A QoE-driven traffic steering algorithm for LTE networks

María Luisa Marí-Altozano, Salvador Luna-Ramírez, Matías Toril, Carolina Gijón
Email: {mlma, sluna, mtoril, cgm}@ic.uma.es
Department of Communication Engineering, University of Málaga, 29071, Málaga, Spain.

***Abstract*—Due to the diversity of mobile services and raising user expectations, mobile network management has changed its focus from Quality of Service (QoS) to Quality of Experience (QoE). As a consequence, classical network optimization procedures must be updated accordingly. One of these procedures is traffic sharing, whose aim is to redistribute traffic among adjacent cells so as to provide an adequate QoE to subscribers. In this work, 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, 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. Method assessment is based on a dynamic system-level simulator implementing a realistic LTE scenario. Results show that the proposed QoE-driven traffic sharing algorithm alleviates QoE problems by equalizing user QoE throughout the scenario.**

***Index Terms*—Long Term Evolution (LTE), self organizing network (SON), self-tuning, fuzzy, quality of experience.**

I. Introduction

Over the last few years, there has been an exponential growth in the demand of mobile services. At the same time, the success of smartphones and tablets has changed traffic patterns in mobile networks due to the introduction of new services [1]. These changes will continue in the coming years with the deployment of 5G systems, which will introduce new mobile use cases [2].

In parallel, technological advances have raised users' expectations, forcing operators to change the way they manage their networks. Traditionally, network management has been based on objective performance indicators (*Quality of Service*, QoS) measuring user or network performance (e.g., accessibility, retainability, integrity...). Recently, operators have shifted their focus from network performance to end user opinion (a.k.a. *Quality of Experience*, QoE). In this context, QoE is defined as the overall satisfaction of a service as subjectively perceived by the user [3]. Customer experience management (CEM) will be even more important in 5G, as services with very different requirements will coexist (e.g., high-definition television, virtual/augmented reality, autonomous vehicle, sensor networks ...) [4]. Thus, maximizing the QoE should be the main criterion for assigning radio resources [5] or dynamically selecting paths in software-defined architectures in future 5G systems [6].

The above paradigm change will make network management more complicated. The need for an efficient network management has caused intense research on automation techniques, referred to as *Self Organizing Networks* (SON). SON procedures are classified into three use cases: self-planning, self-healing and self-optimization [7]. In particular, self-optimization includes those techniques designed to cope with network changes so that optimal network performance is always ensured during the operational stage. Traffic steering (a.k.a. traffic sharing or load balancing) is one of the key use cases of self-optimization [8]. The aim of traffic sharing is to alleviate congestion problems due to the uneven traffic demand by redistributing users among neighbor cells. This is achieved by changing cell service areas with new base station parameter settings, such as, e.g., transmit power [9], cell reselection offset [10], antenna tilt angle [11] or HandOver (HO) margin [12], the latter being the preferred option (referred to as mobility load balancing, MLB). Likewise, load balancing algorithms can be classified into static or dynamic approaches [13]. Static approaches can make use of analytical approaches to ensure optimal performance proactively in the long term [14] [15]. In contrast, dynamic approaches rely on simple reactive schemes, prone to instabilities. First dynamic MLB algorithms designed for Long Term Evolution (LTE) were based on a simple proportional controller driven by the load imbalance between adjacent cells [16] [17]. As shown in [15], these algorithms may lead to severe network performance degradation due to the tight frequency reuse in LTE. More sophisticated algorithms use fuzzy logic controllers with reinforcement learning [18] or combine MLB with remote electrical tilting [11] or power re-planning [19].

All the above-mentioned MLB algorithms are driven by simple indicators, such as average cell load or call blocking ratio. Thus, QoE is not taken into account. In current mobile networks, QoS control is carried out by packet scheduling (PS) algorithms, dynamically assigning radio resources to user data requests based on QoS constraints [20] [21]. More sophisticated schedulers exploit multiuser diversity gain to achieve optimal system performance and ensure user fairness [22]. Several QoE-aware schedulers have been proposed in the literature to optimize the overall QoE, while ensuring a minimum QoE for all users. Such advanced schedulers are designed for specific services (e.g., web [23], progressive video streaming [24] or adaptive video streaming [25], [26],

[27]). However, the aim of most schedulers is to ensure a minimum QoS/QoE for the worst users or equalize the average QoS/QoE per service within a cell, rather than equalizing the average QoE of services across cells in the network. Thus, QoE balance between users or services in the spatial domain is not guaranteed. A QoE imbalanced network implies an unfair allocation of radio resources: while fully satisfied users in underutilized cells waste resources without increasing their QoE, there are completely unsatisfied users in congested cells who lack these resources. In the end, such an unfairly distributed users' satisfaction may turn into higher churn rates and revenue loss. Moreover, implementing the aforementioned advanced QoE-aware schedulers would require upgrading network equipment, which is not desired by network operators that have already made an important investment to upgrade to the latest radio access technology. Alternatively, a standard scheduler can be tuned to improve the system QoE. In [28], a self-tuning algorithm for a classical multi-service packet scheduler is proposed to balance QoE across services by re-prioritizing users in a LTE cell. Similarly, in [29], a different self-tuning algorithm for the same scheduler is designed based on optimality criteria to ensure the best overall system QoE driven by network performance statistics. However, none of these self-tuning schemes manages to equalize the QoE across cells in the system. Even if MLB may potentially reduce QoE differences between cells, to the best of authors' knowledge, no MLB algorithm in the literature explicitly takes QoE into account.

In this work, a novel QoE-aware MLB algorithm is proposed for LTE systems. Unlike previous approaches, the proposed algorithm aims to minimize QoE differences across cells and services by adjusting handover parameters on a per-adjacency basis. Parameter tuning is performed by a fuzzy logic controller driven by QoE estimates obtained from key service performance indicators. The algorithm is validated in a dynamic system-level simulator implementing a realistic macrocellular LTE scenario. The main contributions of this work are: a) uncovering the limitations of traditional traffic sharing schemes from a QoE perspective, b) a novel self-tuning algorithm for balancing QoE by modifying handover margins, and c) the validation of the algorithm via simulations in a realistic macrocellular LTE scenario. The rest of the work is organized as follows. Section II discusses the limitations of classical load balancing schemes in terms of QoE. Section III describes the system model used in this work. Section IV describes the proposed QoE balancing algorithm. Section V presents algorithm assessment. Finally, Section VI summarizes the main conclusions.

II. Problem formulation

In mobile networks, the HO process ensures a seamless connection between neighbor cells when the user moves. Specifically, a HO is triggered when the following condition is fulfilled

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

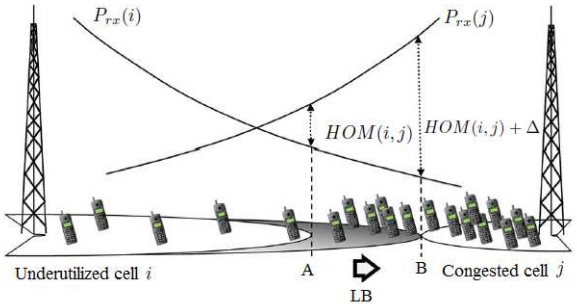


Fig. 1: Traffic sharing by changing handover margins [15].

where $P_{rx}(j)$ is the pilot signal level received from neighbor cell j, $P_{rx}(i)$ is the pilot signal level received from the serving cell i, and HOM is the HO margin, defined on a per-adjacency basis (i.e., one value for each pair of cells and direction of the adjacency). In most cases, HO margins are set complementarily in both directions of the adjacency to prevent ping-pong effect, so that

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

where H represents the hysteresis value.

