Backhaul Aware User-Specific Cell Association Using Q-Learning

Paulo Valente Klaine, *Student Member, IEEE*, Mona Jaber, *Student Member, IEEE*, Richard Demo Souza, *Senior Member, IEEE*, and Muhammad Ali Imran, *Senior Member, IEEE*

***Abstract*— With the advent of network densification and the development of other radio interface technologies, the major bottleneck of future cellular networks is shifting from the radio access network to the backhaul. The future networks are expected to handle a wide range of applications and users with different requirements. In order to tackle the problem of downlink user-cell association, and allocate users to the best cell, an intelligent solution based on reinforcement learning is proposed. A distributed solution based on Q-Learning is developed in order to determine the best cell range extension offsets (CREOs) for each small cell (SC) and the best weights of each user requirement to efficiently allocate users to the most appropriate SC, based on both backhaul constraints and user demands. By optimizing both CREOs and user weights, a user-specific allocation can be achieved, resulting in a better overall quality of service. The results show that the proposed algorithm outperforms current solutions by achieving better user satisfaction, mitigating the total number of users in outage, and minimizing user dissatisfaction when satisfaction is not possible.**

***Index Terms*— Self organizing networks, 5G, cell association, backhaul, reinforcement learning, Q-learning.**

I. Introduction

THE Next Generation of Mobile Networks (NGMN), 5G, is under heavy pressure in order not only to overcome limitations of current cellular networks, but also to enable and push the boundaries of future networks to a next level. With 5G being in the imminence of commercial deployment, a consensus between some of its requirements has been agreed upon. Some requirements for 5G networks are [1]–[3]: address the growth in coverage and capacity; provide peak data rates at the gigabit level; support ultra high reliability and low latency; provide better Quality of Service (QoS) to end-users; coexist with different Radio Access Network (RAN) technologies; provide higher network energy efficiency.

In addition, one key differentiator of 5G networks is that their users will have very diverse network requirements and expectations. As such, current cell association approaches, which are centered in two major assumptions (that the radio interface is the bottleneck and that there is little variation in user requirements) renders static association approaches ineffective. Thus, due to the rising challenges of 5G networks and demands of users, a new paradigm must be enabled in the association of users and Base Stations (BSs). Furthermore, new technologies, such as network densification and new air interfaces, are shifting the bottleneck of future cellular networks to the backhaul [4]–[7]. However, due to the sub-optimality of current cell association methods, the backhaul of the associated cell might not be sufficient to satisfy specific user needs, and as such, more intelligent approaches that consider the end-to-end connection (RAN and backhaul) and users requirements is needed [7].

In this paper, a user-specific cell association algorithm is proposed in order to tackle the problem of allocating users with distinct requirements to the best fitting Small Cells (SCs) with different backhaul parameters. The proposed solution aims to tune both SCs Cell Range Extension Offsets (CREOs) and user requirements weights, in order to determine the best combination of CREOs and weights that satisfy each user, or, if that is not possible, minimize its dissatisfaction. Users can have different needs in terms of network parameters, such as throughput, latency, resilience, energy efficiency, or security, while each SC has certain attributes associated with these parameters as well. The main idea and innovation behind the proposed Reinforcement Learning (RL) based algorithm is, to perform two different optimizations, one at the network level, in which the algorithm optimizes the CREO of SCs via Q-Learning, followed by another optimization at the user level, in which the algorithm determines the best weights for each user, also via Q-Learning. Combining both Q-Learning solutions and optimizing both network and user parameters, the proposed solution is able to provide user-specific allocation and achieve better results in terms of user satisfaction and QoS.

A. Related Work

Since the main bottleneck of future cellular networks is expected to shift from the RAN to the backhaul, its optimization and the BS association problem have gained increased

Manuscript received May 29, 2018; revised November 14, 2018 and April 3, 2019; accepted April 29, 2019. Date of publication May 14, 2019; date of current version July 10, 2019. This work was supported by the Distributed Autonomous and Resilient Emergency Management System (DARE) Project under the EPSRC's Global Challenges Research Fund (GCRF) Allocation under Grant EP/P028764/1, and in part by the National Council for Scientific and Technological Development (CNPq), Brazil, under Grant 304503/2017-7. The associate editor coordinating the review of this paper and approving it for publication was M. Li. *(Corresponding author: Paulo Valente Klaine.)*

P. Valente Klaine and M. A. Imran are with the School of Engineering, University of Glasgow, Glasgow G12 8QQ, U.K. (e-mail: paulo.valenteklaine@glasgow.ac.uk; muhammad.imran@glasgow.ac.uk).

M. Jaber is with Fujitsu Laboratories on Europe, Hayes UB4 8FE, U.K. (e-mail: m.jaber@uk.fujitsu.com).

R. D. Souza is with the Department of Electrical and Electronics Engineering (EEL), Federal University of Santa Catarina, Florianópolis 88040-370, Brazil (e-mail: richard.demo@ufsc.br).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2019.2915083

This work is licensed under a Creative Commons Attribution 3.0 License. For more information, see http://creativecommons.org/licenses/by/3.0/

attention recently. In [5], the authors optimize the backhaul and BS assignment problem using a novel heuristic algorithm. However, only user throughput was considered as a QoS metric. Moreover, due to its heuristic nature, the proposed solution might not be computationally feasible, as it must determine for every network configuration, the parameters of all users and BSs. Olmos *et al.* [6], build upon [5], and consider a more generic model based on Markov chains to solve the problem of cell selection with backhaul constraints. Despite being more general, the authors do not consider user QoS requirements and both [5], [6] do not consider a heterogeneous network scenario, in which BSs have different transmit powers and backhaul characteristics.

In [8], a method to balance network load of BSs backhaul based on their geometric location is proposed. In [9], the authors aim to optimize the user-cell association in a decoupled uplink and downlink heterogeneous network scenario. However, both [8], [9], do not consider user QoS requirements when performing cell association. In [8], for example, the authors attempt to perform backhaul load balancing, while in [9] the Reference Signal Received Power (RSRP), cell load and backhaul capacity were regarded as limiting factors in cell association. In [10], the authors optimize the network backhaul throughput by improving the cell association process. However, as the authors mention, conventional search algorithms would not work for this problem, as the cell association problem is NP-hard, becoming infeasible for a large number of users an BSs. Thus, they propose a heuristic centralized algorithm to associate users. However, [10] also does not consider users with different QoS requirements, nor cells with different backhaul links. Pantisano *et al.* [11], address the the cell association issue by considering that SCs can cache content in order to overcome backhaul capacity limitations and improve users QoS. However, for the proposed solution to work, user location must be known (or estimated) and only user throughput was considered as a QoS requirement.

Han *et al.* [12], aim to optimize user association and resource allocation in a heterogeneous network considering radio resource consumption, energy and backhaul constraints. However, because the problem is NP-hard, the authors propose decomposition methods to reduce the problem to a smaller version, and to build an online solution. Also, the authors considered optimizing only the resource allocation and utility of the network via cell association and did not attempt to improve user QoS. In [13], network frame design, resource allocation and user association optimization in a heterogeneous massive multiple input multiple output (MIMO) network scenario is proposed. Although this solution can adapt to different network scenarios, it does not investigate user QoS requirements and only optimizes total network sum rate.

Ma *et al.* [14], investigate the user association and resource allocation in a massive MIMO heterogeneous network scenario and attempt to maximize network utility. The authors develop an analytical solution and despite considering a heterogeneous network scenario, users QoS requirements is not considered. On the other hand, Lee *et al.* [15] address the user cell association problem considering backhaul load balancing and minimizing user call blocking probability. The problem is formulated as 0-1 integer problem, but due to its complication it is relaxed to become a convex optimization problem. Lastly, works by Jaber *et al.* [7], [16], [17], propose an algorithm based on Q-Learning to solve the cell association problem by considering backhaul limitations. The proposed distributed solution aims to tune the CREO of SCs so that users can connect to the SCs with the backhaul that would better match their needs.

