balance QoE among cells and cells and services throughout the network. In contrast, the aim of OE is to optimize average user QoE across the system. Both algorithms follow

an iterative approach that modifies HOMs to achieve their goals.

EB presents two variants, EB-C and EB-CS, depending on whether HOMs are modified on an adjacency basis, or on an adjacency and service basis. Method assessment has been carried out by means of simulations in a dynamic system-level LTE simulator. Results have shown that, in the considered realistic scenario, the average cell QoE imbalance is reduced in 0.22 and 0.26 MOS points with EB-C and EB-CS, respectively.

In contrast, OE relies on an analytical approach to follow a gradient ascent approach that ensures that HOM changes always improve global system QoE. For this purpose, the impact of a HOM change in the system QoE is estimated with a closed-form network performance model. Results have shown that, in the considered scenario, OE increases average global system QoE in the network in a 11.45 %, outperforming classical balancing algorithms.

B.3. QoE-driven Mobility Robustness Optimization

This Section is organized as follows. Section B.3.1 reviews the technique used in this section. Section B.3.2 describes the handover performance events considered in this work. Section B.3.3 presents the proposed QoE-driven MRO algorithm, whose performance is analyzed in Section B.3.4. Finally, Section B.3.5 discusses implementations issues, and Section B.3.6 summarizes the main conclusions.

B.3.1. Review of the technique

One of the most important processes to manage in a mobile network is the Handover procedure. HO is in charge of providing mobile users with seamless connectivity while they move across the network. If HO parameters are not well configured, instabilities arise, causing unnecessary HOs (Ping-Pong effect, PP) or Radio Link Failures (RLF) (i.e., too early/late HOs). To avoid these issues, Mobility Robustness Optimization (MRO) is a self-optimization feature that automatically tunes HO parameters to improve HO performance (i.e., minimize PP and RLF) [117].

In this Section, a novel QoE-aware ML-based MRO algorithm is proposed for LTE systems. This work is a follow up of [28]. As in [28], the proposed adaptive HO scheme aims to reduce the number of PP and RLF in the network by displacing HO trigger location through changes in HO Margin (HOM) and Time-To-Trigger (TTT) parameters, widely used in many previous works. Likewise,

adaptation is achieved by combining well-known RL and ANN techniques. However, unlike previous works, the scheme proposed here adds QoE concerns to MRO by also considering the QoE of cell edge users affected by the HO process as an input. The resulting scheme is validated in a dynamic system-level simulator implementing a realistic macrocellular LTE scenario.

The main contributions of this Section are: a) uncovering the limitations of traditional MRO schemes from a QoE perspective, b) the inclusion of QoE criteria in an adaptive HO scheme to increase user QoE at cell edge while decreasing RLF and PP by modifying handover margins and Time To Trigger, and c) the validation of the algorithm via simulations in a realistic macrocellular LTE scenario.

B.3.2. Handover performance events

As explained above, a HO is triggered when (B.50) is fulfilled.

$$P_{rx}(j) - P_{rx}(i) \geq HOM(i,j)\,, \tag{B.50}$$

Nevertheless, if HOM and TTT are not well configured, one of the following events can occur.

1 RLF due to Late HandOver (LHO): it happens when a user is moving from a serving cell i to a target cell j, but HO is triggered too late (or even not triggered at all). In this case, the pilot signal level from cell i drops below a certain threshold during a specific time window and a RLF occurs in cell i. As a result, the user is disconnected from cell i and reconnected to cell j some time later.

2 RLF due to Early HandOver (EHO): it takes place when a user is moving from a serving cell i to a target cell j, but HO is triggered too early. After HO, the pilot signal level from cell j drops below a certain threshold during a specific time window and a RLF occurs. As a result, the user is disconnected from cell j and reconnected to cell i (or another cell).

3 Ping-Pong (PP): when a user is moving from a serving cell i to a target cell j, but HO is triggered early, the corresponding HO algorithm will try to hand over the user back to cell i within a particular time window after the first HO took place. This is an unnecessary HO causing useless increase in signaling load.

LHO, EHO and PP are mutually exclusive. If a HO is performed and none of the above occurs,

HO is considered successful (denoted as SHO). Only A3 event-based HOs are considered in this work [126].

MRO schemes modify HOM and TTT values to maximize the ratio of SHO (or, conversely, minimize LHO, EHO and PP). Ideally, the optimal configuration should ensure that HO occurs at that point where the pilot signal level (RSRP) from the serving and target cells are comparable and both high enough. This is achieved with low values of HOM and TTT. When tuning HOM and TTT, a trade-off exists between LHO, EHO and PP. The lower HOM and TTT, the faster the HO is triggered, ensuring that the user is always connected to the best cell, but the more likely that a EHO and PP occurs. In contrast, the larger HOM and TTT, the longer the HO is delayed, avoiding EHO and PP, but the user remains connected to the source cell offering worse signal level than the target cell, which temporarily degrades link performance and might end up in a LHO.

The above signal impairments may decrease user throughput and increase packet delay, especially at cell edge, but these changes may not be fully perceived by the user. This fact points out that there is not a direct relationship between radio link performance and QoE [115], but a more complex one. Thus, minimizing HO failures and PP does not necessarily lead to an increase in the QoE of handed-over users. This Section proposes means to learn the HO settings that are optimal not only for link robustness, but that concurrently maximize QoE both before and after the handover event.

B.3.3. QoE-aware MRO algorithm

In this section, a new adaptive QoE-aware MRO algorithm based on reinforcement learning is presented. For clarity, the QL adaptation framework is introduced first. Then, the baseline QL MRO algorithm only driven by HO performance indicators is explained, hereafter denoted as Quality-MRO (Q-MRO) [28]. Finally, the proposed algorithm adding QoE criteria, referred to as Experience MRO (E-MRO), is described.

Q-Learning framework

QL is a model-free reinforcement learning algorithm to solve learning problems. It is selected here due to its ability to learn and improve system performance through experience. A QL problem is defined by the triplet (X, A, r), where X and A are the sets of all possible system states and actions, respectively, and $r : X \cdot A \rightarrow R$ is the *reward* function, representing the reward (i.e.,

performance improvement) obtained by executing each action in a given system state. In this work, a state is defined by the combination of variables determining the specific radio environment of each HO event (e.g., user speed, cell load, interference...). Likewise, an action is a particular setting of HO parameters. Both states and actions can only take discrete values.

Every time some event n occurs at time t_n, a QL (a.k.a. learning) agent checks the benefit from executing some action $a_{t_n} \in A$ with the system in state $x_{t_n} \in X$ by computing the associated reward, $r(x_{t_n}, a_{t_n})$. From these observations, a function, $Q(x, a)$, is derived with the expected reward from action a in state x. Such a function is approximated by recursively computing reward estimates with the Bellman equation, as

$$Q^{(n+1)}(x, a) = (1 - \alpha)Q^{(n)}(x, a) + \alpha r(x_{t_n}, a_{t_n}) \,, \tag{B.51}$$

where α is the learning rate. The above estimates are stored in the so-called Q-table.

At the end of the optimization process, the Q-table is considered as the Q function, so that the best action for every state, $a_{max}(x)$, can easily be determined as

$$a_{max}(x) = \max_{a} \left(Q^{(n_{end})}(x, a) \right) \quad , \tag{B.52}$$

which is the aim of the QL process. Superscript n_{end} reflects the last event.

The QL optimization algorithm is an iterative scheme that requires a time division in slots, hereafter referred to as Action Intervals (AI). During an AI, (a, x, r) values are collected for every event and the Q-table is updated accordingly every time an event takes place. Once the AI ends, the best action for every state, $a_{max}(x)$, is selected and used for the next AI (in this work, 30 seconds of network time). Thus, a best action per state can be defined for every AI, $a_{max}(x, n_{AI})$.

The dilemma of how much effort to spend on testing new actions (probably suboptimal) or take those already explored (getting high reward) is known as the *exploration-exploitation* trade-off [127]. Such a trade-off is controlled by a decaying Epsilon-greedy (ε-greedy) policy, which indicates to the learning agent how much to explore and how much to exploit as time goes by. The learning agent has to explore all the possible states and actions while training and exploits this knowledge during the exploitation phase. Epsilon ε value varies from 1 to 0. Figure B.13 illustrates a typical ε evolution with n_{AI}. The exploration phase lasts for $N_{explore}$ AIs, the training phase lasts for N_{train} AIs, and the final exploitation phase lasts for $N_{exploit}$ AIs.

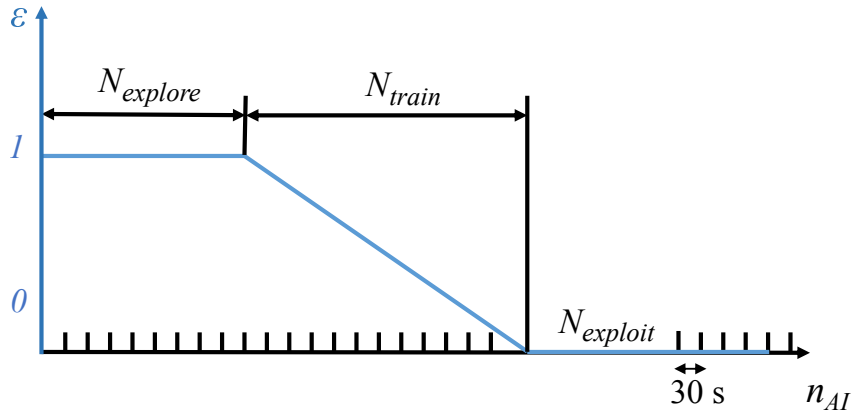


Figure B.13: Evolution of ε parameter in the proposed Q-learning scheme.

The basic structure of a generic QL scheme is summarized in Algorithm B.1. The main loop represents iterations across AIs. Note that the same structure is shared by all MRO algorithms tested in this work. Algorithms only differ in the definition of the reward function, r, as will be presented in next sections.