Figure 1 illustrates how modifying HO margins can be used as a traffic sharing technique [15]. An increase of Δ dB in $HOM(i,j)$ enlarges the serving area of cell i while decreasing that of cell j. Thus, the carried traffic (and load) in cell i increases, whereas the carried traffic (and load) of cell j decreases. Conversely, a decrease in $HOM(i,j)$ reduces the serving area of cell i, while increasing that of cell j.

Classical Load Balance (LB) schemes modify HO margins to equalize the load between neighbor cells in the hope that the overall call blocking ratio (or some other global QoS indicator) is improved [16][18]. Such a positive effect is often achieved at the expense of deteriorating network spectral efficiency, since users are reassigned to cells that do not prove the largest signal level [15].

An evenly loaded network does not necessarily imply a QoE-balanced network. User satisfaction is highly dependent on the required service, causing that the QoE of users of different services can differ significantly even if they receive the same amount of resources. More importantly, balancing the load between adjacent cells does not necessarily reduce QoE differences among users in the cells, since some services are more sensitive to load increments than others (as will be shown later). As a consequence, equalizing the load between neighbor cells does not necessarily reduce QoE differences if the service mix (i.e., ratio of connections for each service) is not exactly the same in both cells.

The previous considerations suggest that a classical QoS-based traffic sharing scheme does reach an evenly loaded network, but might lead to a more unevenly balanced QoE distribution among cells in the system. This is the main hypothesis that will be tested in this paper.

TABLE I: 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. $\lambda_{VoIP} \simeq 0$.
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. $\lambda_{VIDEO} = 4 \cdot 10^{-3}$.
FTP	File size: log-normal distribution (avg. 20 MB) [30]. $\lambda_{FTP} = 2.5 \cdot 10^{-3}$.
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) [30]. $\lambda_{WEB} = 3.7 \cdot 10^{-3}$.

III. System model

This section outlines the traffic and QoE models of the mobile services covered in this work.

A. Traffic models

Table I shows the main characteristics of the four services considered in this work: voice over Internet Protocol (VoIP), progressive video streaming (VIDEO), file download service via File Transfer Protocol (FTP) and web browsing (WEB) [30]. VoIP is a guaranteed bit rate (GBR) service with low data rates. VoIP is modeled to generate 20 bytes of voice every 10 ms, with a bit rate of 16 kbps. In contrast, VIDEO, FTP and WEB services are non-guaranteed bit rate (non-GBR) services. The video service model (inspired in [31]) corresponds to buffered live video streaming with fixed quality (720 p) and variable bit rate. For this purpose, a simple model of the player's buffer at the client side is implemented. In live video streaming, content generation and playback request occur at the same time (unlike in video on demand, where the whole content is available at the start of the session); thus, the video server starts sending frames to the client as they are generated, which are stored in client buffer until reaching a minimum video content (3 seconds, in this work). This is modeled as a fixed video playback start delay (i.e., initial buffering time, LTI, of 3 seconds). Later, if the buffer runs out, the video stops (i.e., stalling event) and the player waits until the buffer is re-filled again. Video duration follows a uniform distribution between 0 and 540 s. Obviously, videos of less than 3 s do not experience stalling. Frame sizes are taken from a real H.264 video trace [32]. A video session drop model is also simulated, where the connection is terminated if session time is more than twice the video content duration. The other two data services FTP and WEB are best-effort services. FTP is a file download service and WEB consists of downloading several web pages with different sizes with reading time between them.

Traffic appears as data bursts; therefore, new connections follow a Poisson distribution for all services [33] [34].

B. QoE models

QoE is often measured using the *Mean Opinion Score* (MOS) scale, ranging from 1 (bad) to 5 (excellent). In absence of surveys, QoE can be estimated from QoS measurements. For this purpose, QoS measurements gathered on a session basis are mapped into QoE figures by utility functions [35]. A utility function describes the relationship between objective QoS performance indicators and subjective QoE for each service. Utility functions provide an estimate of the user QoE, although they miss contextual factors. (e.g., location, time of day, $\cdots$). Thus, network operators that do not take explicit QoE measurements can estimate user QoE by processing passive measurements of key performance indicators.

For VoIP service, user QoE can be estimated as [36]:

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

where $QoE^{(VoIP)}$ is the MOS value for a VoIP connection, and R is a parameter representing the connection quality, with values from 0 (minimum) to 93 (maximum), that only depend on the delay experienced by VoIP packets (mouth-to-ear delay). Note that $\max(QoE^{(VoIP)})$ = 4.4054 (when $R = 93$), i.e., MOS never reaches the value of 5, showing that, even with the best possible network performance, some individuals may not score their experience as excellent. Likewise, QoE is set to the minimum (i.e., $QoE^{(VoIP)}$ = 1) when the connection is dropped.

VIDEO utility function is defined as [28]:

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

where $QoE^{(VIDEO)}$ is the MOS estimated for the video connection, L_{ti} denotes the initial buffering time (in seconds), L_{fr} is the average stalling frequency (s^{-1}) (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 in VoIP, $QoE^{(Video)}$ = 1 if connection is dropped.

The utility function for FTP service is [37]

$$QoE^{(FTP)} = \max(1, \min(5, 6.5 \cdot TH - 0.54))\ , \quad (5)$$

where TH denotes the average user throughput in Mbps.

Finally, the utility function for WEB service is [37]

$$QoE^{(WEB)} = 5 - \frac{578}{1 + (\frac{TH+541.1}{45.98})^2}\ , \quad (6)$$

where TH is the average user throughput in kbps. Note that, $\max(QoE^{(WEB)})$ = 5. No dropping of web connections is considered, so that low MOS values for web are reached when TH is zero (i.e., $QoE^{(WEB)}$ = 1 when $TH \simeq 0$ kbps).

Note that the above-described QoE models do not depend on traffic model parameters (e.g., video sequence duration or file/web page size).

IV. Experience balancing algorithm

In this section, a new QoE balancing algorithm with two variants is presented.

A. Experience balancing algorithm

In this section, a new self-tuning algorithm to equalize the QoE across cells in a LTE network is presented. The proposed algorithm, referred to as Experience Balancing (EB), is inspired in the MLB algorithm described in [19] (hereafter denoted as LB). Similar to LB, EB is implemented by Fuzzy Logic Controllers (FLC) that decide whether to increase or decrease HOM on a per-adjacency basis. Compared to classical proportional-integrative-derivative (PID) controllers, fuzzy logic controllers are simpler and easier to understand, since they are described in natural language, taking advantage of operator experience. Unlike LB, EB aims to equalize average user QoE, instead of cell load, among cells. For this purpose, users are sent from a cell with a lower QoE to an adjacent cell experiencing higher QoE.