However, due to the analytical or heuristic aspect of these solutions, [5], [6], [8]–[15] may not be adaptable enough in order to enable future cellular network paradigms, such as Self Organizing Networks (SON), as they often require unrealistic assumptions. Most of these works require the knowledge of how many users and BSs or SCs are in the network, or user positions and requirements, for example. Also, as it could be seen, almost none of the reviewed works consider user QoS requirements or the utilization of different applications and backhaul links. Furthermore, as these works often depend on searches or analytical expressions, periodical optimizations are often required and no network or user data is utilized, not fully exploiting the potential of SON. Moreover, as future networks are expected to be much more intelligent and adaptable, by using historical or online data, solutions that do not require lots of assumptions and that can learn intrinsic patterns in data as the network changes are preferred. Thus, more general solutions that can analyze data and take online decisions, such as machine learning, should be designed [3].

Thus, in this work a distributed RL based solution is proposed. Due to the inherit nature of RL solutions, a machine learning technique based on a goal-seeking approach [18], a model-free solution to the problem of user-cell association is proposed. In this case, differently than [5], [6], [8]–[15], no assumptions or prior knowledge are necessary, as all the data needed for the algorithm to learn is generated online by the network and its users. As such, the proposed method in this paper is more robust and generic, as it can adapt to different network conditions, while the previously reviewed literature require previous knowledge about their environment and are limited by the specific application designed to fit the solutions.

Other works, such as [7], [16], [17], also utilize RL to perform cell association, however the main drawback of these solutions is that only network parameters are optimized and user parameters and requirements are assumed to be random, achieving a sub-optimal solution. In [7], [16], and [17], for example, it is assumed that users weights are binary random and do not depend on users QoS requirement. As such, the weights associated for each user and its requirements would not always conform with its demands leading to a limited optimization. Furthermore, this assumption can lead to network resource wastage, as users that did not have a stringent demand, could end up having high weights, while more demanding users could be assigned low weight. As such, the works in [7], [16], and [17] do not optimize user parameters, but only network parameters (SCs CREOs), leading to a network centric (or BS-specific) solution and are not capable of solving the problem for each user individually. In addition, in [7], [16], and [17], the proposed solutions are denoted as

user-centric because the metrics evaluated are considered from a user perspective, but they do not actually perform any user optimization.

Thus, unlike [7], [16], [17], in which user weights are assumed to be random, in this work, an optimization of both network and user parameters is performed, so that user-specific cell association can be achieved, leading to better network resource consumption and user satisfaction. Moreover, as future cellular networks are expected to be more user-oriented and deal with several applications with different requirements, it is only natural that solutions which try to optimize individual user and network parameters are developed. In addition, not only will different types of users need to be addressed, but also the same user could have different requirements at different times of the day, as it utilizes different applications. Hence, a solution that can adapt itself to different user needs and that can treat users differently based on their current requirements is needed, and, for that to be possible, an intelligent user-specific approach is necessary.

B. Objectives and Contributions

As seen from the literature review, backhaul-aware cell association has been a focal topic of research in the recent years. However, solutions in this area still remain network-centric and agnostic to the diversity of user requirements. On the other hand, BS-centric association has been studied in the past in the works of Jaber *et al.* [7], [16], [17], which endows the cell association process with the ability to distinguish and prioritize users requirements. However, these efforts are still limited by the network parameter tuning and do not account for the users' ability to improve its choice. This work is the first to address this issue of both tuning network and user parameters and to propose a two-step association scheme that outperforms prior state-of-the-art solutions with minimum added complexity.

The proposed solution is based on RL, more specifically Q-learning, and it is shown to be robust and flexible to enable an autonomous cell association, enabling the stringent requirements needed for future cellular networks in a heterogeneous and diverse environment. By optimizing both network and user parameters, the proposed solution is able to allocate to each user what it needs without wasting network resources and making other users suffer, this, in its turn, enables more users to be allocated to the network while also satisfying their needs, improving individual and overall (by consequence) QoS. The contributions of this paper are summarized as follows:

- To provide an end-to-end paradigm in terms of downlink user-cell association, considering the radio access network, backhaul conditions and users QoS requirements;
- To optimize the user-cell association process by delivering to each user only what was requested, minimizing network resource wastage;
- To perform both network and user parameters optimization, considering both network constraints and user requirements to achieve user-specific cell association, in an adaptable and intelligent manner via RL (Q-Learning).

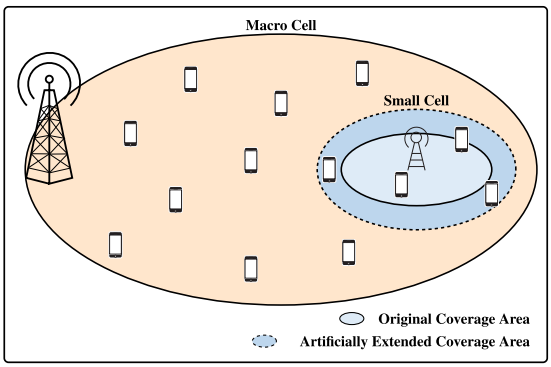


Fig. 1. Example of how applying CREO can change the cell association of users in the network.

The remainder of this paper is organized as follows. Section II overviews the problem and explains key concepts, such as CREO, RL and Q-Learning. Section III presents the system model and the proposed solution, Section IV discusses numerical results and Section V provides a summary of the key finds of the paper and suggestions for future works.

II. Background

A. Cell Range Extension Offset (CREO)

Current downlink cell association methods rely solely on radio interface parameters, such as the RSRP or Signal to Interference plus Noise Ratio (SINR), to determine which cell a user should be associated to. In the future, however, as NGMNs are expected to be much more diverse and heterogeneous by nature, the bottleneck of cellular networks will shift from the RAN to the backhaul and current association methods will probably not be adequate [5]–[7], [17], [19]. Since the transmit power of a macro BS and a SC are very different, much in favor of the macro BS, the problem of load imbalance in the network is created. If only the RSRP or SINR is considered, most users would prefer to connect to the macro BS, as it has a higher transmit power, overloading it and leaving the SCs underloaded [20], [21].

To solve this issue, a technique known as CREO was developed, in which SCs artificially extend their coverage area by adding an artificial offset to the user perceived RSRP in the association process [22]. Figure 1 shows an example, in which, a SC is overlaid on top of a macro cell. Initially, the SC covers only the light blue area, however, when CREO is applied, the SC's coverage area is artificially extended to the darker dotted area. Hence, users within this greater coverage area will now prefer to connect to the SC instead of the macro cell.

Although adding a CREO can provide several benefits, such as enhanced uplink data rates, increased capacity (by means of load balancing), and improving network robustness, by making SCs less sensitive to their deployment location, only artificially increasing the perceived transmit power of SCs is not enough to improve the performance of the network [22]. If a fixed CREO was applied to all SCs the problem of load balance would be solved, but the problem of backhaul congestion would be created. Also, since tuning SCs CREOs only considers the problem from a RAN perspective, this would not completely attend different users requirements and

would not be able to provide enough QoS, nor meet the requirements for future networks [7], [16], [17].

Hence, it is clear that optimizing only the radio interface is not enough, and that a joint optimization between the radio interface and the backhaul is needed. In addition, SCs should also adjust their CREOs more efficiently and intelligently, so that users with different requirements can connect to the SCs that best fit their needs. Also, if only the parameters from the network side are tuned, only a group of users can be satisfied, as those with the highest priority to a specific requirement [7], [17]. Thus, in order to provide a user-specific cell association, which attends to a wide range of user requirements, it is clear that an optimization at the user side must also be done, so that users can be intelligently associated to the best fitting SC.