Q-MRO algorithm

The Q-MRO algorithm was already proposed in [28]. It is used here as a benchmark. The core of Q-MRO is the definition of the reward function, r (i.e., how good or bad a particular HO event has performed), calculated as

$$r(x_{t_n}, a_{t_n}) = -w_{RLF} X_{EHO}(n) - w_{RLF} X_{LHO}(n) - w_{PP} X_{PP}(n)\,, \tag{B.53}$$

where $X_{EHO}(n) = 1$ if event n occurring in t_n is categorized as an EHO ($X_{EHO}(n) = 0$, otherwise). Analogously, $X_{LHO}(n) = 1$ or $X_{PP}(n) = 1$ if event n in t_n is a LHO or PP, respectively (0, otherwise).

Algorithm B.1 Structure of Q-MRO algorithm.

Input: states x_{t_n} and actions a_{t_n} $\varepsilon(n_{AI})$
Output: actions for next iteration $a(x, n_{AI}+1)$

FOR $n_{AI} = 1, 2, ...$

calculate $\varepsilon(n_{AI})$
FOR each event n occurring in $t_n \in n_{AI}$

calculate $r(x_{t_n}, a_{t_n})$, with (B.53)
update element in Q-table $Q^{(n+1)}(x_{t_n}, a_{t_n})$, with (B.51)

END

FOR every $x \in X$

IF rand () $> \varepsilon(n_{AI})$
$a(x, n_{AI}+1) = a_{max}(x, n_{AI}+1) = \underset{a}{max}\left(Q^{(n)}(x, a)\right)$, with (B.52)
ELSE
select a random action a for $a(x, n_{AI}+1)$, with (B.52)
END

END

$n_{AI} = n_{AI} + 1$

END

E-MRO algorithm

Unlike Q-MRO, the E-MRO algorithm proposed here not only aims to reduce RLF and PP, but also to optimize cell-edge user QoE. For this purpose, HOM and TTT are adjusted on an adjacency basis with the aim of increasing the average QoE of cell-edge users. In this work, the time window for evaluating the QoE of a user experiencing a HO event comprises 1 second before and after the HO trigger point. Note that event n is associated to the particular user u experiencing that n^{th} HO event, so that indexes n and u_n can be interchanged.

The reward function for E-MRO, differently to Q-MRO, is computed as the sum of three components as

$$r(x_{t_n}, a_{t_n}) = r_{radio}(x_{t_n}, a_{t_n}) + r_{QoE_{step}}(x_{t_n}, a_{t_n}) + r_{QoE_{prev}}(x_{t_n}, a_{t_n}) \ , \tag{B.54}$$

where r_{radio} is a reward related to radio robustness, $r_{QoE_{step}}$ is a reward due to the change in QoE caused by the HO event, and $r_{QoE_{prev}}$ is a negative reward (i.e., a penalty) due to a bad QoE before

HO, defined as

$$r_{radio}(x_{t_n}, a_{t_n}) = -w_{RLF}X_{EHO}(n) - w_{RLF}X_{LHO}(n) - w_{PP}X_{PP}(n)\,,$$
$$r_{QoE_{step}}(x_{t_n}, a_{t_n}) = \max(-1, \min(1, (QoE^A(u_n) - QoE^B(u_n))))\,,$$
$$r_{QoE_{prev}}(x_{t_n}, a_{t_n}) = \max(-1, \min(0, (\frac{QoE^B(u_n)}{QoE^B(w,x)} - \frac{\overline{QoE^B(w,x)}}{\overline{QoE^B(w,x)}}))). \quad \text{(B.55)}$$

The first term, r_{radio}, shows the HO performance from the radio perspective, $r_{QoE_{step}}$ focuses on maximizing the difference between the QoE experienced by the user after and before the HO, $QoE^A(u_n)$ and $QoE^B(u_n)$, respectively and $r_{QoE_{prev}}$ in (B.54) avoids wrong HO settings by penalizing situations when QoE before HO is lower than average, $\overline{QoE^B(w,x)}$. Specifically, $\overline{QoE^B(w,x)}$ is the average QoE before HO from any user w performing a HO between any cell pair in the scenario within the same state x as the state in the adjacency of user u_n.

Two variants of E-MRO are proposed, denoted as EQ-MRO and EN-MRO. EQ-MRO follows the above-described process of updating a Q-table with Q-values as described in (B.51), with $\alpha = 0.005$. Alternatively, EN-MRO replaces the computation of Q-values per action in each row of the Q-table (representing states) by an ANN as a function approximator [128]. This work uses a shallow ANN [1] with two hidden layers as a substitute of each row of the Q-table [129].

B.3.4. Performance analysis

The above-described algorithms are tested in the same dynamic system-level simulator used in the previous chapter with slight differences to reduce computational load. Thus, time resolution is set to 20 ms and system bandwidth is reduced to 5 MHz (25 PRB).

Experimental Methodology

States and actions spaces

All the tested algorithms work over the same set of states and actions. Each state is modeled as a tuple {s,v,d,l}. Table B.7 illustrates the meaning and possible values for each parameter.

[1] A shallow neural network has up to two hidden layers (opposite to deep neural networks, comprising multiple hidden layers).

Table B.7: Parameter defining state space

Parameter	Possible values	Index
Service s	{FTP,VIDEO,WEB}	{1,2,3}
User velocity v **[km/h]**	{30,70}	{1,2}
Inter-site distance d **[km]**	{≤1.25,>1.25}	{1,2}
Target cell load l **[%]**	{≤70,>70}	{1,2}

Table B.8: State space.

State x	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
s	1	1	1	1	1	1	1	1
v	1	2	1	2	1	2	1	2
d	1	1	2	2	1	1	2	2
l	1	1	1	1	2	2	2	2
State x	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**
s	2	2	2	2	2	2	2	2
v	1	2	1	2	1	2	1	2
d	1	1	2	2	1	1	2	2
l	1	1	1	1	2	2	2	2
State x	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**
s	3	3	3	3	3	3	3	3
v	1	2	1	2	1	2	1	2
d	1	1	2	2	1	1	2	2
l	1	1	1	1	2	2	2	2

Service s indicates the type of service (i.e., FTP, VIDEO or WEB). Parameter v denotes user speed, with 30 or 70 km/h as possible values. These values model car users moving in streets or in open roads, which are more affected by HO settings than slow moving users. Finally, d is the Inter-Site Distance (ISD), used to differentiate between close and far neighbor cells. Specifically, a threshold of 1.25 km is used for labeling an adjacency as close or far.

With the above configuration, the number of states is $3 \cdot 2 \cdot 2 \cdot 2 = 24$ states. For an easier analysis, Table B.8 enumerates states with a variable x, ranging from 1 up to 24, together with the corresponding parameter labels.

For each state, a total of 45 possible actions are considered, corresponding to the combination of 15 possible values of HOM (from -7 dB to +7 dB in steps of 1 dB) and 3 values of TTT (40, 100 and 256 ms). For space reasons, Table B.9 only presents a subset of 15 actions, a, with $HOM \in \{-2, -1, 0, 1, 2\}$ dB and $TTT \in \{40, 100, 256\}$ ms, which will be used later.

Table B.9: Action space mapping.

Action a	**1**	**2**	**3**	**4**	**5**
HOM [dB]	-2	-1	0	1	2
TTT [ms]	40	40	40	40	40
Action a	**6**	**7**	**8**	**9**	**10**
HOM [dB]	-2	-1	0	1	2
TTT [ms]	100	100	100	100	100
Action a	**11**	**12**	**13**	**14**	**15**
HOM [dB]	-2	-1	0	1	2
TTT [ms]	256	256	256	256	256

Experiments

Three experiments of increasing complexity are carried out. The first experiment intends to show Q-MRO limitations, i.e., how neglecting QoE issues in the parameter tuning process leads to bad QoE performance. Then, in the second experiment, the proposed E-MRO schemes are compared with the traditional Q-MRO in a naive scenario of a single service. Finally, the third experiment compares the methods in a complete scenario with all the services.

In the first experiment, a simple experiment is defined with only one state in the network, characterized by one service (s = FTP) and a fixed user mobility (v = 30 km/h), while ISD and target cell load are not classified. The action space A involves only one dimension, HOM values, in the range -7 dB to 7 dB adjusted in steps of 1 dB. This setup results in 15 possible action values and a Q-table defined as a 1x15 array. Thus, TTT is fixed to 40 ms, which is the minimum non-zero value according to 3GPP [125], to show the impact of different HOM settings.

A classical Q-MRO algorithm is used in the first experiment with rewards defined as in (B.53). The initial exploration period is set to $N_{explore} = 100$ AIs (50 minutes of network time) followed by a training phase $N_{train} = 360$ AIs (3 hours) and an exploitation phase of $N_{exploit} = 240$ AIs (2 hours), long enough to ensure adequate performance assessment.

Different figures of merit are monitored per AI during the optimization process. The first metric is the average cell edge QoE, defined as

$$\overline{QoE_{edge}} = \frac{1}{2N_{u_{HO}}} \sum_{u} (QoE^{A}(u) + QoE^{B}(u)) \ , \tag{B.56}$$

where $N_{u_{HO}}$ is the number of users experiencing a HO event (i.e, EHO, LHO, PP or SHO) during

an AI.

Another important indicator is the average QoE for all users in the network, considering the whole connection, at the end of the optimization process, $\overline{QoE}$, defined as

$$\overline{QoE} = \frac{1}{N_u} \sum_{\forall u} QoE(u) \ . \tag{B.57}$$

During the optimization process, user QoE, $QoE(u)$, must be calculated by aggregating performance measurements during a temporal window around the HO event. $QoE(u)$ for FTP and WEB services are calculated as in (B.5) and (B.6), which requires estimating average user throughput during the time window when QoE is assessed (i.e., 1 second around the HO event).

Additionally to QoE indicators, average user throughput, $\overline{T}$, is also used as a performance indicator, and defined as

$$\overline{T} = \frac{1}{N_u} \sum_{\forall u} T(u) \ , \tag{B.58}$$

where $TH(u)$ is the average throughput of user$u$. Note that index u in (B.57) and (B.58) denotes all users in the scenario, and not only those users experiencing a HO, as in (B.56). Thus, $\overline{T}$ and $\overline{QoE}$ are global system indicators, while $\overline{QoE_{edge}}$ assesses cell edge users. Also note that throughput figures are intermediate indicators to understand algorithm behavior, whereas QoE indicators reflect the ultimate performance of MRO techniques.