Two variants of EB are defined. In a first variant, referred to as EB-C (for cell), the indicator to be balanced is the cell average QoE, defined as

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

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

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

where $QoE^{(s)}(u)$ is the quality of experience for user u demanding service s, computed as in (3)-(6), N_s is the number of services in cell i and $N_u(i,s)$ is the number of users in cell i demanding service s. In (7), it is implicitly assumed that all services are equally important for the operator. This is aligned to the goal of ensuring that all users have the same service experience, regardless of the particular selected service.
Finally, the global average QoE is defined as

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

where N_C is the number of cells in the network.
As stated before, EB-C aims to balance the cell average QoE (7). Thus, a difference indicator is defined for EB-C as

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

Such an indicator is used as an input for EB-C FLC. Figure 2 shows the structure of the FLC. The output variable is the increment/decrement of the HOM between neighbor cells i and j, $\Delta HOM(i,j)$. As shown in the figure, FLC consists of three stages: fuzzification, inference and defuzzification.
In the fuzzification stage, the value of the input, $QoE_{diff}(i,j)$, is broadly termed with linguistic variables (e.g., very negative, negative, zero...) [38]. Such a mapping is carried out by input membership functions, μ_x, as shown in Figure 3a (x denotes the specific linguistic variable). For simplicity, triangular input membership functions are selected. Experience shows that, in mobile networks, the flexibility provided by more complex functions in the controller is not translated into a finer control due to the stochastic noise of input measurements. Also note that function overlapping causes that a crisp input value is simultaneously assigned to one or more adjectives with different degrees.

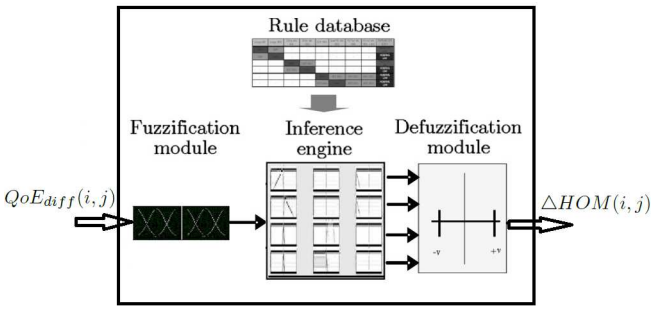


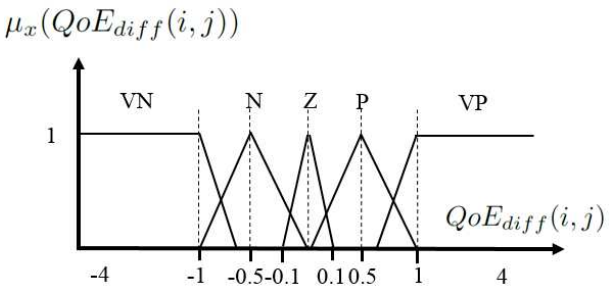
Fig. 2: Structure of fuzzy logic controller [19].

In the inference stage, a set of "IF-THEN" rules define the mapping of the input to the output in linguistic terms. A rule takes the form 'if *x* is *A*, then *y* is *B*', where *A* and *B* are adjectives associated to input and output, respectively (i.e., very negative, negative, zero, positive and very positive). The first part of the rule (*x* is *A*) is the *antecedent*, while the second part of the rule (*y* is *B*) is the *consequent*. Unlike traditional expert systems, several rules can be fired at the same time in a fuzzy inference engine. The firing strength of each rule depends on the degree in which its antecedents are satisfied (referred to as the *truth* value of the rule). This feature of fuzzy controllers ensures smooth control actions.

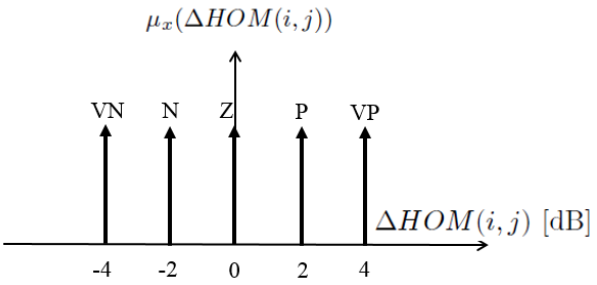
Figure 3c summarizes the set of rules that describe the tuning process in EB. Roughly, HOM is decreased (i.e., $\triangle HOM(i,j)$ is N or VN) when the average QoE in the target cell is better than in the source cell (i.e., $QoE_{diff}(i,j)$ is P or VP).

Finally, in the defuzzification stage, the output value is obtained from the aggregation of rules. In this work, the *centre-of-gravity* method [39] is applied to compute the final output value as a weighted average. Weights are calculated from the truth value of each rule, computed from the degree of fulfillment of their antecedents. For simplicity, a Takagi-Sugeno approach [38] is used, where the output membership functions are constants, as shown in Figure 3b, leading to a more compact representation easier to adjust and reducing computational load. More complex output membership functions tend to give similar results.

FLCs are periodically executed after a fixed period, referred to as reporting output period (ROP). The value of $HOM(i,j)$ for the next iteration, referred to as optimization loop, is calculated from performance measurements in the previous iteration as



(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.

Fig. 3: Fuzzy logic controller for EB-C algorithm.

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

where superscripts n and $n+1$ denotes the iteration number, and all parameters are in dB. Note that HOM changes are limited to the range -7 to 13 dB. The lower limit is the minimum signal-to-interference ratio (SINR) needed for the scheduler to assign any radio resource to a connection. The upper limit is calculated with (2) to ensure a hysteresis level of H = 6 dB. From (2), it is also deduced that any modification in $HOM(i,j)$ automatically implies an opposite change in $HOM(j,i)$. As a consequence, a single FLC is needed for both directions of the adjacency. Thus, the required number of FLC equals to the number of adjacencies in the system.

Due to their similarities, EB-C and LB would work the same (i.e., propose the same HOM changes) if the service mix was the same for all cells. However, as will be shown later, service mix differs from cell to cell in live networks, causing that EB-C and LB take different tuning actions.

Note that, even if EB-C ensures that the cell average QoE is balanced across the network (i.e., $\overline{QoE}(i) \simeq \overline{QoE}(j) \forall i,j$), some services may have larger QoE than others within a cell (i.e., $\overline{QoE}(i,s_1) \neq \overline{QoE}(i,s_2)$) or different to the same service in other cells (i.e., $\overline{QoE}(i,s_1) \neq \overline{QoE}(j,s_1)$). To solve this, a second variant of the algorithm, referred to as EB-CS, aims to balancing the average service QoE of a cell against the average user QoE of its adjacent cells. EB-CS takes advantage of the fact that some LTE vendors now give operators the flexibility to set different HOM values for services within a cell. Thus, HOM can be tuned on a per-adjacency and per-service basis driven by a new difference indicator

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

Similarly to EB-C, $QoE_{diff}(i,j,s)$ in EB-CS is defined per adjacency, but, differently to EB-C, it is segregated by service. Thus, in EB-CS, one FLC is needed per service and adjacency, each proposing a service-specific margin modification, $HOM(i,j,s)$. EB-CS can be understood as four different QoE balancing mechanisms per adjacency (i.e., one per service), which might push users in different directions, e.g., WEB users are handed over from cell i to j, but VIDEO users are handed over from j to i. Membership functions and inference rules are identical to those in EB-C, shown in Figure 3, with the only change of the input parameter, $QoE_{diff}(i,j,s)$ instead of $QoE_{diff}(i,j)$.