B. Reinforcement Learning (RL)

RL is a machine learning technique based on a goal-seeking approach [18]. In contrast to other machine learning techniques, such as supervised learning, in which the system learns by analyzing examples provided by an external supervisor, in RL, the learner must discover which actions to take by trying them [18], [23]. In RL, a system, called an *agent*, interacts with its surroundings, the *environment*. These two elements interact continuously, with the agent selecting different actions based on new situations (states) presented by the environment. After taking an action, the agent receives a *reward* from the environment. This reward can be either a positive value, if the action was good, or a negative value (a *penalty*), if the action was bad. The goal of a RL system is to maximize the total reward and to achieve it, an agent must not only exploit the best actions currently known, but also explore new actions, in order to determine if there are better possible actions. This is known as the exploration-exploitation trade-off [18], [23].

1) Q-Learning: one of the most used algorithms in RL is Q-Learning. First proposed by Watkins, in [24], Q-Learning is a learning method that learns an action-value function, Q, which represents the expected value of an agent being in a certain state and taking a specific action. However, Q-Learning, in contrast to others RL methods, directly approximates the optimal action-value functions, Q^*, independently of the policy being followed (it is guaranteed to converge for any chosen policy), hence, the RL problem becomes simpler to implement [18], [24].

Q-Learning is a learning method, that at each step at a state, s_t, chooses an action, a_t, that maximizes the action-value function $Q(s_t, a_t)$. This function indicates how good is taking action, a, at state, s, according to a reward, r. More formally, Q-Learning is defined as [18], [23], [24]

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_{t+1} + \phi \max_a Q(s_{t+1}, a) - Q(s_t, a_t)], \quad (1)$$

where $Q(s_t, a_t)$ is the current action-value function, α, is the learning rate, r_{t+1} is the expected reward at the next time step, ϕ is the discount factor and $\max_a Q(s_{t+1}, a)$ is an estimate of the optimal future action-value function at the next time step.

III. System Model

A downlink heterogeneous network scenario is considered. In this scenario, a single macro cell is positioned in the center and is divided in m sectors, denoted as $\mathbb{M} = \{M_1, \ldots, M_m\}$. On top of each sector, n SCs, $\mathbb{S} = \{S_{M_1,1}, S_{M_2,1}, \ldots, S_{M_m,n}\}$, are randomly distributed, and the set with all cells in the system is defined as $\mathbb{C} = \mathbb{M} \cup \mathbb{S}$. Also, each SC has a single non-ideal last-mile connection, while the backhaul connection between the macro BS and the core network is assumed to be ideal. In addition, each SC is assumed to have η adjustable CREOs, one for each backhaul parameter. These offsets are defined as $\mathbb{O} = \{O_{1,1}, \ldots, O_{c,\eta}\}$, with $c \in [1, |C|]$, and each CREO can assume values from $\mathbb{V} = \{V_1, V_2, \ldots, V_{max}\}$.

On top of this network, $\mathbb{U}$ users are distributed and each sector is considered to have k users, with higher concentration near the SCs, $\mathbb{U} = \{u_{1,1}, u_{m,2}, \ldots, u_{m,k}\}$, and $p = m \cdot k$ is the total number of users. Furthermore, each user has μ required parameters ($\mu = \eta$), which can be seen as QoS parameters that a user is concerned about. In addition, users QoS requirements are represented by $\mathbb{E} = \{E_{1,1}, \ldots, E_{p,\mu}\}$, and, for each requirement, each user has an associated weight to it (defined by the application, for example), denoted as $\mathbb{W} = \{W_{1,1}, \ldots, W_{p,\mu}\}$. These weights are adjustable, and can assume values in $\mathbb{G} = \{G_1, G_2, \ldots, G_{max}\}$ and the network monitors the QoS of users, represented by $\mathbb{E}' = \{E'_{1,1}, \ldots, E'_{p,\mu}\}$.

It is assumed that each user can connect to a single cell at a time and cells have limited radio and backhaul resources. Whenever a user is allocated to a BS, it consumes one Resource Block (RB) and both macro and SCs have a limited amount of RBs. This assumption is made for comparison purposes and, as such, the resulting gains are not defined by the number of RBs each user has, but by finding a more suitable cell association.[1] Regarding interference, macro and SCs share the same frequency band while enhanced Inter-Cell Interference Coordination (eICIC) and Almost Blank Subframes (ABS) are utilized in order to mitigate inter-cell interference between macro and SCs [7], [16], [17], [25]. In addition, a frequency reuse factor of one is considered. Lastly, as per 3GPP current standards, the CREO of serving and neighboring cells are broadcast to users in the vicinity using common control channels.

A. Cell Association

In order to associate users to cells, the received signal power from each cell is computed. The RSRP, $R_{u,c}$, (in dB) of user u and cell c, can be expressed as

$$R_{u,c} = P_c - 10 \cdot \log_{10}(N_{sc}) - H_{u,c} - L, \quad \forall\, c \in \mathbb{C}\,, \quad (2)$$

where P_c is the transmit power of cell c, N_{sc} is the total number of sub-carriers[2] in cell c, $H_{u,c}$ is the path loss between user u and cell c, and L is the penetration loss.

[1] This is assumed for the sake of simplicity, but, in practice, RB allocation could be done dynamically.

[2] Sub-channels in a specific time-slot are considered (definition of a RB).

A log-distance path loss is assumed and is defined as [26]

$$H_{u,c} = \Psi + 10\gamma \cdot \log_{10}\left(d_{u,c}\right) + X_{\sigma}\,, \tag{3}$$

where Ψ is a propagation constant, γ is the propagation exponent, $d_{u,c}$ is the distance between the user u and cell c, and X_{σ} is defined as the log normal shadowing component.

Based on the received power from each cell, users are then going to decide which cell to associate with. This is done by a ranking system, which takes into account only the perceived RSRP, if the user is trying to connect to the macro cell, or the RSRP combined with the SC's CREO and user weights, in case it is a SC. The cell ranking can be expressed as

$$K_{u,c} = \begin{cases} R_{u,c}\,, & \text{if } c \in \mathbb{M}. \\ R_{u,c} + \dfrac{1}{\eta}\displaystyle\sum_{i=1}^{\eta} W_{u,i} \cdot O_{c,i}\,, & \text{if } c \in \mathbb{S}. \end{cases} \tag{4}$$

After each user ranks every cell, the cell association process begins. If a BS has enough space to accommodate a user request, and the user's SINR is above a certain threshold, then the user connects to the desired cell. The perceived SINR of user u, and cell c, can be calculated as

$$\text{SINR}_{u,c} = \begin{cases} \dfrac{R_{u,c}}{N + \zeta_{ABS}\displaystyle\sum_{i=1, i\neq c}^{n} \omega_i R_{u,i}}, \\ \quad \text{if } c \in \mathbb{M}. \\ \dfrac{R_{u,c}}{N + \zeta_{ABS}\displaystyle\sum_{i=1, i\neq c}^{n} \omega_i R_{u,i} + (1-\zeta_{ABS}) R_{u,M}}, \\ \quad \text{if } c \in \mathbb{S}. \end{cases} \tag{5}$$

where N is the noise power, ζ_{ABS} corresponds to the fraction of ABS time that the SCs transmit (in percentage — between 0 and 1), ω_i is the load of SC i, $\sum_{i=1, i\neq c}^{n} R_{u,i}$ is the RSRP from other SCs belonging to the sector that the user is connected to and $(1-\zeta_{ABS})R_{u,M}$ is the interference from the macro cell sector that the user belongs to, scaled down by the percentage of time that the macro cell is not transmitting due to ABS.