Other four metrics are the traditional MRO performance indicators reflecting the ratio of LHO, EHO, PP and SHO, as

$$LHO(n_{AI}) \ [\%] = 100 \frac{N_{LHO}(n_{AI})}{N_{events}(n_{AI})} \ , \tag{B.59}$$

$$EHO(n_{AI}) \ [\%] = 100 \frac{N_{EHO}(n_{AI})}{N_{events}(n_{AI})} \ , \tag{B.60}$$

$$PP(n_{AI}) \ [\%] = 100 \frac{N_{PP}(n_{AI})}{N_{events}(n_{AI})} \ , \tag{B.61}$$

$$SHO(n_{AI})\ [\%] = 100\frac{N_{SHO}(n_{AI})}{N_{events}(n_{AI})} = 100\frac{N_{events}(n_{AI}) - N_{LHO}(n_{AI}) - N_{EHO}(n_{AI}) - N_{PP}(n_{AI})}{N_{events}(n_{AI})}\ , \tag{B.62}$$

where $N_{LHO}(n_{AI})$, $N_{EHO}(n_{AI})$, $N_{PP}(n_{AI})$ and $N_{SHO}(n_{AI})$ are the number of LHO, EHO, PP and SHO during the n_{AI} AI, and N_{events} is the number of events (i.e., HOs) during the n_{AI} AI.

As a result for the first experiment, the best actions (HOM values) found by Q-MRO will be selected as a reduced action space for the second experiment.

In a second experiment, the aim is to find the best settings for both HOM and TTT parameters on an adjacency basis in a simple scenario with a single service. To this end, the space of states is enlarged by including all possible values for v, d and l. Yet, only FTP service is still considered, so that only eight states are simulated (states from 1 to 8 in Table B.8). Likewise, the action space is limited to 15 possible actions, defined by the 5 best HOM values in the first experiment and 3 possible TTT values (i.e., 40, 100 and 256 ms). With these states and actions, three MRO approaches are compared: Q-MRO, EQ-MRO and EN-MRO, with Q-MRO considered as a benchmark [28]. For this second experiment, $N_{explore} = 150$ AIs (75 minutes), $N_{train} = 480$ AIs (4 hours) and $N_{exploit} = 960$ AIs (8 hours). The larger exploitation time (8 hours vs 2 hours in the first experiment) is needed because of the higher number of actions to be tested (15 actions vs 5 actions in the first experiment). EQ-MRO uses a 8x15 Q-table (states·actions), whereas EN-MRO replaces the Q-table by a shallow ANN with two hidden layers of 4 neurons each. The ANN is trained for the first time after $N_{explore}$ AIs and re-trained every 40 AIs (i.e, every 20 minutes), since the system needs more time to collect brand new data to learn from. To speed up the learning process, one action, the one with the lowest Q-value, is dropped from the learning process every 40 AIs (20 minutes). This process is repeated 10 times.

In a third experiment, the complete system with all services (i.e., s = FTP, VIDEO and WEB) is considered.The addition of new services leads to 24 potential states and 15 possible actions per state, shown in Tables B.8 and B.9. Thus, longer simulation times are needed. Specifically, $N_{explore} = 300$ AIs (150 minutes), $N_{train} = 960$ AIs (8 hours) and $N_{exploit} = 1320$ AIs (11 hours). EQ-MRO uses a 24x15 Q-table to store the Q-value per state-action pair, whereas EN-MRO replaces the Q-table by the same shallow ANN described in the second experiment. The ANN is trained for the first time after 300 AIs and re-trained every 75 AIs (37.5 minutes) while $\varepsilon > 0.75$, or every 50 AIs (25 minutes) otherwise. To speed up the learning process, one action, the one with the lowest Q-value, is dropped from the learning process every 75 AIs (37.5 minutes) if $\varepsilon > 0.75$, or every 50 AIs (25

Table B.10: Q-MRO performance (Experiment 1).

	Initial	**End**
$\overline{EHO}$ [%]	1	0.1
$\overline{LHO}$ [%]	18.52	2
$\overline{PP}$ [%]	21.6	19
$\overline{SHO}$ [%]	58.88	78.9
$\overline{QoE_{edge}}$ [MOS]	1.48	1.28

minutes) otherwise.

Experiment 1: Q-MRO limitations Table B.10 presents initial and final values for the different metrics in the first experiment. As shown in the table, LHOs are practically removed at the end of the optimization process ($\overline{LHO}$ = 2 %, compared to $\overline{LHO}$ = 18 % at the beginning). In contrast, EHOs were already low at the beginning of the process (1 %), due to the way the HO scheme is designed in the simulation tool. Nonetheless, EHOs are also reduced at the end of the process (0.1 %). Regarding PP, higher figures at the beginning are not significantly reduced at the end of the process (21.6 and 19 %, respectively). This is due to the minor weight of the PP metric in the reward function (B.53). Ultimately, SHO increase as a result of the improvement (i.e., decrease) in the other HO events.

Finally, Figure B.14 shows the evolution of $\overline{QoE_{edge}}$ during the optimization process. Note that, in this experiment, no QoE metric is included in the reward function. Specifically, the initial and final values of $\overline{QoE_{edge}}$ are 1.48 and 1.28, respectively. Thus, the HO performance improvement is obtained at the expense of deteriorating the QoE of cell edge users by 0.2 MOS points, in line with the reward function in (B.53), which does not take QoE into account.

Experiment 2: QoE-aware algorithms (single service) Table B.11 shows the initial and final metrics in the second experiment. All methods share the same initial value and the best final value is highlighted for each metric. Q-MRO manages to reduce LHO (from 18.7 to 4.05 %), but QoE is unaltered ($\overline{QoE_{edge}}$ = 1.45). In contrast, EN-MRO halves PP (from 20.65 to 11.44 %), while LHO is increased (from 18.7 to 21.35 %), resulting in $\overline{SHO}$ = 66.68 %. EQ-MRO achieves the best SHO by improving all indicators (from 18.7 to 9.85 % for LHO and 20.65 to 17.84 % in PP). In this second experiment, cell edge QoE is slightly improved by both E-MRO schemes (from 1.45 up to 1.48 with EQ-MRO and up to 1.52 with EN-MRO).

As for $\overline{QoE_{edge}}$, $\overline{QoE}$ also improves with EQ-MRO and EN-MRO compared to Q-MRO (4.17 for

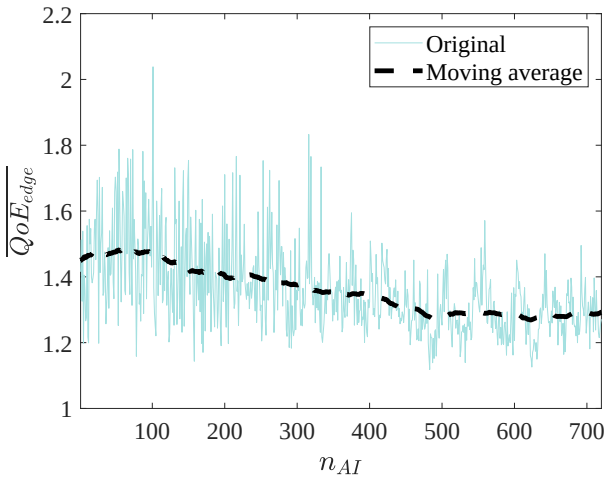


Figure B.14: $\overline{QoE_{edge}}$ over time.

Table B.11: Method performance (Experiment 2).

	Initial	**Q-MRO**	**EQ-MRO**	**EN-MRO**
$\overline{EHO}$ [%]	0.46	1.12	0.47	0.53
$\overline{LHO}$ [%]	18.7	4.05	9.85	21.35
$\overline{PP}$ [%]	20.65	25.14	17.84	11.44
$\overline{SHO}$ [%]	60.19	69.7	71.84	66.68
$\overline{T}$ [Mbps]	-	4	4.19	4.29
$\overline{QoE}$ [MOS]	-	4.09	4.17	4.17
$\overline{QoE_{edge}}$ [MOS]	1.45	1.45	1.48	1.52

both EQ-MRO and EN-MRO against 4.09 for Q-MRO). Likewise, $\overline{T}$ improves (4.19 for EQ-MRO and 4.29 Mpbs for EN-MRO, compared to 4 Mbps for Q-MRO). These results show that E-MRO algorithms not only improve cell edge users, as it is reflected in the reward function, but also the global average QoE and throughput figures compared to Q-MRO.

Table B.12 details the best action, a, selected per state, x. As expected, Q-MRO selects those actions with the HOM settings that trigger the HO when the signal level received from the serving cell is the same as that from the target cell (i.e., action $a = 3$, with $HOM = 0$ dB and $TTT = 40$ ms) for 6 out of the 8 states in this second experiment (states $x = 3$ to 8). Only in states $x = 1$ and 2 (corresponding to adjacencies with small ISD and unloaded target cell), HO is delayed by selecting higher HOM values (actions 4 and 5, with $HOM = 1$ and 2 dB, respectively). In contrast, EQ-MRO labels as best actions those delaying the HO trigger (i.e, action 4 with $HOM = 1$ dB and $TTT = 40$ ms, action 5 with $HOM = 2$ dB and $TTT = 40$ ms and action 15 with $HOM = 2$ dB

Table B.12: Best actions selected per state (Experiment 2).

	State index x							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Q-MRO	4	5	3	3	3	3	3	3
EQ-MRO	15	5	5	8	5	5	4	3
EN-MRO	14	10	9	4	15	13	4	3

Table B.13: Method performance (Experiment 3).

	Initial	**Q-MRO**	**EQ-MRO**	**EN-MRO**
$\overline{EHO}$ [%]	1.27	0.64	0.43	0.43
$\overline{LHO}$ [%]	19.92	5.08	12.78	12.8
$\overline{PP}$ [%]	25.87	27.22	13.28	14.75
$\overline{SHO}$ [%]	52.94	67.06	73.51	72.02
$\overline{T}$ [Mbps]	-	3.59	3.64	3.69
$\overline{QoE}$ [MOS]	-	4.08	4.12	4.12
$\overline{QoE_{edge}}$ [MOS]	2.04	2.07	2.09	2.14
$\overline{QoE_{edge}^{(VIDEO)}}$ [MOS]	2.05	2.11	2.1	2.11
$\overline{QoE_{edge}^{(FTP)}}$ [MOS]	1.58	1.55	1.59	1.66
$\overline{QoE_{edge}^{(WEB)}}$ [MOS]	2.38	2.43	2.46	2.56
$\overline{QoE_{diff}}$ [MOS]	0.6	0.4	0.75	0.75

and $TTT = 256$ ms) in 6 out of the 8 states. Finally, EN-MRO delays HO triggering even more, since the best actions in 5 out of the 8 states show $TTT \geq 100$ ms.