Another significant difference of EB-CS compared to EB-C is the loss of symmetry. The difference indicator in (12) is calculated by subtracting $\overline{QoE}(i,s)$ from $\overline{QoE}(j)$, so that $QoE_{diff}(i,j,s) \neq QoE_{diff}(j,i,s)$. Thus, unlike EB-C, two FLCs must be executed in EB-CS for the same adjacency, one per direction. This separate FLC executions in two adjacent cells i and j might lead to changes of different magnitude for both directions of the same adjacency, $\Delta HOM(i,j,s)$ and $\Delta HOM(j,i,s)$, causing that (2) is not fulfilled. To enforce a constant hysteresis level, an average ΔHOM value is calculated in EB-CS for both directions of the adjacency as

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

It can be argued that the proposed schemes are based on QoS rather than on QoE, since QoE metrics used to drive the tuning process are derived from QoS measurements. Note that the utility functions mapping QoS into QoE are not linear, and a large QoS increment does not necessarily lead to a large QoE increase. Such a non-linearity is critical when evaluating the overall system performance gain (in terms of user satisfaction) of reassigning users to a different cell. This issue is avoided by explicitly computing QoE.

V. Performance analysis

The proposed algorithm is tested in a dynamic system-level LTE simulator [40]. For clarity, the simulation set-up is presented first and results are shown later. Then, implementation issues are discussed.

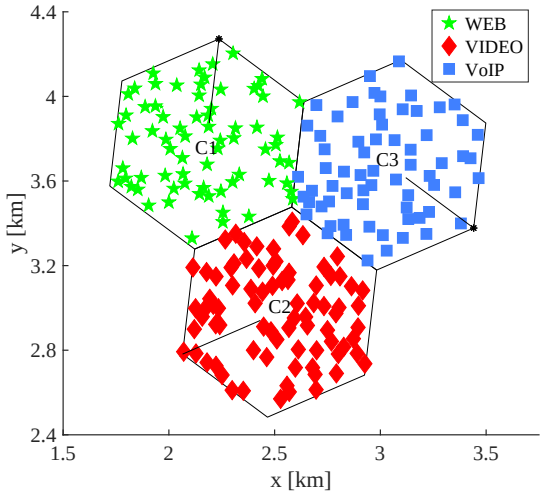


Fig. 4: Naïve scenario.

TABLE II: Simulation parameters.

Time resolution	10 TTI (10 ms)
Propagation model	Pathloss Okumura-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 [22]
Link adaptation	CQI-based
Cell geometry	cell radius = 0.5 km, Inter-site distance = 1.5 km

A. Assessment methodology

Two main experiments are performed. In a first experiment, the aim is to check how balancing load and balancing QoE behave differently in a naïve scenario. Then, in a second experiment, the proposed experience balancing schemes are compared with other self-tuning approaches in a realistic scenario to assess their performance gain. In both experiments, only the downlink is simulated to reduce the computational load.

1) First experiment (proof of concept): This experiment is a proof of concept whose aim is to reveal that load balance between two neighbor cells does not imply QoE balance. Figure 4 shows the naïve scenario used in the first experiment, consisting of a regular tri-sectorized scenario. Table II shows relevant simulation parameters.

Only for this proof of concept, traffic demand is confined to three cells (denoted as C1, C2 and C3) and it is forced that all users in a cell demand the same service (WEB, VIDEO and VoIP for C1, C2 and C3, respectively), as shown in Figure 4. User locations are represented by a different symbol depending on the requested service. Likewise, it is assumed that users are static (only in this naïve scenario) to ensure that users do not change cell for mobility reasons. Thus, it is easier to segregate users per service, which facilitates the analysis. Note that, once users are handed over to a different cell as a result of HOM tuning, different services can be found in the same cell.

In the above-described scenario, the first experiment has two stages. In a first stage, the aim is to evaluate the impact of network load on the QoE of each service. With this aim, offered traffic is swept by increasing the number of users in one of the three cells (e.g., WEB users in C1) while traffic intensity in the others is kept constant (e.g.,VIDEO and VoIP in C2 and C3, respectively). For each offered traffic value, cell load is measured as the average PRB utilization, $U(i)$, for the whole simulated period (i.e., 1 hour of network time). Likewise, offered traffic in cells with constant traffic (VIDEO in C2 and VoIP in C3 in the example) is fixed to a sufficiently large value ($U(C2) \simeq U(C3) \simeq 80\ \%$) to generate a background interference level. Users are uniformly distributed within every cell. The same test is repeated by sweeping traffic intensity in the other two cells. Note that, in this first stage, due to the specific spatial user distribution, $\overline{QoE}(i)$ coincides with the average QoE for the service demanded in the cell under study (e.g., $\overline{QoE}(C1) = \overline{QoE}(C1, WEB)$).

In a second stage, the aim of the experiment is to illustrate the limitations of classical load balancing in terms of QoE. With this aim, $HOM(i,j)$ is swept in 1 dB steps in a cell (e.g., C3). Margins from that cell to the other two cells are simultaneously swept (e.g., $HOM(C3,C1) = HOM(C3,C2)$). Hysteresis is maintained by synchronizing changes in both directions of the adjacencies to ensure (2).

2) Second experiment (algorithm assessment): The aim of the second experiment is: to check the behavior of the proposed experience balancing schemes and compare their performance against other traffic steering algorithms.

During the analysis, five self-tuning approaches are compared. The first two are the proposed schemes that aim to equalize user experience across cells, EB-C, or cells and services, EB-CS. A third scheme is the legacy MLB algorithm [19], LB, whose aim is to equalize the average PRB utilization between neighbor cells. For a fair comparison, a fourth scheme, referred to as throughput-based balancing, TB [41], is also included, to show the benefit of explicitly considering QoE instead of QoS (user throughput). TB aims to balancing the mean user throughput, $T(i)$, across cells by tuning handover margin settings on a per-adjacency basis. For this purpose, a fuzzy controller is implemented to steer users from cells with lower user throughput to cells with larger user throughput. The input to the controller is the difference of mean user throughput between adjacent cells, and the output is the handover margin of the adjacency. Finally, a fifth scheme referred to as QoE-based reprioritization, QR [28], is included to show the benefit of redistributing users between cells (as in EB-C and EB-CS) instead of reprioritizing services inside a cell by packet scheduling. QR aims to balance the QoE of users within a cell by reprioritizing services in a classical scheduler. For this purpose, a set of four proportional controllers (1 per service) are implemented per cell to tune service priority on a long-term basis so that users of services with worse QoE are prioritized. The input to each controller is the average QoE difference of a service against other services, and the output is the service priority parameter of that service in the scheduler.

A live LTE network scenario is implemented in the simulator for this second experiment. Figure 5 shows the simulated scenario, consisting of 108-macrocells (36 sites with

TABLE III: Parameters in real scenario.