If the BS does not have enough RBs or if the SINR is not high enough, then the user tries to connect to the next best cell. This process is repeated for the next four BSs until a connection can be established, or if that is not possible, the user is then assumed to be out of coverage in that time slot [7]. If however, a BS has more than enough RBs to serve its users, the remaining RBs are assumed to be unused during that time slot. If a user is connected, then the maximum user throughput is estimated as

$$T_{u,c} = B \cdot \log_2\left(1 + \text{SINR}_{u,c}\right), \tag{6}$$

where B is the bandwidth occupied by one RB. In addition, the amount of backhaul throughput required for all users connected to a cell is computed as

$$\lambda_c = \rho_c \cdot \sum_{u=1}^{U_c} T_{u,c}\,, \tag{7}$$

where $\rho_c > 1$ represents the backhaul overhead [7], and U_c denotes the total number of users connected to the cell.

Since SCs have limited backhaul capacities, whenever their required backhaul throughput exceeds its total capacity, the effective throughput of all users connected to that cell is reduced. The effective throughput of users is expressed as

$$T'_{u,c} = \begin{cases} T_{u,c}\,, & \text{if } \lambda_c \leq C_c\,, \\ T_{u,c} - \dfrac{C_c - \lambda_c}{U_c}\,, & \text{if } \lambda_c > C_c\,, \end{cases} \tag{8}$$

where C_c is the maximum backhaul capacity of cell, c.

The throughput of each cell, c, can be calculated as

$$T_c = \sum_{u=1}^{U_c} T'_{u,c}\,, \tag{9}$$

and the total throughput of the system can be determined by

$$T = \sum_{c=1}^{|\mathbb{C}|} \sum_{u=1}^{U_c} T'_{u,c}. \tag{10}$$

IV. Proposed Scheme

The objective of the proposed system is to maximize the total effective cumulative throughput of all users, given a set of constraints. This can be done by tuning both CREOs of SCs and user weights in a centralized manner. However, centralized solutions can be impractical, as it would require an extra layer of communication between the SCs, users, and the centralized unit in order to disseminate changes in the network, increasing signaling overhead. In addition, synchronization would become an issue, as using an outdated values fetched from the centralized unit could impact the performance of the system. As such, a distributed approach is preferred.

The proposed solution aims to divide the global optimization problem of maximizing the total system throughput into smaller sub-problems. These sub-problems can be defined as maximizing the throughput of each individual cell of the system, given certain backhaul constraints. More formally, the optimization objective can be formulated as

$$\underset{\mathbb{O},\mathbb{W}}{\text{maximize}} \;\; T_c\left(\mathbb{O}, \mathbb{W}\right) \tag{11a}$$

$$\text{Subject to } \text{RB}_c \leq \text{RB}_{max} \tag{11b}$$

$$C_c \geq \lambda_c \tag{11c}$$

$$\sum_{u=1}^{U_c} \frac{E'_{u,\mu} - E_{u,\mu}}{E_{u,\mu}} \leq \theta_{\mu}, \quad C_c \geq \lambda_c, \quad \forall\, \mu \tag{11d}$$

$$\frac{E'_{u,\mu} - E_{u,\mu}}{E_{u,\mu}} \geq 0, \quad \forall u \in \mathbb{U}, \quad \forall\, \mu. \tag{11e}$$

where θ_{μ} represents a threshold that determines how much over satisfaction, on average, is allowed for each parameter μ. Note that constraints (11e) and (11e) have their signal changed when latency is considered (latency value is minimized, while the other parameters are maximized).

As it can be seen, maximizing the throughput of each individual cell of the network, in (11a), is subject to four different constraints. The first constraint (11c) states that

each cell c is limited by a maximum number of RBs, or in other words is limited in the number of users it can serve. The second one, (11d), states that each cell has a maximum backhaul capacity and that if the total capacity required by the users associated to cell c exceeds it, then the throughput of all users associated to that cell is reduced, as defined in (8). In addition to this constraint, each SC keeps track of how many RBs it has, so the CREO optimization takes into account both radio and backhaul parameters. The third constraint, (11e), states that the average satisfaction level of users connected to a cell must be below a certain threshold, given that there are backhaul resources available in the SC. This constraint attempts to limit the amount of over-satisfaction users can have and aims to distribute better the backhaul resources. By respecting this constraint, the system perceives allocating few users with too much of a certain resource as a bad maneuver, as more users would be left starving. Hence, the system will try to find a better user-cell association in order to reduce this over-satisfaction and distribute better the backhaul resources. It is also important to note that this constraint does not deal with resource allocation, it only attempts to satisfy users by changing the cell each user is associated with, specially in idle mode. For example, if a user requires 5ms latency, but is associated with a SC that provides 1ms of latency, this association is not very efficient, as this user is over-satisfied and is wasting resources that could serve other users that require lower latency. Thus, changing association of this user to a SC with higher latency would be more efficient, as the user could still be satisfied and the precious latency resource is freed for a more demanding user. Based on that, the value of θ_μ can then be chosen as a system parameter, which determines how much over-satisfaction, on average, is allowed at the expense of less satisfaction of other users. However, by considering only these two constraints, as in [7], [16], and [17], only the aggregate performance of users connected to a certain cell is optimized, making the system not able to track individual user performance.

Based on these issues of dealing only with the aggregate performance of users, a fourth constraint, (11e), is proposed. This constraint states that each user should be allocated more than its target QoS (each user should be satisfied). It should be noted that constraints (11e) and (11e) have opposing optimization objectives. Consequently, satisfying both constraints results in a solution where each user measures a QoS value that is as close as possible to its target $E'_{u,\mu} \rightarrow E_{u,\mu}$. In other words, each user should be allocated only enough of each resource, so that it is satisfied. By doing this, the system guarantees that each user is satisfied, while enabling more backhaul resources to be shared, avoiding the limitation of being constrained by the aggregate performance of users as in [7], [16], and [17], and achieving a user-specific solution.

In order to accomplish the objective defined in (11a), a formulation based on RL is proposed, consisting of two different optimization processes. First, an optimization from the network perspective of SCs CREOs is performed. In this optimization, the SCs learn the best set of CREOs, $O_{c,\eta}$, that satisfies the majority of their users (this optimization addresses constraints (11d) and (11e)), similar to the optimization performed in [7], [16], and [17]. After that, each user will optimize its own weights, $W_{u,\mu}$, also via a RL formulation, and as highlighted in the introduction, this is the main contribution of the paper, achieving a user-specific cell association.

A. SCs Learning

In order to solve the optimization problem in (11a), an intelligent and distributed solution based on Q-Learning is proposed. The SCs belonging to $\mathbb{S}$ have a set of η adjustable CREOs, that can be learned in order to maximize the perceived throughput of each SC. Hence, each SC is considered to be an agent and the network is the environment.

The actions, a_c, that each SC can take are defined by the changes in their CREO values, $O_{c,\eta}$, described by $\mathbb{V}$. In addition, each SC is considered to have η attributes, and one adjustable CREO for each attribute. Each CREO is learned and adjusted independently from one another (each SC considers independent state-action pairs for each parameter, η). The policy that the agents follow in order to take actions is a completely greedy one, in which the best action is chosen at every iteration. In terms of states, each SC can be in one out of three possible states:

- State 1, if constraint (11d) is not satisfied (the backhaul is currently overloaded).
- State 2, if constraint (11d) is satisfied and (11e) is not (the backhaul has resources available, but users have not been associated in an optimal way, as there are users over-satisfied).
- State 3, if both constraints (11d) and (11e) are satisfied — the SC can accommodate more users (its backhaul is not overloaded) and the user association is good enough.