A more detailed analysis compares the best actions selected by EQ-MRO and EN-MRO for states with small ISD $\{1, 2, 5, 6\}$ and large ISD $\{3, 4, 7, 8\}$. The selected actions in the former group, comprising adjacencies between distant cells, delay the HO point (by increasing HOM, TTT or both) by a larger amount than the second group, comprising adjacencies between nearby cells (which choose lower HOM and/or TTT values). Such a behavior stresses the importance of considering ISD (i.e., parameter d) when selecting optimal HO settings. A similar analysis (not presented here) shows that EN-MRO tends to suggest actions with larger TTT values for the lower user speed of 30 km/h (states $1, 3, 5, 7$).

Experiment 3: QoE-aware algorithms (multiple services) Table B.13 shows the main performance indicators at the end of the optimization process for all methods. For a more detailed analysis, QoE figures are broken down by service.

Similarly to Experiment 2, Q-MRO achieves the best performance in terms of LHO ($\overline{LHO} =$

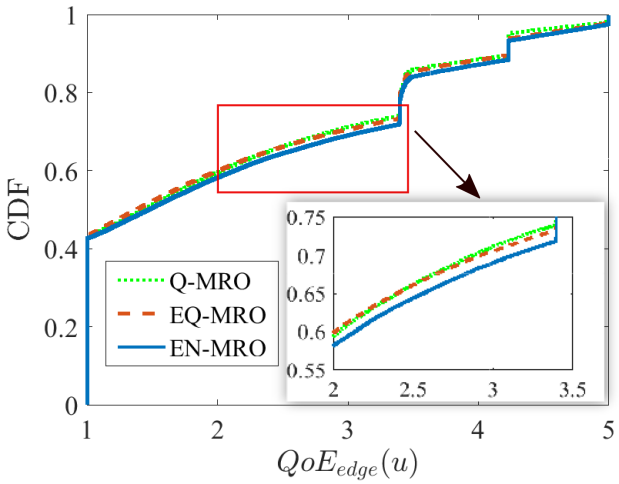


Figure B.15: Cumulative density function of optimized user QoE.

5.08 %) compared to other algorithms. However, EQ-MRO and EN-MRO end up with a better SHO ratio (67.06 % for Q-MRO vs 73.51 % for EQ-MRO and 72.02 % for EN-MRO) and better cell edge QoE. In particular, EN-MRO obtains the best QoE indicators, both aggregated and per service (i.e., $\overline{QoE_{edge}}$, $\overline{QoE_{edge}^{(VIDEO)}}$, ...). Likewise, the overall QoE figure for Q-MRO ($\overline{QoE} = 4.08$) is outperformed by EQ-MRO and EN-MRO in a similar amount ($\overline{QoE} = 4.12$). Finally, $\overline{T}$ is also improved by EQ-MRO and EN-MRO ($\overline{T} = 3.59$ Mbps for Q-MRO, while $\overline{T} = 3.64$ Mbps for EQ-MRO and 3.69 Mbps for EN-MRO).

For a more detailed analysis of cell edge performance, Figure B.15 shows the cumulative density function for individual users at the end of the optimization process. To this end, users are ordered from worst to best QoE (i.e., from 1 to 5). The curves of Q-MRO and EQ-MRO are above the EN-MRO curve. Thus, both E-MRO methods not only improve cell edge QoE, but also users with medium/best QoE values. Specifically, when comparing the $70^{th}-$percentile of MOS, Q-MRO obtains 2.87 against 2.93 and 3.13 points for EQ-MRO and EN-MRO, respectively.

Table B.14 breaks down the best actions per state and algorithm at the end of the optimization process. Recall that states 1-8 refer to FTP users, states 9-16 refer to video users and states 17-24 correspond to web users, as shown in Table (B.8). Thus, states $x = 1$, 9 and 17 show the same network state except for the service (FTP, VIDEO and WEB, respectively). A detailed analysis (not presented here) shows that the best HOM and TTT settings per state are similar to those in Experiment 2, with low positive values for both parameters.

Table B.14: Best actions per state (Experiment 3).

	State index x							
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Q-MRO	3	3	3	2	4	3	3	3
EQ-MRO	5	10	4	3	9	10	4	4
EN-MRO	5	4	4	3	10	11	4	4
	State index x							
	9	**10**	**11**	**12**	**13**	**14**	**15**	**16**
Q-MRO	3	5	3	3	3	4	3	3
EQ-MRO	5	5	3	3	5	12	3	3
EN-MRO	5	5	4	4	4	9	4	3
	State index x							
	17	**18**	**19**	**20**	**21**	**22**	**23**	**24**
Q-MRO	3	4	3	4	3	3	3	3
EQ-MRO	5	5	5	4	10	5	5	5
EN-MRO	5	4	4	4	10	5	5	5

B.3.5. Implementation issues

All the algorithms proposed in this Ph.D. thesis are implemented in Matlab R2016b. In particular, EN-MRO is implemented with the Deep Learning Toolbox.

The proposed EQ/EN-MRO algorithms are executed periodically (every 30 s of network time) until the best actions have been discovered. In terms of execution time, the most limiting factor is the large period to collect enough HO events to train the ANN in EN-MRO. This time grows linearly with the number of system states. In this work, simulations are carried out in a personal computer with a 3.6-GHz octa-core processor and 24 GB of RAM. With the above-mentioned toolbox, training an ANN with 2 hidden layers of 4 neurons takes more time than updating a bidimensional Q-table matrix, even if the former is shallow (0.07 seconds per training operation). Specifically, the average execution time of one execution of EQ-MRO (Q-table) and EN-MRO (ANN) is 0.12 and 0.19 seconds, respectively. Overall, EQ-MRO and EN-MRO take 2.6 and 4.2 minutes when 11 hours of network time are simulated (1320 executions, 1 per AI).

B.3.6. Conclusions

In this Section, a novel QoE-aware mobility robustness optimization scheme for adjusting handover trigger points in a LTE network has been proposed. The aim of the algorithm is to improve QoE at cell edge while increasing the ratio of successful HOs. The proposed learning algorithm

changes handover trigger points periodically (every 30 seconds) based on a Q-learning scheme. Two variants have been presented, depending on the way the expected Q-value per state-action pair is obtained: either with a Q-table or by training an artificial neural network implemented with a multi-layer perceptron. Method assessment has been carried out in a dynamic system-level LTE simulator implementing a realistic macrocellular scenario.

Results have shown that a legacy MRO optimization algorithm only driven by radio performance degrades QoE up to 0.2 MOS points in a simplified scenario with a single service and medium user speed. In contrast, the two variants of the proposed QoE-driven algorithm improves cell edge QoE, while also increasing the ratio of successful handovers. Compared to the legacy method, both variants improve the successful handover ratio by more than 5 % in absolute terms, while cell edge QoE of some services is increased by up to 0.13 MOS points. Web services experience larger improvements than file sharing and video streaming. Moreover, from the analysis of the parameter settings suggested by the algorithm, it has been deduced that the handover trigger point should be delayed more in adjacencies between distant cells and slow moving users.

B.4. Final conclusions and future work

In this Ph.D. thesis, it has been carried out a deep search, analysis and experimentation work for solving QoE problems in LTE networks. The first two contributions of this thesis have been presented in section B.2. The problem formulation brings out the limitations of the classical load balancing algorithms in mobile networks, which do not necessarily equalize or optimize QoE, which is crucial in current operator policies. The aim of a first preliminary experiment is to uncover the limitations of these traditional techniques when the focus is on QoE. To overcome the limitations of legacy load balancing schemes in LTE, several traffic steering algorithms have been conceived for macrocellular LTE networks. The proposed methods alleviate QoE imbalance and degradation problems by re-distributing traffic demand among neighbor cells. With the aim of balancing average cell QoE or maximize the global system QoE, the proposed methods modify cell service areas by adjusting handover margins per cell, or per cell and service, so that part of the traffic originated in cells with the worst cell-average or average user QoE is steered to adjacent cells with better QoE, either cell or user. For simplicity, the first average cell QoE balancing method has been developed using fuzzy logic controllers, which take advantage of expert knowledge by defining the control problem with simple <<IF-THEN>> rules. In this first approach, the strategy EB-CS, adjusting handover margins per cell and service, has proven to achieve the best results when the aim is to minimize average cell QoE imbalance with different service mixes. The second approach relies on analytical performance model to estimate the impact of handover settings on system QoE. With

that model, a classical gradient ascent algorithm based on measurements can be used to ensure that every change in handover margins increases the global system QoE.

The third contribution of this Ph.D. thesis is the adaptive algorithm for QoE-driven mobility robustness optimization presented in section B.3. To this end, a classical Q-learning framework has been extended to consider not only handover performance, but also QoE measurements. Thus, the main novelty of the proposed handover parameter is the evaluation of user QoE in a time window comprising 1 second before and 1 second after the HO takes place. The aim of this temporal window is to isolate the effect of handover from other possible effects in user QoE. Two variants of the algorithm have been proposed, where the rewards per state-action are computed by the Bellman equation or by an artificial neural network.

All the proposed self-tuning algorithms have been validated in a macrocellular LTE system-level simulator specifically modified for this thesis. Whenever possible, the scenario has been designed to reflect a real environment by collecting configuration parameters and traffic statistics from the live network. The flexibility provided by the simulator has been used to show the robustness of the proposed techniques for different (and sometimes extreme) traffic conditions.

The optimization schemes for QoE-driven traffic steering proposed in section B.2 make use of information available in the OSS. Thus, they are conceived as centralized solutions. Due to their low computational complexity, they can be integrated in automatic network planning and optimization tools offered by manufacturers for the operations support system. In contrast, the adaptive schemes for mobility robustness optimization proposed in section B.3 are conceived to be implemented in the base station, provided that instantaneous QoE statistics must be available.