Bandwidth	10 MHz (50 PRB)
Base station model	EIRP$_{max}$= 67 dBm $f_{carrier}$=1850 MHz
Traffic model	Spatial traffic distribution and service mix based on live statistics collected on a per-cell basis
Mobility model	Random direction, constant speed, 3km/h
$HOM^{(0)}(i,j)$	3 dB $\forall\,(i,j)$
$SPI^{(0)}(i,s)$	7 $\forall\,(i,s)$

3 tri-sectorized antennas per site). Table III shows the main simulation parameters, taken from the live network. In this experiment, users move at 3 km/h in a straight path randomly selected. Likewise, the default HOM and SPI settings are 3 dB and 7, respectively.

The five self-tuning algorithms (LB, TB, QR, EB-C and EB-CS) are tested along 15 optimization loops. It is checked a posteriori that the system reaches stability after 15 iterations in the five algorithms. The duration of every optimization loop (i.e., the ROP) is 1 hour, long enough to ensure reliable performance statistics. 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. For a fair comparison, it is ensured that all optimization loops for the five algorithms are executed under identical conditions by pre-generating a realization of all random variables. Thus, performance differences between loops are only due to the different HOM/SPI settings, and not to the stochastic nature of simulation. Network performance with the default HOM/SPI settings is considered as a baseline.

The aim of the proposed algorithms (EB-C and EB-CS) is to reduce differences between users of different cells and services. This is achieved by improving the worst users/services at the expense of deteriorating 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 QoE, computed as the average of all services and cells in the scenario,

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

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

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

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

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

where

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

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 (17).

Similarly, an overall throughput imbalance indicator is defined as

$$\overline{T_{imb}} = \frac{1}{N_C}\sum_i |T_{imb}(i)| = \\ = \frac{1}{N_C}\sum_i \left| \overline{T}(i) - \frac{\sum_{j\in A(i)}\overline{T}(j)}{N_{adj}(i)} \right|, \quad (18)$$

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

$$\overline{T}(i) = \frac{\sum_s \overline{T}(i,s)}{N_s(i)}\;, \quad (19)$$

where

$$\overline{T}(i,s) = \frac{\sum_{u\in(i,s)} T(u)}{N_u(i,s)}\;. \quad (20)$$

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

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

where $QoE_{imb,c}(i)$ is the average cell imbalance indicator of cell i, computed by comparing its average QoE against that

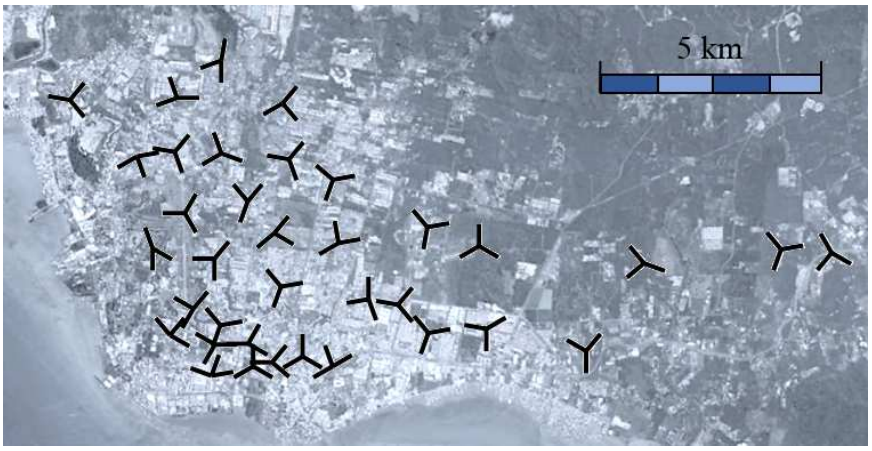


Fig. 5: Real scenario.

TABLE IV: Baseline network performance.

Indicator	Avg.	Max	Min
$N_u(i,VoIP)/N_u(i)$ [%]	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

of its neighbors. Finally, an overall QoE imbalance indicator across services in the scenario is defined as

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

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.

For clarity, Table IV shows some relevant network performance indicators with the default HOM/SPI settings. Both spatial user distribution and service mix, giving the probability of a user initiating a connection of a service in a cell, are taken from real statistics. Only the call arrival rate is artificially modified (i.e., increased) to generate a highly loaded scenario. From the table, it is deduced that VIDEO is the most popular service in the area, followed by WEB. Likewise, with the default HOM and SPI settings, cell load may differ in up to 88.1 % and cell-average QoE may differ in up to 2.53 MOS points, justifying the need for the tuning process. 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 (i.e., $HOM(i,j,VoIP) = 3$ in EB-CS) as well as QR is not allowed to change SPI settings (i.e., $SPI(i,VoIP) = 7$ in QR).

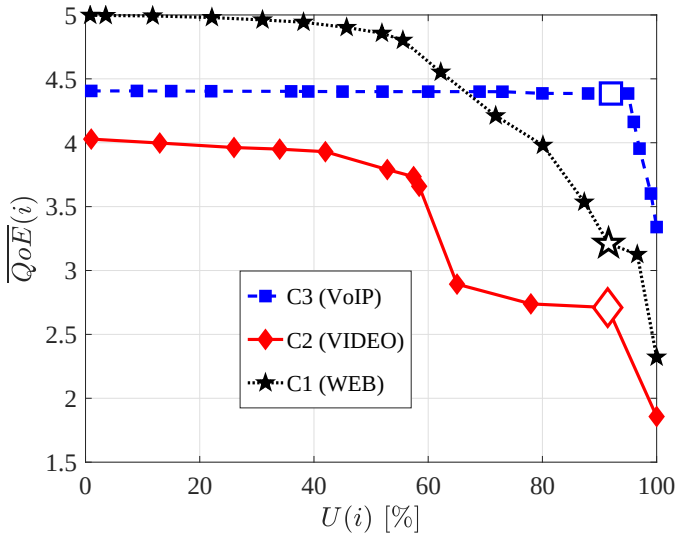


Fig. 6: QoE dependence on cell load.

B. Results

1) First experiment (proof of concept): Figure 6 shows the sensitivity of cell average QoE, $\overline{QoE}(i)$, to cell load, $U(i)$ (i.e., PRB utilization). Each curve represents one of the three cells (services) in the naïve scenario. As expected, similar load conditions do not lead to the same QoE values in the three services. Specifically, $\overline{QoE}(C3) > \overline{QoE}(C1) > \overline{QoE}(C2)$ when $U(i) > 68$ %. Thus, it is inferred that, for the scheduling algorithm in the simulator, VoIP has better experience than WEB or VIDEO for high cell load. It can also be observed that the QoE of VoIP keeps almost constant and high up to a very large cell load (i.e., $\overline{QoE}(C3) \simeq 4.4\ \forall\, U(C3) \lesssim 97$ %). The same holds for VIDEO and WEB services, but with lower load thresholds ($U(i) \approx 58$ % and 56 %, respectively).