More formally, the states, υ_c, that each SC can be are

$$\upsilon_c = \begin{cases} 1, & \text{if } \lambda_c > C_c\,, \\ 2, & \text{if } \sum_{u=1}^{U_c} \dfrac{E'_{u,\mu} - E_{u,\mu}}{E_{u,\mu}} > \theta_\mu \mid C_c \geq \lambda_c\,, \\ 3, & \text{otherwise.} \end{cases} \quad (12)$$

For each state-action pair a reward, r_{υ_c,a_c}, is associated, and it can be seen as a value corresponding to the consequence of taking certain action and ending up in a specific state [18]. The reward in this case is defined as

$$r_{\upsilon_c,a_c} = \begin{cases} A_1, & \text{if } \upsilon_c = 1\,, \\ \sum_{u=1}^{U_c} \dfrac{E'_{u,\mu} - E_{u,\mu}}{E_{u,\mu}}, & \text{if } \upsilon_c = 2\,, \\ A_2 \cdot \dfrac{C_c - \lambda_c}{C_c}, & \text{if } \upsilon_c = 3. \end{cases} \quad (13)$$

In State 1, as a cell should always try to avoid having its backhaul overloaded, a low reward (e.g. $A_1 = -1000$) is defined. In State 2, however, despite the backhaul of the SC not being overloaded, the association performed is not the best, as some users are over-satisfied (indicating that other users might be starving). Since this state is also not ideal, but not as bad as State 1, a low reward based on the percentage difference between what the majority of users achieved and requested is assigned. Lastly, in State 3, which is the best

possible state a cell can be, the reward is defined as the percentage difference between a cell's maximum and current backhaul capacity, multiplied by a constant (e.g. $A_2 = 100$), and represents how many more users are able to fit in cell c. A constant is added so that whenever a cell moves from one state to state 3, the algorithm will yield a high reward value.

Based on that, for each state-action pair and its reward, each agent learns and updates its η Q-Tables. Since these tables depend only on the state-action pairs, each Q-Table, $Q_{c,\eta}$ is an $[a_c \times \upsilon_c]$ matrix, and, for each iteration of the algorithm, they are updated following the formulation in (1). Lastly, since the algorithm operates in an iterative manner, it is only natural that a stopping criteria is devised to guarantee the convergence of the proposed solution. In this case, two stopping criteria are formulated. The first guarantees that the optimization is not perform indefinitely, as such, the SCs perform their CREOs optimization for a maximum number of iterations (M_{sc}), while the second states that if the reward does not improve from one iteration to the other more than a threshold, (r_{th}), it is also accepted that the algorithm has converged.

B. User Weights Learning

After the SCs learn their CREOs, which represent the best offset that will please the majority of the users connected to each cell, the user weights learning begins. Each user learns the weights, $W_{u,\mu}$, given to each parameter, μ, also using a Q-Learning formulation. In this learning problem, each user is considered an agent of the system and the network represents their environment. Each user can take certain actions, a_u, represented by changes in their weights $W_{u,\mu}$, described by $\mathbb{G}$ and the same greedy policy from the SCs learning is assumed. For each parameter, μ, each user can be in one of two states:

- State 1, if the user is not satisfied with respect to parameter μ (constraint (11e) is violated).
- State 2, if the user is satisfied with respect to parameter μ (constraint (11e) is satisfied). [3]

Hence, the states that a user can be, υ_u, are represented by

$$\upsilon_u = \begin{cases} 1, & \text{if } E'_{u,\mu} < E_{u,\mu}\,, \\ 2, & \text{if } E'_{u,\mu} \geq E_{u,\mu}. \end{cases} \tag{14}$$

In terms of reward, r_{υ_u,a_u}, for both states the reward is given as the relative difference between what was achieved and what was requested. The reward is defined as

$$r_{\upsilon_u,a_u} = \frac{E'_{u,\mu} - E_{u,\mu}}{E_{u,\mu}}, \tag{15}$$

and represent how far away each user is from being satisfied or how much a user is over satisfied.

For each state-action pair and reward, each agent updates its μ Q-Tables independently for each parameter, resulting in a change of weights for each user. In this case, each Q-Table is represented by a matrix $[a_u \times \upsilon_u]$ and for each time-step of the algorithm, they are updated via (1). Lastly, similar to the SC learning, the same two stopping criteria were devised for the user weights learning, in which the algorithm stops either after a fixed number of maximum iterations (M_{uw}) or after its reward did not improve more than a threshold from one iteration to the other.

[3]Since a lower value of latency is preferred, (14) will have its signal changed when the latency parameter is considered.

C. Proposed Algorithm

Based on the system model and learning phases, an iterative algorithm for the proposed solution can be elaborated, in which the optimization of SCs CREOs and users weights is performed. The proposed solution is distributed, in which SCs update their CREOs independently from other cells and users also update their weights independently from one another. Furthermore, the algorithm is composed of two different parts, the first, *SC learning*, deployed in every SC of the network, performs an optimization of CREOs. The second, *user weights learning*, is deployed in all users devices, and optimizes user weights in order to achieve user-specific cell association.

In terms of the network optimization, each algorithm in every SC needs to have certain parameters initialized, such as backhaul characteristics (load and η parameters, mainly: capacity, latency and resiliency). In addition, SCs CREOs, Q-Tables, and the number of users connected to it are all initialized as zero when the cells are turned on. In terms of users requirements, they could be initialized by different applications, such as whenever an audio/video stream application is open, a higher preference for high throughput and low latency could be requested.

It is envisioned that the SC learning takes place whenever the network detects that its performance is below a threshold, for example, if the total network throughput is below a certain value. As such, whenever this conditions is triggered, each SC learns the best CREOs that satisfy the majority of the users connected to them. These offsets depend not only on the state the cell is currently in, and environment conditions, such as shadowing, backhaul load and the number of available RBs, but also on user's requirements and weights. Based on that, each cell selects the best action, according to what it knows, for each time-slot. In addition, due to the way the problem is formulated and the way that the states are given by the constraints defined in (11a), the SCs will always be in only one of the possible three states. Hence, for that time-slot, depending on the current SCs states, they will try to find the optimal CREO that maximizes the system total reward, as given by the RL formulation. In other words, the RL optimization problem can be seen as a system that tries to maximize its total reward, by dividing its goal (total cumulative reward) into smaller micro goals (maximize the reward of each iteration). As such, in every iteration, independently of the state a SC is, it will always try to find the best solution for that time slot. In addition, since there is a certain correlation between successive time slots in the network, the SCs keep their Q-Tables between time slots, in order to utilize previous gathered knowledge in order to find better actions in the future.

At the end of this stage, the new CREOs are communicated to the users via the control channel. For the user learning, it is planned that users can change their weights whenever their perceived QoS is below a target. This can happen due to several reasons, such as changes in SCs CREOs, network

failures or outages, or network congestion. If a user triggers its learning, the best weights that are assigned to each of its μ parameters are going to be learned. Similarly to the learning of SCs, each user evaluates the best actions that it can take based on its current state for that time-slot, which depends on parameters such as the RSRP, the user's position, and the SCs' CREOs. Then, for each parameter, the users choose the best available weight, while keeping μ Q-Tables between time-slots. Similarly to the SCs scenario, the Q-Learning of user weights can also be seen as each agent trying to maximize their total cumulative reward (being satisfied with respect to each parameter), but by dividing it into iterations, instead of each agent trying to maximize just one global goal, smaller goals at every iteration are pursued. After the user update its weights, the network association process, according to (4) is performed in order to decide if the user stays in the same cell or is handed over to a better more fitting cell.

Because of this iterative process, it is inevitable that ping-pongs occur in the network. However, due to the way the system is modeled, ping-pongs can only occur whenever a cell does not have enough resources to accommodate a user or if the channel conditions between a user and that cell are not good enough, resulting in a poor SINR. However, if any of these conditions are true, the user should be reallocated to a better cell anyway, independently of the proposed algorithm. In addition, because the proposed algorithm only occurs whenever certain thresholds are met, meaning that the network is not operating at its optimal point or that users are not satisfied, users should also attempt to connect to another cell, resulting in no number of increased connections.

Furthermore, it is envisioned that the weights learned by each individual user are kept in his device and can depend on the type of application being utilized. As such, the proposed solution presents no issues regarding the utilization of different applications. On the one hand, regarding mobility management, the proposed algorithm presents the same issues as current solutions for heterogeneous networks, in which user devices in idle mode are continuously ranking potential serving cells. On the other hand, the proposed framework is more robust and can adapt to changes faster, as user devices have the advantage of performing a use-centric selection based on learned weights, while also utilizing previous historical data and gathered knowledge.