B.4.1. Future work

The time frame of any thesis is limited, which leaves room for future works. Some of them are improvements on the proposed algorithms, whereas some others are extensions of the work to other problems in upcoming cellular networks.

- *Development of a proactive load balancing algorithm.* Currently, one of the main problems of SON techniques is the lack of proactivity. To be proactive, a QoE forecasting model is needed to predict user satisfaction in the medium and long term. As an additional feature, QoE predictions could be integrated in the QoE balancing algorithm. These predictions would allow to react to traffic changes through planning and/or optimization actions that

take QoE into account. For long term prediction, classical time series analysis techniques are usually used. Examples of these techniques are *Holt-Winters, Auto Regressive Integrative Moving Average* (ARIMA) or *AutoRegressive Conditional Heteroscedasticity* (GARCH) [132]. Modern schemes use machine learning algorithms (e.g., decision trees [150], support vector machines [151], neural networks [152]). It is still to be checked if these approaches can be used to predict QoE trends with different time horizons (minutes, hours, days or months). Likewise, QoE estimation in network management relies on encrypted traffic classification, which is still an open problem [153].

- *Extension to new services in 5G systems.* Services used in this work may be labeled as eMBB (Enhanced Mobile Broadband). In future 5G systems, new services will be offered based on Massive Machine Type Communications (mMTC) and Ultra-Reliable and Low Latency Communications (URLLC). mMTC services need a very low bandwidth, since short messages are sent. However, URLLC services do not require they present high delay and reliability requirements. These differences imply important changes in the definition, design and development of the algorithms proposed in this thesis. An interesting point is how to define QoE models for services that do not involve a human being. Likewise, the evaluation of self-tuning schemes for URLLC services is challenging, since the time resolution needed for these services increases the computational load of simulations, limiting the number of optimization loops that can be executed.

- *Real-time self-tuning algorithms.* The algorithms proposed in this thesis are conceived for persistent QoE problems, setting a fixed periodicity for the optimization process of 1 hour. However, in practice, if the information needed is available for shorter time periods (e.g., minutes), this periodicity can be reduced to solve instantaneous QoE problems. Thus, parameter tuning is carried out continuously, so as to quickly adapt to any change in network conditions. The main drawback would be the lack of reliability in the parameter tuning process, given that the algorithms would take decisions based on a small number of samples. In particular, the reliability of QoE statistics would have to be guaranteed by establishing confidence intervals for the indicators or even not to make use of the optimization process when the number of samples is not enough.

- *Improvement of the learning process through deep learning.* The third proposed mobility robustness optimization scheme is based on Q-learning. Its main goal is to increase the successful handover rate as well as optimizing the average cell edge user QoE. With this aim, discrete states and actions have been maintained. A feasible extension of this work is to make a more precise and sophisticated optimization algorithm using the machine learning technique Deep-Q-Networks (DQN). DQN is also based on Q-learning and neural networks, but it allows

to have a continuous state and action spaces, being capable of learning the best action per state without exploring all of them.

- *Analysis of other 5G system scenarios.* The proposed traffic steering and MRO schemes can be applied to scenarios combining macrocells with small cells. Small cells have reduced service areas, which might require to configure a lower hysteresis value to limit the maximum handover margin deviation, or establish a maximum value for TTT between smalls cells or between a small cell and a macrocell. In addition, including small cells in the considered scenario significantly increases the number of adjacencies in the network, which inevitably leads to a higher computational complexity. Likewise, the use of the millimeter band might require different handover settings for MRO, but smaller cell overlapping may limit the traffic steering capability.

B.4.2. Publications

The list of publications resulted from this Ph.D. thesis are listed below.

Papers

I M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, and C. Gijón, A QoE-Driven Traffic Steering Algorithm for LTE Networks, IEEE Transactions on Vehicular Technology, vol. 68, no. 11, pp. 11271-11282, Nov. 2019.

II M. L. Marí-Altozano, M. Toril, S. Luna-Ramírez and C. Gijón, "A Self-Tuning Algorithm for Optimal QoE-Driven Traffic Steering in LTE," IEEE Access, vol. 8, pp. 156707-156717, Sep. 2020.

III M. L. Marí-Altozano, S. Mwanje, S. Luna-Ramírez, M. Toril, H. Sanneck and C. Gijón, "A service-centric Q-learning algorithm for mobility robustness optimization in LTE," IEEE Transactions on Network and Service Management, Apr. 2021.

Conference papers

IV M.L. Marí-Altozano, S. Luna-Ramírez, M. Toril, "Limitaciones del equilibrio de carga para la mejora de la calidad de experiencia en redes LTE", XXXII Simposio de la Unión Científica Internacional de Radio (URSI) 2017, Cartagena(España), Sep. 2017.

v M.L. Marí-Altozano, S. Luna-Ramírez, M. Toril, C. Gijón, "Algoritmo de control de carga basado en calidad de experiencia en redes LTE", XXXIII Simposio de la Unión Científica Internacional de Radio (URSI) 2018, Granada(España), Sep. 2018.

vi M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, "Load balance performance analysis with a quality of experience perspective in LTE networks", CA15104 IRACON, Graz(Austria), Sep. 2018.

vii M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, "A QoE-driven Traffic Steering algorithm for LTE networks", CA15104 IRACON, Dublin(Ireland), Jan. 2019.

viii M.L. Marí-Altozano, S. Luna-Ramírez, M. Toril, C. Gijón, "Optimización de la calidad de experiencia en redes LTE mediante el reparto de tráfico", XXXIV Simposio de la Unión Científica Internacional de Radio (URSI) 2019, Sevilla(España), Sep. 2019.

ix M.L. Marí-Altozano, S. Mwanje, S. Luna-Ramírez, M. Toril, C. Gijón, "Una visión basada en QoE para algoritmo MRO en redes LTE", XXXV Simposio de la Unión Científica Internacional de Radio (URSI), Málaga(España), Sep. 2020.

Traffic steering algorithms presented in section B.2, are described in [i, iv, v, vi, vii]. References [iv] and [vi] present the analysis of limitations of classical load balancing algorithms from the QoE perspective. References [i, v and vii], present QoE balancing algorithms for macrocellular LTE networks. In [ii, viii], the analytical algorithm for QoE optimization is presented. Finally, references [iii, ix] describe MRO algorithm for cell edge QoE and HO success rate optimization, as described in Section B.3.

Bibliografía

[1] Ericsson AB, "Ericsson mobility report," Nov. 2017.

[2] O. G. Aliu, A. Imran, M. A. Imran, and B. G. Evans, "A Survey of Self Organisation in Future Cellular Networks," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 1, pp. 336–361, 2013.

[3] G. Gómez, J. Lorca, R. García, and Q. Pérez, "Towards a QoE-Driven Resource Control in LTE and LTE-A Networks," *Journal of Computer Networks and Communications*, 2013.

[4] J. Ramiro and K. Hamied, *Self-Organizing Networks (SON): Self-Planning, Self-Optimization and Self-Healing for GSM, UMTS and LTE.* Wiley, UK, 2011.

[5] J. Kojima and K. Mizoe, "Radio mobile communication system wherein probability of loss of calls is reduced without a surplus of base station equipment, U.S. Patent 4435840," vol. 54, no. 5, pp. 1875–1886, Sep. 1984.

[6] A. J. Fehske, H. Klessig, J. Voigt, and G. P. Fettweis, "Concurrent Load-Aware Adjustment of User Association and Antenna Tilts in Self-Organizing Radio Networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 1974–1988, Jun. 2013.

[7] V. Bratu and C. Beckman, "Base station antenna tilt for load balancing," Jan. 2013, pp. 2039–2043.

[8] Y. Khan, B. Sayrac, and E. Moulines, "Centralized self-optimization in LTE-A using Active Antenna Systems," in *2013 IFIP Wireless Days (WD)*, 2013, pp. 1–3.

[9] N. Papaoulakis, D. Nikitopoulos, and S. Kyriazakosin, "Practical radio resource management techniques for increased mobile network performance," *12th IST Mobile and Wireless Communications Summit*, Jun. 2003.

[10] V. Wille, S. Pedraza, M. Toril, R. Ferrer, and J. Escobar, "Trial Results from Adaptive Hand-Over Boundary Modification," *IEE Electronics Letters*, vol. 39, pp. 405–407, Feb. 2003.

[11] A. Lobinger, S. Stefanski, T. Jansen, and I. Balan, "Load Balancing in Downlink LTE Self-Optimizing Networks," in *2010 IEEE 71st Vehicular Technology Conference*, May. 2010, pp. 1–5.

[12] R. Kwan, R. Arnott, R. Paterson, R. Trivisonno, and M. Kubota, "On Mobility Load Balancing for LTE Systems," in *2010 IEEE 72nd Vehicular Technology Conference - Fall*, Sep. 2010, pp. 1–5.

[13] S. Luna-Ramírez, M. Toril, M. Fernández-Navarro, and V. Wille, "Optimal Traffic Sharing in GERAN," *Wireless Personal Communications*, vol. 57, no. 4, pp. 553–574, Apr. 2011.

[14] M. Toril and V. Wille, "Optimisation of Handover Parameters for Traffic Sharing in GERAN," *Wireless Personal Communications*, vol. 74, no. 3, pp. 315–336, Feb. 2008.

[15] A. Lobinger, S. Stefanski, T. Jansen, and I. Balan, "Load Balancing in Downlink LTE Self-Optimizing Networks," in *2010 IEEE 71st Vehicular Technology Conference*, May. 2010, pp. 1–5.

[16] P. Muñoz, R. Barco, and I. de la Bandera, "Optimization of load balancing using fuzzy Q-learning for next generation wireless networks," *Expert Systems with Applications*, vol. 40, no. 4, pp. 984 – 994, 2013.

[17] L. Gimenez, I. Z. Kovács, J. Wigard, and K. I. Pedersen, "Throughput-Based Traffic Steering in LTE-Advanced HetNet Deployments," Sep. 2015.

[18] J. M. Ruiz-Avilés, S. Luna-Ramírez, M. Toril, and F. Ruiz, "Traffic steering by self-tuning controllers in enterprise LTE femtocells," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, no. 1, p. 337, Nov. 2012.