Large markers in Figure 6, represent the working point selected for the next stage of the experiment, whose aim is to show the benefit of QoE balancing. Such settings correspond to a situation where the three cells have a similar cell load close to 90.8%, but completely different QoE values ($\overline{QoE}(C3)$ = 4.4, $\overline{QoE}(C2)$ = 3.21 and $\overline{QoE}(C1)$ = 2.71). This situation

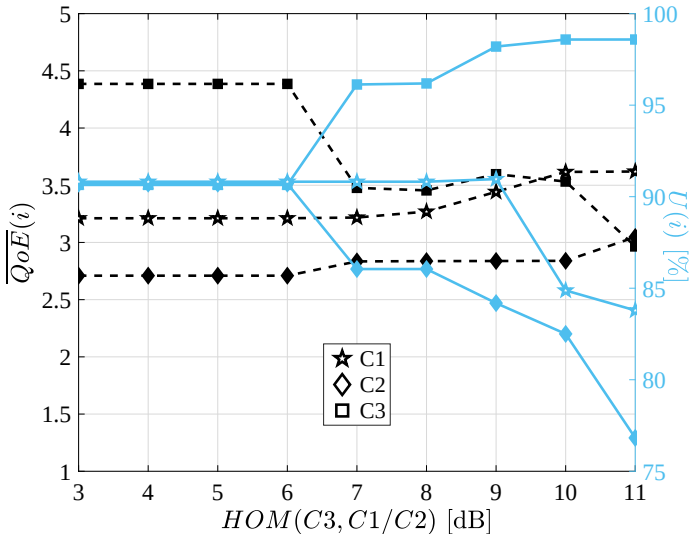


Fig. 7: Sensitivity of cell load and QoE to HOM changes.

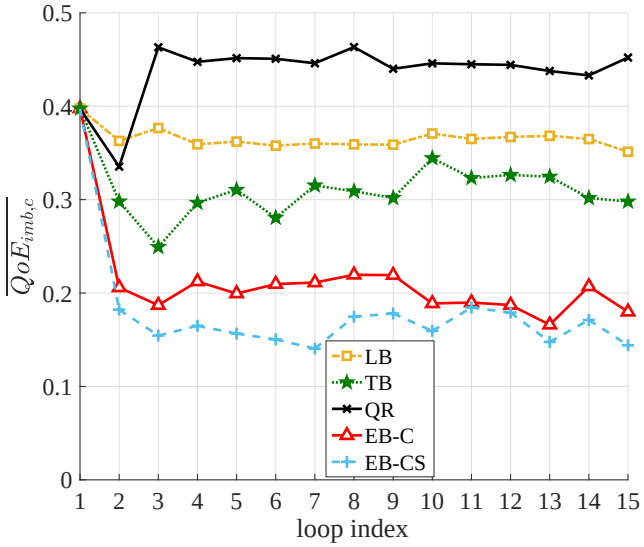


Fig. 8: Evolution of QoE imbalance.

reflects an evenly balanced scenario in terms of cell load, but an unevenly balanced scenario in terms of QoE. It is thus expected that a load balancing algorithm would not modify HOM values, even if large QoE differences exist among cells. In contrast, a QoE-driven balancing algorithm would change HOMs to equalize QoE among cells.

In the second stage of the experiment, HOMs are tuned to steer traffic from C1 and C2 (the two cells with the worst QoE) to C3 (the cell with the best QoE). Such an effect is achieved by enlarging the cell service area of C3, produced by increasing HOMs in the outgoing adjacencies of C3.

Figure 7 shows cell load (solid lines) and QoE (dashed lines) values for the three cells, when $HOM(C3, C1)$ and $HOM(C3, C2)$ are simultaneously swept in 1 dB steps from 3 (the default setting) to 11 dB. In the figure, it is observed that load imbalance increases as QoE imbalance decreases. Specifically, load imbalance increases from $\overline{U_{imb}}$= 0.3 % to 12.7 % while $\overline{QoE_{imb,c}}$ decreases from 0.96 % to 0.36 %. Thus, a load balancing algorithm would end up in $HOM(C3, C1)$=$HOM(C3, C2)$= 3 dB, while a QoE balancing algorithm would set a completely different balance point with $HOM(C3, C1/C2)$= 11 dB. This is clear evidence that load balancing and QoE balancing might drive the system to very different states in the presence of different service mixes in cells.

2) Second experiment (algorithm assessment): Figure 8 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 9 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. To clarify this capability, an average HOM offset is computed for EB-C and EB-CS as the average deviation of HOM values from the initial default settings, i.e.,

$$\overline{\delta HOM} = \frac{\sum_{(i,j,s)} \delta HOM(i,j,s)}{N_{adjs}} = \frac{\sum_{(i,j,s)} |HOM(i,j,s) - 3|}{N_{adjs}}, \quad (23)$$

where N_{adjs} is the total number of adjacencies in the network. Figure 10 illustrates the average HOM deviation evolution across iterations in all the schemes. At the 15^{th} loop, $\overline{\delta HOM}$ reaches 7 dB, 5.6 dB and 0 dB and for LB, TB and QR, respectively, showing that LB and TB produce a significant displacement of HOMs in many adjacencies, while, as expected, QR does not change HOM. EB-C and EB-CS also produce a deviation of HOMs in many adjacencies. In EB-C, $\overline{\delta HOM}$ reaches 4.6 dB at the 15^{th} optimization loop, which is less than the deviation needed by LB and TB to reach load or throughput balance. This proves again that an evenly balanced load or user throughput across the network do not necessarily imply an evenly balanced QoE. In EB-CS, $\overline{\delta HOM}$ ranges from 0 to 6 dB depending on the service. VIDEO service requires a larger HOM deviation, indicating that, with the current service mix, it is needed (on average) to hand over a larger amount of VIDEO users than FTP or WEB users to reach QoE balance among services of neighbor cells.

For comparison purposes, Table V 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 throughput 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

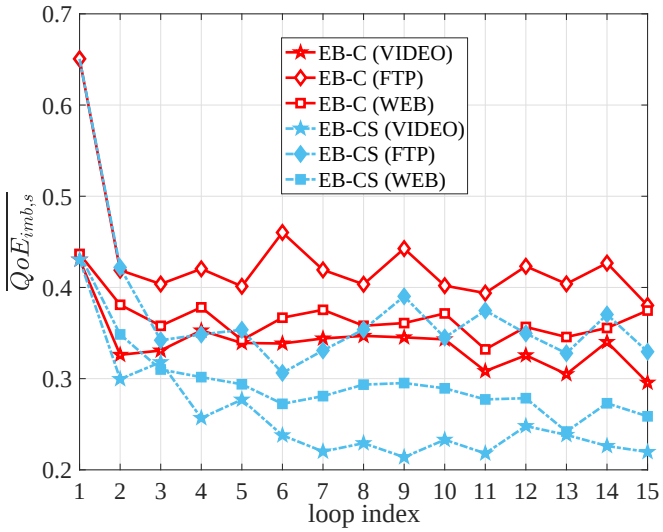


Fig. 9: Evolution of QoE imbalance per service.