Algorithms 1 and 2 show an implementation of the SCs and users learning, respectively, while Fig. 2 shows a diagram of the overall proposed solution. In the diagram it can be seen that both users and SC keep monitoring their performance in order to decide when to trigger the proper algorithm. The diagram shows that user 1 (UE1), in active mode, keeps monitoring the network at certain time instants (which can be defined according to application, for example) and when it detects that the performance is below a threshold it triggers Algorithm 2, updating its weights. After that UE1 then changes SC and re-evaluates the network, determining that its condition is back to the desired level. It can also be seen that the SC monitors the network performance and whenever the network conditions are below a threshold it triggers Algorithm 1 resulting in a change of CREOs. These new CREOs are then broadcast to all users, independently if they are in idle or active mode and also of Algorithm 2. Lastly, the diagram also shows what happens if a user is in idle mode (UE2). In this case, when UE2 joins the network, it first performs an initial cell selection, to determine which to camp on and then, after new CREOs are received, it re-evaluates the cell selection procedure to determine if it will remain or handover to a new SC. When UE2 is in idle mode it only needs to reselect cells when new CREOs are broadcast and no user weights optimization is performed in this stage. Only after UE2 has moved from idle to active mode that it starts monitoring the network and performing Algorithm 2, if necessary.

Algorithm 1 Small Cells Q-Learning

inputs: backhaul conditions, cell load, $\mathbb{E}$, $\mathbb{W}$
output: $\mathbb{O}$

```
for all small cells do
    for each parameter η do
        for all iterations do
            Measure λ_c and θ_μ
            Determine SC current state using (12), υ_c
            Choose action: select new CREO value, O_{c,η}
            Determine reward using (13), r_{υ_c,a_c}
            Perform action: change SC CREO value
            Measure new λ_c and θ_μ
            Update SC state
            Update Q-Tables according to (1)
            if Stopping Criteria is met then
                Stop
            end
        end
    end
end
Return 𝕆
```

Algorithm 2 User Weights Q-Learning

inputs: RSRP, $\mathbb{E}$, $\mathbb{E}'$, $\mathbb{O}$
output: $\mathbb{W}$

```
for all users do
    for each parameter μ do
        for all iterations do
            Measure user (dis)satisfaction
            Determine current user state using (14), υ_u
            Choose action: select new weight, W_{u,μ}
            Determine reward using (15), r_{υ_u,a_u}
            Perform action: change user weight
            Measure new user (dis)satisfaction
            Update user state
            Update Q-Tables according to (1)
            if Stopping Criteria is met then
                Stop
            end
        end
    end
end
Return 𝕎
```

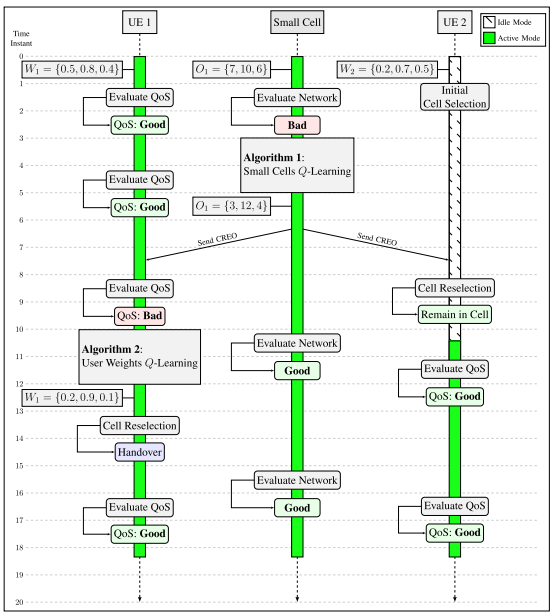


Fig. 2. Diagram showing how the proposed solution can work. In this diagram, only 2 users are shown for convenience, but it is assumed that more users are connected to the SC. Both users and SC monitor the network and change their parameters if the performance is below a threshold.

D. Complexity, Convergence and Overhead Analysis

The proposed solution is analyzed in terms of computational complexity, convergence and signaling complexity to highlight its feasibility and practical implementation.

1) Complexity Analysis: it is shown in [27] that for the worst case scenario, the Q-Learning complexity scales linearly with the number of states and actions, assuming a computational complexity of $O(s \cdot a)$, where s denotes the total number of possible states, and a represents the total number of possible actions. For the proposed algorithm, since the η CREOs and μ weights optimizations can run in parallel, the increase in complexity for each SC is given by $O(\upsilon_c \cdot a_c)$, and each user would also require an extra computation of $O(\upsilon_u \cdot a_u)$.

When compared to the fixed CREOs solutions, the proposed solution is slightly more complex, however, this extra complexity at both the network and user sides translate to extra QoS gains. When compared to the BS-centric solution, the proposed method adds only an additional level of complexity at the user side, but as the results show, this increase in complexity is traded-off by gains in user satisfaction.

2) Convergence Analysis: Q-Learning has already been shown to converge independently of the policy chosen in [18] and [24]. As previously mentioned, the RL optimization problem can be seen as a system that tries to maximize its total reward by dividing the problem into smaller micro goals. Hence, from a convergence perspective, it can be said that the algorithm converges at every episode (network snapshot), while also maintaining its Q-Tables in between episodes. In other words, the proposed solution attempts to find, for the current network configuration, the best CREO and weights settings. In addition, although the network changes in between episodes, there is a quite strong correlation between successive time instants, hence the algorithm is able to take future actions based on previous knowledge and maximize its total reward.

3) Overhead Analysis: the proposed scheme can be implemented in current LTE networks with minimal modifications to the standards. One possible modification may be a small change in the current CREO settings supported by LTE, in which the optimized CREO values of each cell are broadcast using different frequencies and a cell identifier [28]. In this case the frequency of broadcasting the optimized CREO values would remain unaltered, whereas the frequency in which UEs in idle mode access this information may be changed as it depends on specific implementation. It may also be beneficial that users in idle mode change the frequency in which they access the CREO information, although it is not necessary. As such, the proposed changes would be to associate multiple offsets with every neighboring cell, requiring only $n \cdot \eta \cdot b$ extra overhead, where n represents the number of SCs, η is the number of extra parameters, and b is the number of bits currently used for one offset. One possible alternative to deal with this is to design a system in which the CREOs are broadcast one after the other repetitively in such a way that users are signaled the number of offsets to expect and how often they should get an update. If that is the case, then no additional signaling for broadcasting the CREOs is required.

Another source of overhead increase is the need to continuously inform all neighbors of all dynamically optimized CREOs (over the X2 or the S1 interface). This additional overhead has the same cost as before, as $n \cdot \eta \cdot b$. However, despite the increase in overhead, the user-specific scheme is advantageous from a signaling perspective when compared to a user-QoS or backhaul constraint agnostic association policy. The reason is that, in the latter, the probability of a user associating with an unsuitable cell is higher, leading to handovers being triggered to improve user QoS. Hence, by reducing the number of handovers in the network, providing a better user-cell association should be advantageous. Compared to current systems, the proposed scheme increases the signaling proportional to the number of cells, but reduces the overhead in proportion to the number of users, so the cumulative overhead is expected to reduce considerably. Lastly, the user weights optimization does not require that users send their weights to the network, as each user will perform its own optimization, and this optimization can be implementation specific, depending on vendors or applications, and does not require any changes in current standards.