[19] M. M. Hasan, S. Kwon, and J. Na, "Adaptive Mobility Load Balancing Algorithm for LTE Small-Cell Networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 4, pp. 2205–2217, 2018.

[20] B. Abuhaija, "3GPP Technologies: Load Balancing Algorithm and InterNetworking," in *2014 4th International Conference on Artificial Intelligence with Applications in Engineering and Technology*, 2014, pp. 267–271.

[21] J. M. Ruiz-Avilés, M. Toril, S. Luna-Ramírez, V. Buenestado, and M. A. Regueira, "Analysis of Limitations of Mobility Load Balancing in a Live LTE System," *IEEE Wireless Communications Letters*, vol. 4, no. 4, pp. 417–420, Aug. 2015.

[22] 3GPP Technical Report 36.902 v9.0.0, "Self-configuring and self-optimizing network use cases and solutions (Release 9)," Oct. 2009.

[23] I. M. Bălan, B. Sas, T. Jansen, I. Moerman, K. Spaey, and P. Demeester, "An enhanced weighted performance-based handover parameter optimization algorithm for LTE networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, no. 1, p. 98, Sep 2011.

[24] T. Jansen, I. Balan, J. Turk, I. Moerman, and T. Kürner, "Handover Parameter Optimization in LTE Self-Organizing Networks," Sep. 2010, pp. 1–5.

[25] M. M. Hasan, S. Kwon, and J. Na, "Adaptive mobility load balancing algorithm for LTE Small-Cell Networks," *IEEE Transactions on Wireless Communications*, vol. 17, no. 4, pp. 2205–2217, 2018.

[26] D. Lopez-Perez, I. Guvenc, and X. Chu, "Mobility management challenges in 3gpp heterogeneous networks," *IEEE Communications Magazine*, vol. 50, no. 12, pp. 70–78, 2012.

[27] D. López-Pérez, I. Guvenc, and X. Chu, "Theoretical analysis of handover failure and ping-pong rates for heterogeneous networks," in *2012 IEEE International Conference on Communications (ICC)*, 2012, pp. 6774–6779.

[28] S. S. Mwanje and A. Mitschele-Thiel, "Distributed cooperative Q-learning for mobility-sensitive handover optimization in LTE SON," in *2014 IEEE Symposium on Computers and Communications (ISCC)*, vol. Workshops, Jun. 2014, pp. 1–6.

[29] J. Wu, J. Liu, Z. Huang, and S. Zheng, "Dynamic fuzzy Q-learning for handover parameters optimization in 5G multi-tier networks," in *2015 International Conference on Wireless Communications Signal Processing (WCSP)*, Oct 2015, pp. 1–5.

[30] P. Sapkale and U. Kolekar, *Handover Decision Algorithm for Next Generation*, Jan. 2020, pp. 269–277.

[31] I. Shayea, M. Ismail, R. Nordin, and H. Mohamad, "Adaptive Handover Decision Algorithm Based on Multi-Influence Factors through Carrier Aggregation Implementation in LTE-Advanced System," *Journal of Computer Networks and Communications*, vol. 2014, Nov. 2014.

[32] M. Peuster, H. Küttner, and H. Karl, "A flow handover protocol to support state migration in softwarized networks," *International Journal of Network Management*, vol. 29, Apr. 2019.

[33] P. Singh, M. Kumar, and A. Das, "A design approach to maximize handover performance success rate and enhancement of voice quality samples for a GSM cellular network," in *2014 International Conference on Signal Propagation and Computer Technology (ICSPCT 2014)*, 2014, pp. 294–299.

[34] M. Schinnenburg, I. Forkel, and B. Haverkamp, "Realization and optimization of soft and softer handover in UMTS networks," in *2003 5th European Personal Mobile Communications Conference (Conf. Publ. No. 492)*, 2003, pp. 603–607.

[35] Y. Wang, J. Chang, and G. Huang, "A Handover Prediction Mechanism Based on LTE-A UE History Information," in *2015 18th International Conference on Network-Based Information Systems*, 2015, pp. 167–172.

[36] H. Holma, A. Toskala, and J. Reunanen, *Small Cell Optimization*, 2015, pp. 121–144.

[37] P. Oliver-Balsalobre, M. Toril, S. Luna-Ramírez, and J. M. Ruiz Avilés, "Self-tuning of scheduling parameters for balancing the quality of experience among services in LTE," *EURASIP Journal on Wireless Communications and Networking*, vol. 2016, no. 1, p. 7, Jan. 2016.

[38] P. Oliver-Balsalobre, M. Toril, S. Luna-Ramírez, and R. G. Garaluz, "Self-Tuning of Service Priority Parameters for Optimizing Quality of Experience in LTE," *IEEE Transactions on Vehicular Technology*, vol. 67, no. 4, pp. 3534–3544, Apr. 2018.

[39] E. Hossain and M. Hasan, "5g cellular: Key enabling technologies and research challenges," *Instrumentation & Measurement Magazine, IEEE*, vol. 18, Feb. 2015.

[40] NGMN, "5G White paper," White paper, Feb. 2015.

[41] Scenarios, requirements and KPIs for 5G mobile and wireless system ICT-317669 METIS project, D6.1 v1-2, 2013.

[42] 3GPP Technical Report 25.892 v6.0.0, "Feasibility Study for Orthogonal Frequency Division Multiplexing(OFDM) for UTRAN Enhancement(Rel-6), pp. 62-63," 2004.

[43] P. Seeling and M. Reisslein, "Video Transport Evaluation With H.264 Video Traces," *IEEE Communications Surveys Tutorials*, vol. 14, no. 4, pp. 1142–1165, Apr. 2012.

[44] R. R. Tyagi, F. Aurzada, K. Lee, and M. Reisslein, "Connection Establishment in LTE-A Networks: Justification of Poisson Process Modeling," *IEEE Systems Journal*, vol. 11, no. 4, pp. 2383–2394, Dec. 2017.

[45] T. L. Casavant and J. G. Kuhl, "A taxonomy of scheduling in general-purpose distributed computing systems," *IEEE Transactions on Software Engineering*, vol. 14, no. 2, pp. 141–154, Feb. 1988.

[46] O. ØsterbØ and O. GrØndalen, "Benefits of Self-Organizing Networks (SON) for Mobile Operators," *Journal of Computer Networks and Communications*, Sep. 2012.

[47] J. L. Bejarano-Luque, M. Toril, M. Fernández-Navarro, A. J. García, and S. Luna-Ramírez, "A Context-Aware Data-Driven Algorithm for Small Cell Site Selection in Cellular Networks," *IEEE Access*, vol. 8, pp. 105 335–105 350, 2020.

[48] A. Kurrne, "Mobile Measurement based frequency planning in GSM network, m.s. thesis, helsinki university of technology, dept. of applied mathematics," 2001.

[49] H. Hafez, F. E. El-Taher, R. A. Ramadan, and A. Gaber, "Scrambling code planning and optimization for umts system," in *2014 IEEE Wireless Communications and Networking Conference (WCNC)*, 2014, pp. 1968–1973.

[50] R. Acedo-Hernárndez, M. Toril, S. Luna-Ramírez, and C. Ubeda, "A PCI planning algorithm for jointly reducing reference signal collisions in LTE uplink and downlink," *Computer Networks*, vol. 119, pp. 112–123, Jun. 2017.

[51] M. Toril, I. Molina-Fernández, V. Wille, and C. Walshaw, "Analysis of Heuristic Graph Partitioning Methods for the Assignment of Packet Control Units in GERAN," *Wireless Personal Communications*, vol. 60, pp. 611–633, Oct. 2011.

[52] M. Toril, P. Guerrero-García, S. Luna-Ramírez, and V. Wille, "An efficient integer programming formulation for the assignment of base stations to controllers in cellular networks," *Computer Networks*, vol. 56, pp. 303–314, Jan. 2012.

[53] M. Toril, S. Luna-Ramírez, and V. Wille, "Automatic replanning of tracking areas in cellular networks," *IEEE Transactions on Vehicular Technology*, vol. 62, no. 5, pp. 2005–2013, 2013.

[54] M. Toril, R. Ferrer, S. Pedraza, V. Wille, and J. Escobar, "Optimization of Half-Rate codec assignment in GERAN," *Wireless Personal Communications*, vol. 34, pp. 321–331, Aug. 2005.

[55] P. Frohlich, W. Nejdl, K. Jobmann, and H. Wietgrefe, "Model-based alarm correlation in cellular phone networks," in *Proceedings Fifth International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, 1997, pp. 197–204.

[56] A. GarcÃa, M. Toril, P. Oliver Balsalobre, S. Luna-Ramírez, and M. Ortiz, "Automatic alarm prioritization by data mining for fault management in cellular networks," *Expert Systems with Applications*, vol. 158, p. 113526, May. 2020.

[57] S. Chernov, M. Cochez, and T. Ristaniemi, "Anomaly Detection Algorithms for the Sleeping Cell Detection in LTE Networks," in *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*, 2015, pp. 1–5.

[58] R. Barco, V. Wille, L. Diez, and M. Toril, "Learning of model parameters for fault diagnosis in wireless networks," *Wireless Networks*, vol. 16, pp. 255–271, Aug. 2010.

[59] M. Amirijoo, L. Jorguseski, R. Litjens, and L. C. Schmelz, "Cell Outage Compensation in LTE Networks: Algorithms and Performance Assessment," in *2011 IEEE 73rd Vehicular Technology Conference (VTC Spring)*, 2011, pp. 1–5.

[60] V. Buenestado, M. Toril, S. Luna-Ramírez, J. M. Ruiz-Avilés, and A. Mendo, "Self-tuning of Remote Electrical Tilts Based on Call Traces for Coverage and Capacity Optimization in LTE," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 5, pp. 4315–4326, 2017.

[61] A. Durán, M. Toril, F. Ruiz, and A. Mendo, "Self-Optimization Algorithm for Outer Loop Link Adaptation in LTE," *IEEE Communications Letters*, vol. 19, no. 11, pp. 2005–2008, 2015.