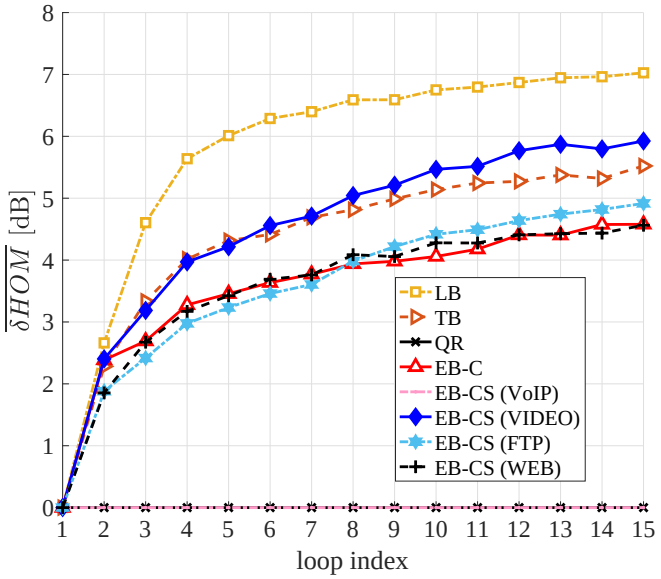


Fig. 10: Evolution of average HOM offset.

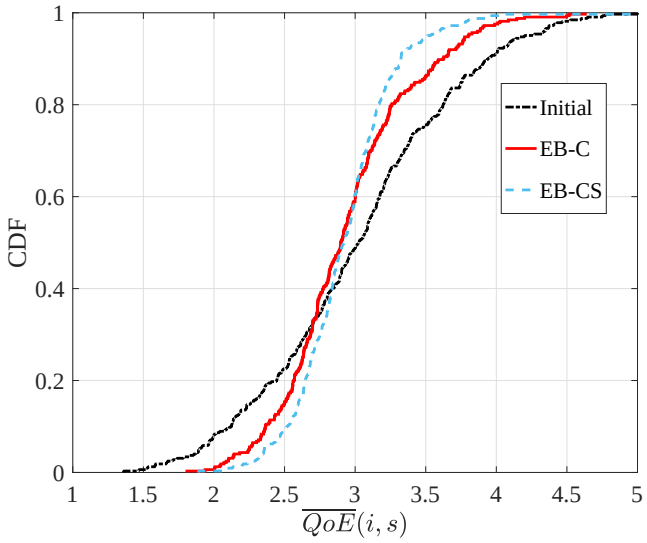


Fig. 11: QoE distribution for services across cells function.

TABLE V: Main performance indicators.

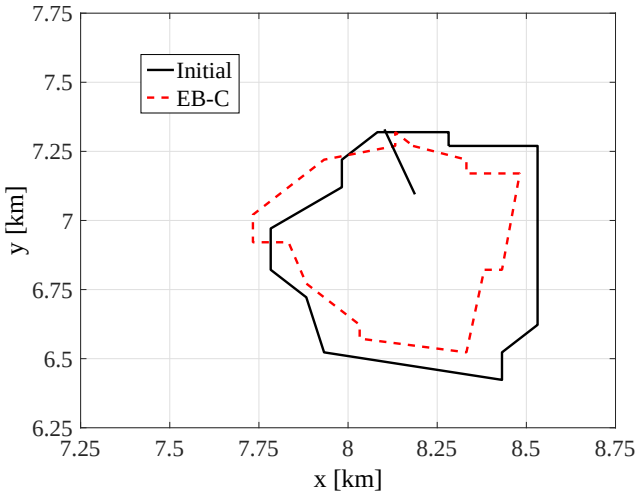
Indicator	Initial	LB	TB	QR	EB-C	EB-CS
$\overline{U_{imb}}$[%]	15.2	6.3	15.2	14.9	17	16.6
$\overline{T_{imb}}$ [$Mbps$]	1.01	4.57	0.28	0.52	4.09	0.79
$\overline{QoE_{imb,f}}$	0.51	0.50	0.49	0.32	0.46	0.33
$\overline{QoE_{imb,c}}$	0.4	0.45	0.30	0.35	0.18	0.14
$\overline{QoE_{imb,s}}$	0.51	0.56	0.45	0.42	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

services ($\overline{QoE_{imb,s}}$ = 0.26). This is the result of adjusting HOM on a per-adjacency and per-service basis.

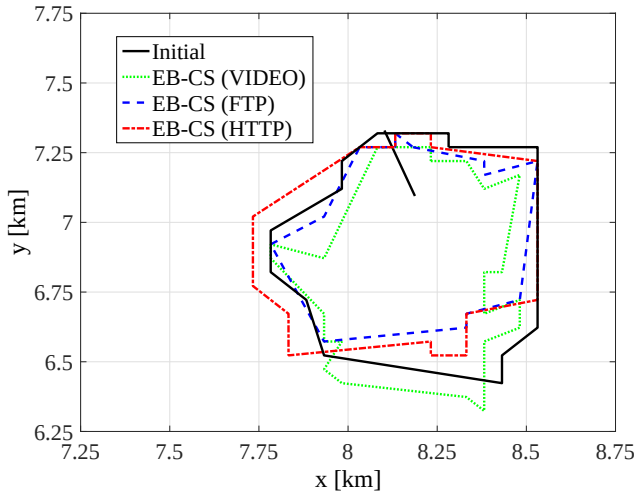
Finally, Figure 11 shows the cumulative distribution function of $\overline{QoE}(i,s)$ achieved by EB and EB-CS. It is observed that both balancing approaches deteriorate the QoE of the best cells/services to improve that of the worst cells/services. Focusing on the worst cells and services (lower left), 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 V, where EB-CS has the highest value for the QoE indicator representing the worst users (i.e., $\overline{QoE}^{(5\%-tile)}(i,s) = 2.36$).

The stronger QoE balancing effect with EB-CS is achieved by modifying cell service areas on a per-adjacency and per-service basis. Figure 12a illustrates how the service area of a particular cell is modified by EB-C at the end of the tuning process (15^{th} optimization loop). Note that EB-C algorithm modifies HOM only on a per-adjacency basis. Thus, the cell service area is shared by all services. Figure 12b depicts the service area of the same cell with EB-CS at the 15^{th} optimization loop. EB-CS algorithm modifies HOM on a per-adjacency and per-service basis. Therefore, the cell service area is different depending on the service (VIDEO, FTP or WEB). VoIP case is not shown because VoIP traffic in the network under analysis is negligible. Note that the cell service area produced by EB-C, shown in Figure 12a, is completely different from those of EB-CS, shown in Figure 12b. This flexibility of changing cell service areas on a per-adjacency and per-service basis is the reason for the superiority of EB-CS when equalizing the QoE among cells and services.

Another experiment has been carried out to check the ability of the proposed iterative algorithm (EB-CS) to adapt to changes in the number of users and the traffic mix. For this purpose, once the system is stable and has reached QoE balance, the number of users is modified by increasing the Poisson arrival rate of each service by 4%. This change modifies the balance point, so imbalance is expected to appear again and EB-CS starts to modify margins searching for the new balance point. After this second balance stage, the traffic mix is modified now by changing the percentage of FTP, VIDEO and WEB. WEB traffic is decreased by 24%, whereas FTP and VIDEO traffic is increased by 12%. Table VI summarizes the results for these traffic changes, showing the value of the QoE imbalance indicator, $\overline{QoE_{imb,c}}$, and the



(a) Initial and EB-C.



(b) EB-CS.

Fig. 12: Cell service area per service.

TABLE VI: System performance to network changes.