V. Simulation Results

A. Simulation Scenario

In order to provide a proof of concept, an illustrative simulation scenario was set up in MATLAB. For this scenario, a single macro cell, with $m = 3$ sectors was considered, and, on top of each sector, $n = 7$ SCs were overlaid in a random manner. Each SC is considered to have one backhaul link, which can be of one of four types: optical fibre, mm Wave, microwave or copper wire. Each backhaul has $\eta = 3$ attributes that define its performance, as seen in Table I, in terms of: capacity, the total data rate that each backhaul is able to

TABLE I
BACKHAUL PARAMETERS [7], [17]

	Capacity (Mbps)	Latency (ms)	Resilience (%)
Fibre	500	1	99.999
mmWave	500	3	90
Microwave	100	5	99.999
Copper	50	10	99.999
Macro	∞	1	100

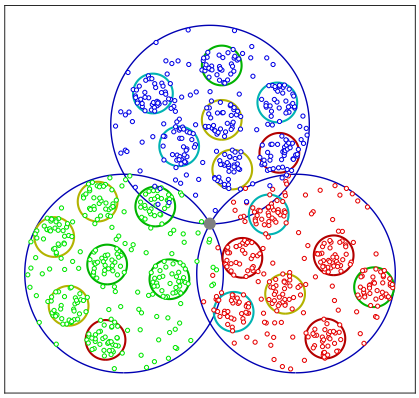


Fig. 3. Simulation scenario. The macro BS in the center (in gray) covers an $m = 3$ sectored area (dark blue circles). On each sector, $n = 7$ SCs, with different backhauls (represented by different colors), and $k = 315$ users are randomly distributed, with higher concentration near the SCs.

support; latency, the delay that users experience if connected to that link; [4] resilience, the reliability of the connection.

In each sector $k = 315$ users are distributed. A third of the users were distributed uniformly and randomly all over the sector, while the other two thirds were uniformly and randomly distributed near the SCs. It is also considered that each user has $\mu = 3$ requirements based on throughput, latency and resilience. In the simulated environment these requirements were generated randomly, assuming that users had an equal probability of requesting either a low or high value for each requirement, however, in a real situation, these could be dictated by the application. Figure 3 shows one possible configuration of the scenario, in which the macro cell, represented by the gray dot in the center, covers a three sectored area represented by the dark blue circles. On top of each sector, 7 SCs are randomly positioned, each with a different backhaul connection, and 315 users are overlaid. Table II shows the simulation parameters, which conform to 3GPP specifications as proposed in [29].

The system is ran for a total of ten independent runs, with different starting conditions, such as user requirements and positions, SC locations and backhaul links. At the beginning of each run the η and μ Q-Tables of SCs and users, respectively, are initialized to zero, but as previously mentioned, the corresponding matrices will be maintained in between episodes, being reset only after another episode begins. Also, other parameters such as channel conditions (fading and shadowing), backhaul loads, and user positions vary from one episode

[4]It is assumed that other latencies, such as queuing delay, or the delay caused by different ABS patterns can be dealt with other state-of-the-art algorithms, and that the backhaul latency is the minimum latency that can be achieved, bounded by the fixed link.

TABLE II
SIMULATION PARAMETERS [16], [25], [29]–[31]

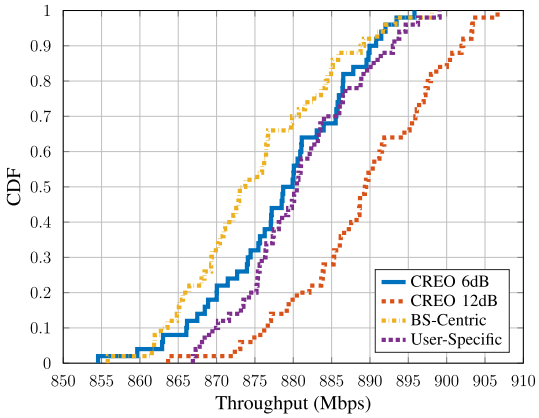
Parameters	**Value**
Number of Sectors (m)	3
SCs per Sector (n)	7
Users per Sector	315
Ratio of Users in SCs	2/3
Sector Radius	250 m
SC Radius	50 m
Macro BS EIRP	20 dBW
SC EIRP	7 dBW
Macro Cell Shadowing	4 dB
SC Shadowing	5 dB
Receiver Noise Figure	7 dB
Penetration Loss	18 dB
RBs per Cell (RB_{max})	50
Backhaul Overhead Factor (ρ)	1.3
Bandwidth of 1 RB (B)	180 kHz
Number of sub-carriers (N_{sc})	600
mmWave Outage	16%
ABS pattern (ζ_{ABS})	40%
Satisfaction Threshold (θ_μ)	0
Throughput req.† (low / high)	0.2 / 1 Mbps
Latency req.† (low / high)	5 / 10 ms
Resiliency req.† (low / high)	90 / 99.999%
Learning Rate (α_c, α_u)	0.5
Discount Factor (ϕ_c, ϕ_u)	0.9
Macro Cell Path Loss	$128.1 + 37.6 \cdot \log_{10}(d)$ dB
SC Path Loss	$140.7 + 36.7 \cdot \log_{10}(d)$ dB
CREO Values ($\mathbb{V}$)	$\{0, 1, \ldots, 12\}$ dB
Weights Values ($\mathbb{G}$)	$\{0, 0.1, \ldots, 1\}$ dB
Total number of episodes	50
Max. iterations ($M_{\text{uw}}/M_{\text{sc}}$)	30 / 50
Reward threshold (r_{th})	10%

†Requirement per RB.

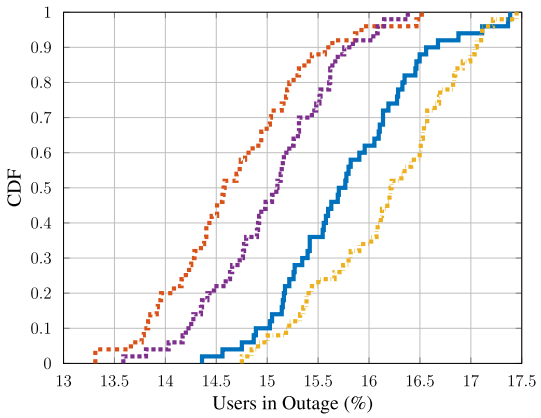
to another. In each run, a total of fifty episodes are performed and the metrics are computed and averaged out. In addition, for the first episode of the algorithm, an allocation process based only on the RSRP is done, so that a real network scenario with users already allocated to the cells of the system can be simulated. During the other episodes of the algorithm, the user-specific solution, based on Q-Learning, is evaluated. The computed metrics are then averaged out, in order to measure the performance of the system and evaluate the robustness of the proposed solution. Moreover, each episode is assumed to be one snapshot of the network, in which network conditions remain static and the SCs and users perform their optimization process over a certain amount of iterations (according to their stopping criteria). For example, in every episode it is assumed that channel and network conditions, such as RAN and backhaul, as well as user mobility remain the same. This is performed for the sake of simulation and in a real system, this optimization would be done in real time. Lastly, for the mm Wave backhaul an outage probability is assumed and it is evaluated in every iteration of the algorithm. When an outage occurs, users perceive a very low RSRP from that SC (e.g. -500dBm) and no connections to that cell are allowed in that iteration.

B. Performance Metrics

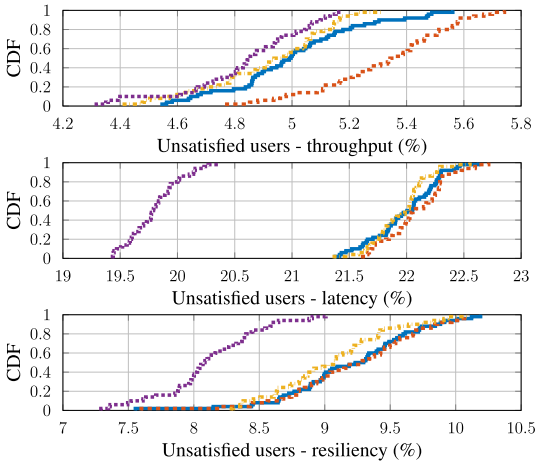
The proposed solution is compared to the BS-Centric approach [17] and both 6dB and 12dB fixed CREO.