[62] A. Vallejo-Mora, M. Toril, S. Luna-Ramírez, M. Regueira, and S. Pedraza, "Analytical Model for Estimating the Impact of Changing the Nominal Power Parameter in LTE," *Mobile Information Systems*, vol. 2018, pp. 1–7, Aug. 2018.

[63] Z. Zhang, K. Long, J. Wang, and F. Dressler, "On Swarm Intelligence Inspired Self-Organized Networking: Its Bionic Mechanisms, Designing Principles and Optimization Approaches," *IEEE Communications Surveys Tutorials*, vol. 16, no. 1, pp. 513–537, 2014.

[64] Z. Zhang, W. Huangfu, K. Long, X. Zhang, X. Liu, and B. Zhong, "On the designing principles and optimization approaches of bio-inspired self-organized network: A survey," *Science China Information Sciences*, vol. 56, 07 2013.

[65] G. Feng, "A Survey on Analysis and Design of Model-Based Fuzzy Control Systems," *IEEE Transactions on Fuzzy Systems*, vol. 14, no. 5, pp. 676–697, 2006.

[66] T. Ross, *Fuzzy logic with engineering applications.* McGraw-Hill, 1995.

[67] I. de la Bandera, S. Luna-Ramírez, R. Barco, M. Toril, F. Ruiz, and M. Fernández-Navarro, "Inter-system cell reselection parameter auto-tuning in a joint-RRM scenario," in *2010 Fifth International Conference on Broadband and Biomedical Communications*, 2010, pp. 1–6.

[68] M. Saeed, H. Kamal, and M. El-Ghoneimy, "A new fuzzy logic technique for handover parameters optimization in LTE," in *2016 28th International Conference on Microelectronics (ICM)*, 2016, pp. 53–56.

[69] P. Zhang, "Neural Networks for Classification: A Survey," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 30, pp. 451 – 462, 12 2000.

[70] G. I. Papadimitriou, "A new approach to the design of reinforcement schemes for learning automata: stochastic estimator learning algorithms," *IEEE Transactions on Knowledge and Data Engineering*, vol. 6, no. 4, pp. 649–654, 1994.

[71] Z. Zhang, K. Long, J. Wang, and F. Dressler, "On Swarm Intelligence Inspired Self-Organized Networking: Its Bionic Mechanisms, Designing Principles and Optimization Approaches," *IEEE Communications Surveys Tutorials*, vol. 16, no. 1, pp. 513–537, 2014.

[72] R. Narasimhan and D. C. Cox, "A handoff algorithm for wireless systems using pattern recognition," in *Ninth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (Cat. No.98TH8361)*, vol. 1, Sep. 1998, pp. 335–339 vol.1.

[73] M. Ekpenyong, J. Isabona, and E. Isong, "Handoffs Decision Optimization of Mobile Celular Networks," in *2015 International Conference on Computational Science and Computational Intelligence (CSCI)*, Dec 2015, pp. 697–702.

[74] F. Shaoshuai, H. Tian, and C. Sengul, "Self-optimization of coverage and capacity based on a fuzzy neural network with cooperative reinforcement learning," *EURASIP Journal on Wireless Communications and Networking*, vol. 2014, p. 57, 04 2014.

[75] R. Chai, J. Cheng, X. Pu, and Q. Chen, "Neural Network Based Vertical Handoff Performance Enhancement in Heterogeneous Wireless Networks," in *2011 7th International Conference on Wireless Communications, Networking and Mobile Computing*, 2011, pp. 1–4.

[76] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1, no. 2.

[77] P. V. Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A Survey of Machine Learning Techniques Applied to Self-Organizing Cellular Networks," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2392–2431, 2017.

[78] A. Imran, A. Zoha, and A. Abu-Dayya, "Challenges in 5G: how to empower SON with big data for enabling 5G," *IEEE Network*, vol. 28, no. 6, pp. 27–33, 2014.

[79] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. The MIT Press, 2018.

[80] R. Amiri, M. A. Almasi, J. G. Andrews, and H. Mehrpouyan, "Reinforcement Learning for Self Organization and Power Control of Two-Tier Heterogeneous Networks," *IEEE Transactions on Wireless Communications*, vol. 18, no. 8, pp. 3933–3947, 2019.

[81] Wencong Qin, Yinglei Teng, Mei Song, Yinghai Zhang, and Xiaojun Wang, "A Q-learning approach for mobility robustness optimization in Lte-son," in *2013 15th IEEE International Conference on Communication Technology*, Nov 2013, pp. 818–822.

[82] Y. Xu, W. Xu, Z. Wang, J. Lin, and S. Cui, "Deep Reinforcement Learning Based Mobility Load Balancing Under Multiple Behavior Policies," in *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)*, May 2019, pp. 1–6.

[83] Y. Xu, W. Xu, Z. Wang, J. Lin, and S. Cui, "Load Balancing for Ultradense Networks: A Deep Reinforcement Learning-Based Approach," *IEEE Internet of Things Journal*, vol. 6, no. 6, pp. 9399–9412, 2019.

[84] K. Brunnström, K. Moor, A. Dooms, S. Egger-Lampl, M.-N. Garcia, T. Hossfeld, S. Jumisko-Pyykkö, C. Keimel, C. Larabi, B. Lawlor, P. Le Callet, S. Möller, F. Pereira, M. Pereira, A. Perkis, A. Pinheiro, U. Reiter, P. Reichl, R. Schatz, and A. Zgank, *Qualinet White Paper on Definitions of Quality of Experience*, Mar. 2013.

[85] A. Díaz, P. Merino, and F. Javier Rivas, "Customer-centric measurements on mobile phones," in *2008 IEEE International Symposium on Consumer Electronics*, 2008, pp. 1–4.

[86] J. K. Pathak, S. Krishnamurthy, and R. Govindarajan, "System, method, and apparatus for measuring application performance management," Patent number: WO03007115, 2003.

[87] S. Barakovic and L. Skorin-Kapov, "Survey and Challenges of QoE Management Issues in Wireless Networks," *Journal of Computer Networks and Communications*, vol. 2013, Mar. 2013.

[88] ITU-T G.107 Recommendation, "The E-model: A Computational Model for Use in Transmission Planning," Dec. 2011.

[89] R. K. P. Mok, E. W. W. Chan, and R. K. C. Chang, "Measuring the quality of experience of HTTP video streaming," in *12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops*, 2011, pp. 485–492.

[90] A. Wattimena, R. Kooij, J. Vugt, and O. Ahmed, "Predicting the perceived quality of a first person shooter: the Quake IV G-model," in *5th ACMSIGCOMM NetGames 06*, Oct. 2006, p. 42.

[91] H. Batteram, G. Damm, A. Mukhopadhyay, L. Philippart, R. Odysseos, and C. Urrutia Valdés, "Delivering quality of experience in multimedia networks," *Bell Labs Technical Journal*, vol. 15, no. 1, pp. 175–193, 2010.

[92] 3GPP Technical Specification 26.234 v10.1.0, "Transparent end-to-end packet-switched streaming service (PSS); Protocols and codecs," Jun. 2011.

[93] 3GPP Technical Specification 26.114 v11.0.0, "IP multimedia subsystem (IMS); Multimedia telephony; Media handling and interaction," Jun. 2011.

[94] ITU-T G.107 Recommendation, "Opinion model for network planning of video and audio streaming applications," Nov. 2016.

[95] P. Reichl, B. Tuffin, and R. Schatz, "Logarithmic laws in service quality perception: where microeconomics meets psychophysics and quality of experience," *Telecommunication Systems*, vol. 52, no. 2, pp. 587–600, Feb. 2013.

[96] ITU-T G.114 Recommendation, "One-Way Transmission Time," 2003.

[97] J. Navarro-Ortiz, J. M. Lopez-Soler, and G. Stea, "Quality of experience based resource sharing in IEEE 802.11e HCCA," in *2010 European Wireless Conference (EW)*, Apr. 2010, pp. 454–461.

[98] R. Baloch and I. Awan, "A mathematical model for wireless channel allocation and handoff schemes," *Telecommunication Systems*, vol. 45, pp. 275–287, 12 2010.

[99] C. Kim, S. A. Dudin, O. S. Dudina, and A. N. Dudin, "Mathematical Models for the Operation of a Cell With Bandwidth Sharing and Moving Users," *IEEE Transactions on Wireless Communications*, vol. 19, no. 2, pp. 744–755, 2020.

[100] X. Zhao, X. Luo, T. Wu, and D. Xiao, "The Prediction Mathematical Model for HO Performance in LTE Networks," in *2013 IEEE Eighth International Conference on Networking, Architecture and Storage*, 2013, pp. 191–197.

[101] P. Muñoz, I. de la Bandera Cascales, F. Ruiz, S. Luna-Ramírez, R. Barco, M. Toril, P. Lázaro, and J. Rodríguez, "Computationally-Efficient Design of a Dynamic System-Level LTE Simulator," *International Journal of Electronics and Telecommunications*, vol. 57, pp. 347–358, Nov. 2011.

[102] P. Bergman, J. Moe, and F. Gunnarsson, "Self-optimizing handover oscillation mitigation - Algorithms and field evaluations," in *2012 International Symposium on Wireless Communication Systems (ISWCS)*, 2012, pp. 31–35.

[103] K. I. Pedersen, T. E. Kolding, F. Frederiksen, I. Z. Kovacs, D. Laselva, and P. E. Mogensen, "An overview of downlink radio resource management for UTRAN long-term evolution," *IEEE Communications Magazine*, vol. 47, no. 7, pp. 86–93, Jul. 2009.

[104] F. R. M. Lima, T. F. Maciel, W. C. Freitas, and F. R. P. Cavalcanti, "Resource Assignment for Rate Maximization With QoS Guarantees in Multiservice Wireless Systems," *IEEE Transactions on Vehicular Technology*, vol. 61, no. 3, pp. 1318–1332, Mar. 2012.

[105] J. Rhee, J. M. Holtzman, and D.-K. Kim, "Scheduling of Real/Non-real Time Services: Adaptive EXP/PF Algorithm," in *The 57th IEEE Semiannual Vehicular Technology Conference, VTC 2003-Spring*, vol. 1, Apr. 2003, pp. 462–466 vol.1.