Stage	Initial network conditions		Network load increase, $\Delta\lambda$		Change in traffic mix, $\Delta\lambda(s)$	
Loop index	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

value of the QoE imbalance per service, $\overline{QoE_{imb,s}}$, at the beginning/end of each stage. It is observed that any change in network conditions produce a temporary QoE imbalance among cells and services, ($\overline{QoE_{imb,s}}$ = 0.43 and $\overline{QoE_{imb,c}}$ = 0.29 when traffic is generally increased, and $\overline{QoE_{imb,s}}$ = 0.48 and $\overline{QoE_{imb,c}}$ = 0.32 when traffic mix is changed). These imbalances are successfully corrected by EB-CS after a few iterations, leading network performance to a similar balance point than that of the first stage of the experiment in the 15^{th} iteration. Thus, it is shown that EB-CS can cope with fluctuations of traffic demand during a day.

C. Implementation issues

The EB-C algorithm is executed on a per-adjacency basis, and, therefore, its worst-case time complexity is $O(N_{adjs})$. In contrast, The EB-C algorithm is executed on a per-adjacency and per-service basis, so that its time complexity is $O(N_{adjs} * N_s)$. Both algorithms have been implemented with the Fuzzy Logic Toolbox in Matlab. For the considered scenario, consisting of 108 cells, 11664 adjacencies and 3 services, the average execution time of 1 iteration of EB-C and EB-CS is 2.8 and 6.5 seconds in a personal computer with a 3.6-GHz octa-core processor and 16 GB of RAM.

VI. Conclusions

In this paper, a self-tuning algorithm for adjusting handover margins in a LTE network has been proposed. The aim of the algorithm is to balance the QoE between cells and services. The proposed iterative algorithm changes handover margins between adjacent cells to push users from cells with a lower QoE to neighbor cells with a higher QoE. Two variants have been presented, depending on whether margins are tuned on a per-adjacency or per-adjacency and per-service basis. Method assessment has been carried out in a dynamic system-level LTE simulator implementing a realistic macrocellular scenario. Results have shown that the average QoE of cells becomes less imbalanced after parameter tuning. Specifically, the average QoE difference among cells is reduced by 0.22 and 0.26 MOS points with EB-C and EB-CS, respectively, with an average HOM change between 0 and 6 dB depending on the service.

The proposed algorithm is conceived as a centralized solution for the network management system, since QoE statistics needed by the algorithm are currently obtained by packet inspection techniques in selected core network interfaces [42]. The underlying iterative algorithm is devised to be executed after each reporting output period (e.g., 1 hour). Such a time window ensures reliable QoE measurements for long video streaming sessions. If faster changes are needed, the proposed algorithm could be executed with a shorter periodicity (e.g., minutes), provided that reliable QoE estimates are available. It is envisaged that such information will be delivered to SON frameworks as part of big data generated by future 5G mobile communication systems.

Acknowledgments

This work has been partially funded by the Spanish Ministry of Economy and Competitiveness (TEC2015-69982-R), the Spanish Ministry of Science, Innovation and Universities (RTI2018-099148-B-I00) and the Horizon 2020 project ONE5G (ICT-760809), receiving funds from the European Union.

References

[1] Nokia Siemens Networks, "Understanding Smartphone Behavior in the Network," White paper, 2011.
[2] Ericsson AB, "Ericsson Mobility Report", Nov. 2017.
[3] S. Barakovic and L. Skorin-Kapov, "Survey and Challenges of QoE Management Issues in Wireless Networks," vol. 2013, Mar. 2013.
[4] NGMN, "Next Generation Mobile Networks Recommendation on SON and O&M requirements", 2008.

[5] V. F. Monteiro, D. A. Sousa, T. F. Maciel, F. R. M. Lima, E. B. Rodrigues, and F. R. P. Cavalcanti, "Radio resource allocation framework for quality of experience optimization in wireless networks," *IEEE Network*, vol. 29, no. 6, pp. 33–39, Nov. 2015.
[6] H. Nam, K. H. Kim, J. Y. Kim, and H. Schulzrinne, "Towards QoE-aware video streaming using SDN," in *2014 IEEE Global Communications Conference*, Dec. 2014, pp. 1317–1322.
[7] S. Hämäläinen, H. Sanneck, and C. Sartori, "LTE Self-Organizing Networks (SON): Network Management Automation for Operational Efficiency Hardcover," 2012.
[8] 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.
[9] 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.
[10] 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.
[11] 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.
[12] 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.
[13] 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.
[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] 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.
[16] 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.
[17] 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.
[18] 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.
[19] 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.
[20] 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.
[21] 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.
[22] 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.
[23] 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.
[24] 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.
[25] 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.
[26] 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.
[27] 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.
[28] 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.
[29] 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.
[30] Scenarios, requirements and KPIs for 5G mobile and wireless system ICT-317669 METIS project, D6.1 v1-2, 2013.
[31] 3GPP TSG RAN WG1, 3GPP TR 25.892 v6.0.0; Feasibility Study for Orthogonal Frequency Division Multiplexing(OFDM) for UTRAN Enhancement(Rel-6), pp. 62-63, 2004.
[32] 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.
[33] Y. Fang, I. Chlamtac, and Yi-Bing Lin, "Channel occupancy times and handoff rate for mobile computing and PCS networks," *IEEE Transactions on Computers*, vol. 47, no. 6, pp. 679–692, Jun. 1998.
[34] 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.
[35] 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.
[36] ITU-T G.114 Recommendation, "One-Way Transmission Time,", 2003.
[37] 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.
[38] T. Ross, *Fuzzy logic with engineering applications*. McGraw-Hill, 1995.
[39] 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.
[40] 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, Sep. 2011.
[41] L. Gimenez, I. Z. Kovács, J. Wigard, and K. I. Pedersen, "Throughput-Based Traffic Steering in LTE-Advanced HetNet Deployments," Sep. 2015.
[42] 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.

María Luisa Marí-Altozano received her M.S. degree in Telecommunication Engineering from the University of Málaga, Spain, in 2012. From 2013 to 2016, she was with Ericsson in a collaborative project with the University of Málaga. Since 2017, she has been working toward the Ph.D with the Communication Engineering Department, University of Málaga. Her interests are focused on self-optimization of mobile radio access networks based on quality of experience.

Salvador Luna-Ramírez received his M.S in Telecommunication Engineering and the Ph.D degrees from the University of Málaga, Spain, in 2000 and 2010, respectively. Since 2000, he has been with the department of Communications Engineering, University of Málaga, where he is currently Associate Professor. His research interests include self-optimization of mobile radio access networks and radio resource management.

Matías Toril received his M.S in Telecommunication Engineering and the Ph.D degrees from the University of Málaga, Spain, in 1995 and 2007 respectively. Since 1997, he is Lecturer in the Communications Engineering Department, University of Málaga, where he is currently Full Professor. He has authored more than 110 publications in leading conferences and journals and 3 patents owned by Nokia Corporation. His current research interests include self-organizing networks, radio resource management and data analytics.

Carolina Gijón received her B.S. degree in Telecommunication Systems Engineering from the University of Málaga, Spain, in 2016. Currently, she is working towards the Ph.D. degree. Her research interests include self-organizing networks and radio resource management.