[106] P. Ameigeiras, J. J. Ramos-Munoz, J. Navarro-Ortiz, P. Mogensen, and J. M. Lopez-Soler, "QoE oriented cross-layer design of a resource allocation algorithm in beyond 3G systems," *Computer Communications*, vol. 33, no. 5, pp. 571 – 582, 2010.

[107] F. Wamser, D. Staehle, J. Prokopec, A. Maeder, and P. Tran-Gia, "Utilizing Buffered Youtube Playtime for QoE-Oriented Scheduling in OFDMA Networks," in *24th International Teletraffic Congress (ITC), no. 15*, 2012.

[108] J. Chen, R. Mahindra, M. Khojastepour, S. Rangarajan, and M. Chiang, "A scheduling framework for adaptive video delivery over cellular networks," Sep. 2013, pp. 389–400.

[109] V. Joseph and G. de Veciana, "NOVA: QoE-driven optimization of DASH-based video delivery in networks," in *IEEE INFOCOM 2014 - IEEE Conference on Computer Communications*, April 2014, pp. 82–90.

[110] Z. Yan, J. Xue, and C. W. Chen, "Prius: Hybrid Edge Cloud and Client Adaptation for HTTP Adaptive Streaming in Cellular Networks," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 27, no. 1, pp. 209–222, Jan. 2017.

[111] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller." *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 20, no. 2, pp. 404–418, Mar. 1990.

[112] "G-MON," https://play.google.com/store/apps/details?id=de.carknue.gmon2&hl=en_US&gl=US.

[113] J. L. Bejarano-Luque, M. Toril, M. Fernández-Navarro, R. Acedo-Hernández, and S. Luna-Ramírez, "A Data-Driven Algorithm for Indoor/Outdoor Detection Based on Connection Traces in a LTE Network," *IEEE Access*, vol. 7, pp. 65 877–65 888, 2019.

[114] C. Chandra, T. Jeanes, and W. H. Leung, "Determination of optimal handover boundaries in a cellular network based on traffic distribution analysis of mobile measurement reports," in *1997 IEEE 47th Vehicular Technology Conference. Technology in Motion*, vol. 1, 1997, pp. 305–309.

[115] M. L. Marí-Altozano, S. Luna-Ramírez, M. Toril, and C. Gijón, "A QoE-Driven Traffic Steering Algorithm for LTE Networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 11 271–11 282, Nov 2019.

[116] A. Bär, P. Casas, A. D'Alconzo, P. Fiadino, L. Golab, M. Mellia, and E. Schikuta, "Dbstream: a Holistic Approach to Large-scale Network Traffic Monitoring and Analysis," *Computer Networks, vol. 107, part 1*, pp. 5–19, 2016.

[117] S. Hämäläinen, H. Sanneck, and C. Sartori, "LTE Self-Organizing Networks (SON): Network Management Automation for Operational Efficiency Hardcover," 2012.

[118] N. O. Tuncel and M. Koca, "Joint ICIC and Mobility Management Optimization in Self-Organizing Networks," in *2017 IEEE Wireless Communications and Networking Conference (WCNC)*, March 2017, pp. 1–6.

[119] R. Vijayan and J. Holtzman, "Model for analyzing handoff algorithms," *Vehicular Technology, IEEE Transactions on*, vol. 42, pp. 351 – 356, 09 1993.

[120] O. Andrisano, M. Dell'Acqua, G. Mazzini, R. Verdone, and A. Zanella, "On the parameters optimization in handover algorithms," in *VTC '98. 48th IEEE Vehicular Technology Conference. Pathway to Global Wireless Revolution (Cat. No.98CH36151)*, vol. 2, 1998, pp. 1400–1404 vol.2.

[121] O. Andrisano, M. Dell'Acqua, G. Mazzini, and R. Verdone, "On the parameters optimization in handover algorithms," vol. 2, 06 1998, pp. 1400 – 1404 vol.2.

[122] G. Hui and P. Legg, "Soft Metric Assisted Mobility Robustness Optimization in LTE Networks," in *2012 International Symposium on Wireless Communication Systems (ISWCS)*, 2012, pp. 1–5.

[123] V. Buenestado, J. M. Ruiz-Aviles, M. Toril, and S. Luna-Ramirez, "Mobility Robustness Optimization in Enterprise LTE Femtocells," in *2013 IEEE 77th Vehicular Technology Conference (VTC Spring)*, 2013, pp. 1–5.

[124] Y. Mal, J. Chen, and H. Lin, "Mobility robustness optimization based on radio link failure prediction," in *2018 Tenth International Conference on Ubiquitous and Future Networks (ICUFN)*, July 2018, pp. 454–457.

[125] 3GPP Technical Specification 36.331, "E-UTRA radio resource control (RRC) protocol specification (Release 8)," Dec. 2011.

[126] X. Zhang, *Mobility Optimization*, 2017, pp. 254–359.

[127] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement Learning: A Survey," *Journal of Artificial Intelligence Research*, vol. 4, no. 1, pp. 237–285, May 1996.

[128] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[129] L. Ghignone and M. Barlow, "Shallow Network Training With Dynamic Sample Weights Decay - a Potential Function Approximator for Reinforcement Learning," in *2019 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2019, pp. 149–154.

[130] D. Bertsekas and J. Tsitsiklis, *Neuro-Dynamic Programming*, 01 1996, vol. 27.

[131] E. Even-Dar and Y. Mansour, "Learning rates for q-learning," *J. Mach. Learn. Res*, vol. 5, pp. 1–25, Jan. 2004.

[132] G. E. P. Box and G. Jenkins, *Time Series Analysis, Forecasting and Control.* USA: Holden-Day, Inc., 1990.

[133] C. Gijón, M. Toril, S. Luna-Ramírez, J. L. Bejarano-Luque, and M. L. Marí-Altozano, "Estimating pole capacity from radio network performance statistics by supervised learning," *IEEE Transactions on Network and Service Management*, pp. 1–1, 2020.

[134] P. E. Mogensen and J. Wigard, "COST Action 231: Digital Mobile Radio Towards Future Generation System, Final Report," 1999.

[135] F. Khan, *LTE for 4G Mobile Broadband: Air Interface Technologies and Performance*, 1st ed. USA: Cambridge University Press, 2009.

[136] J. Parsons, *The Mobile Radio Propagation Channel.* Pentech Press, Jan. 1992.

[137] 3GPP Technical Specification 36.101, "User Equipment (UE) Radio Transmission and Reception (Release 9)," Dec. 2009.

[138] B. Habib, G. Zaharia, and G. E. Zein, "Digital block design of mimo hardware simulator for lte applications," in *2012 IEEE International Conference on Communications (ICC)*, 2012, pp. 4489–4493.

[139] J. Colom Ikuno, M. Wrulich, and M. Rupp, "Performance and modeling of LTE H-ARQ," *International ITG Workshop on Smart Antennas (WSA 2009)*, 01 2009.

[140] 3GPP R1-040500, "OFDM-HSDPA System level simulator calibration."

[141] S. Pratschner, B. Tahir, L. Marijanovic, M. Mussbah, K. Kirev, R. Nissel, S. Schwarz, and M. Rupp, "Versatile mobile communications simulation: the Vienna 5G Link Level Simulator," *EURASIP Journal on Wireless Communications and Networking*, vol. 2018, Sep. 2018.

[142] 3GPP Technical Specification 36.521, "E-UTRA; UE conformance specification; Radio transmission and reception; Part 1: Conformance testing," 2009.

[143] C. Mehlführer, M. Wrulich, J. C. Ikuno, D. Bosanska, and M. Rupp, "Simulating the Long Term Evolution physical layer," in *2009 17th European Signal Processing Conference*, 2009, pp. 1471–1478.

[144] J. T. Entrambasaguas, M. C. Aguayo-Torres, G. Gomez, and J. F. Paris, "Multiuser Capacity and Fairness Evaluation of Channel/QoS-Aware Multiplexing Algorithms," *IEEE Network*, vol. 21, no. 3, pp. 24–30, 2007.

[145] A. Imran, A. Zoha, and A. Abu-Dayya, "Challenges in 5G: how to empower SON with big data for enabling 5G," *IEEE Network*, vol. 28, no. 6, pp. 27–33, 2014.

[146] J. M. Ruiz-Avilés, M. Toril, and S. Luna-Ramírez, "A femtocell location strategy for improving adaptive traffic sharing in heterogeneous LTE networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2015, pp. 1–13, 2015.

[147] K. Zheng, Z. Yang, K. Zhang, P. Chatzimisios, K. Yang, and W. Xiang, "Big data-driven optimization for mobile networks toward 5G," *IEEE Network*, vol. 30, no. 1, pp. 44–51, 2016.

[148] A. J. Garcia, M. Toril, P. Oliver, S. Luna-Ramirez, and R. Garcia, "Big Data Analytics for Automated QoE Management in Mobile Networks," *IEEE Communications Magazine*, vol. 57, no. 8, pp. 91–97, 2019.

[149] M. L. Marí-Altozano, M. Toril, S. Luna-Ramírez, and C. Gijón, "A Self-Tuning Algorithm for Optimal QoE-Driven Traffic Steering in LTE," *IEEE Access*, vol. 8, pp. 156 707–156 717, 2020.

[150] F. Cai, Y. Gao, L. Cheng, L. Sang, and D. Yang, "Spectrum sharing for LTE and WiFi coexistence using decision tree and game theory," in *2016 IEEE Wireless Communications and Networking Conference*, 2016, pp. 1–6.

[151] M. Cordina and C. J. Debono, "A support vector machine based sub-band CQI feedback compression scheme for 3GPP LTE systems," in *2017 International Symposium on Wireless Communication Systems (ISWCS)*, 2017, pp. 325–330.

[152] D. Li, D. Li, and Y. Xu, "Machine Learning Based Handover Performance Improvement for LTE-R," in *2019 IEEE International Conference on Consumer Electronics - Taiwan (ICCE-TW)*, 2019, pp. 1–2.

[153] C. Gijón, M. Toril, M. Solera, S. Luna-Ramírez, and L. R. Jiménez, "Encrypted Traffic Classification Based on Unsupervised Learning in Cellular Radio Access Networks," *IEEE Access*, vol. 8, pp. 167252–167263, 2020.