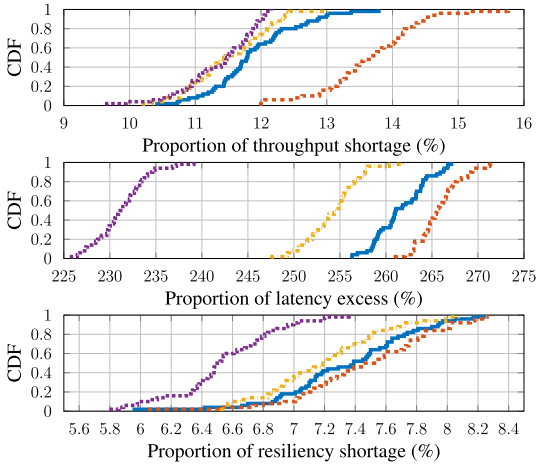
(a) Cumulative throughput.



(b) Total number of users in outage.



(c) Percentage of unsatisfied users.



(d) Dissatisfaction rate.

Fig. 4. System performance in terms of total network throughput (a), number of users in outage (b), percentage of unsatisfied users for each parameter (c) and dissatisfaction rates for each parameter (d).

The performance is measured in terms of four metrics: cumulative throughput; total number of users in outage; percentage of unsatisfied users for each parameter; dissatisfaction rate for each parameter. The dissatisfaction is defined as the percentage difference between what was requested and allocated, considering only unsatisfied users, as

$$D_\mu = \sum_{c=1}^{|\mathbb{C}|} \sum_{u=1}^{|U_c|} \frac{E_{u,\mu} - E'_{u,\mu}}{E_{u,\mu}}, \quad \forall u \in \mathbb{U} | E_{u,\mu} < E'_{u,\mu}, \tag{16}$$

where $\mu \in \{T, L, R\}$, T denotes throughput, L corresponds to latency, and R to resiliency. Lastly, the penalty incurred in throughput due to ABS is considered in the results.

C. Numerical Results

Figure 4a shows the results for the cumulative throughput of the network. As it can be seen, the cumulative throughput is largest when a fixed CREO is applied, performing better for a 12dB CREO. This works as expected, as by artificially increasing the range of SCs, more users are pushed to the SCs, achieving better reuse of the spectrum (more RBs being available). Hence, the 12dB CREO solution achieves the highest cumulative throughput. When comparing the BS-Centric and user-specific solutions, it can be seen that their performance approach the fixed 6dB CREO, with the user-specific solution slightly outperforming both approaches. This also works as expected, as in some cases it is better to apply large CREOs attracting more users to certain SCs, while in others is best to apply smaller CREOs, making users associate with the macro BS more often. Furthermore, because the reward of the intelligent solutions (BS-Centric and user-specific) is not only composed of the cumulative throughput but also of the other QoS metrics, it is natural that a trade-off between these metrics is achieved.

Figure 4b presents the total percentage of users in outage for each solution. Also as expected, the 12dB CREO is able to minimize the number of users in outage, as it is able to attract more users, due to the larger artificially extended coverage area. In addition, it can be seen that the user-specific solution lies in between the fixed 6dB and 12dB CREO approaches and that the BS-Centric approach has the worst performance of all. This highlights the gains of the proposed approach, in which tuning user side parameters enables more users to be covered rather than just tuning the CREOs of SCs.

Regarding users satisfaction, Fig. 4c illustrates the percentage of unsatisfied users in the network with respect to each parameter. It can be seen that fixed CREO solutions do not perform as well as the intelligent solutions, both BS-centric

TABLE III
CONTRIBUTION OF DIFFERENT USERS TO TOTAL NUMBER OF UNSATISFIED USERS AND DISSATISFACTION RATES

		Contribution in (%) to total amount of unsatisfied users			**Contribution in (%) to total dissatisfaction**			**Average of associated users**
		Throughput	Latency	Resiliency	Throughput	Latency	Resiliency	-
CREO 6dB	Macro Cell	17.24	0	0	7.00	0	0	150
	Small Cell	48.48	89.19	89.62	86.04	98.58	98.77	582.80
	CREO Region	34.28	10.81	10.38	6.96	1.41	1.23	62.58
CREO 12dB	Macro Cell	9.98	0	0	4.77	0	0	133.61
	Small Cell	32.25	84.70	84.76	63.51	96.55	96.71	566.07
	CREO Region	57.77	15.30	15.24	31.72	3.45	3.29	106.58
BS-Centric	Macro Cell	19.95	0	0	8.39	0	0	150
	Small Cell	48.58	89.55	90.98	84.22	98.46	98.91	579.06
	CREO Region	31.47	10.45	9.02	7.38	1.54	1.09	61.84
User-Specific	Macro Cell	18.05	0	0	6.28	0	0	150
	Small Cell	55.92	90.40	91.02	89.46	98.80	99.05	593.28
	CREO Region	26.03	9.60	8.98	4.25	1.20	0.95	58.50

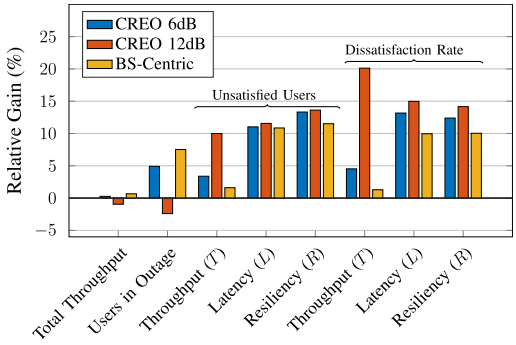


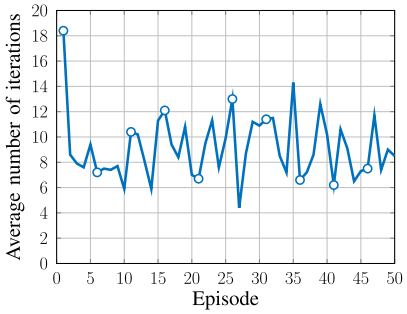
Fig. 5. Relative gain of the proposed algorithm. The proposed solution outperforms other solutions in all metrics, with the exception of the number of users in outage and cumulative throughput when compared to the fixed 12dB CREO approach.

and user-specific. Furthermore, it can be seen that tuning only CREOs of SCs can achieve a better global performance than fixed solutions, but by tuning both CREOs and user weights this optimization can be enhanced. This can be explained by the fact that when both CREOs and weights are considered, together with the proposed constraints, the system tends to deliver what the users have requested, minimizing network resource wastage. This enables more users to be allocated to that SC backhaul, provided that it has enough radio resources available. It can also be seen that tuning both CREOs and weights achieves a better performance with respect to all parameters. Figure 4d shows the total proportion of dissatisfaction of users regarding each parameter, which are obtained according to (16). As it can be seen, the BS-Centric solution slightly outperform the fixed approaches in all metrics, however in the case of the proposed user-specific approach, the dissatisfaction with respect to all parameters can be mitigated even further.

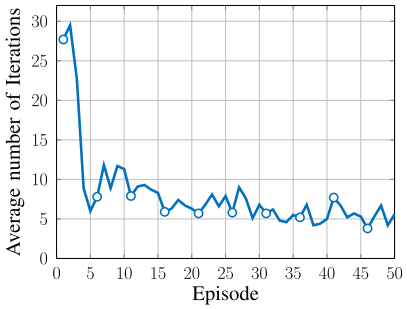
Figure 5 shows the relative gain of the user-specific solution with respect to other methods. As it can be seen, by optimizing both network and user parameters, the proposed solution is able to reduce the number of unsatisfied users and their dissatisfaction rates by around 10%. Furthermore, when compared with the BS-Centric approach in terms of throughput, it can be seen that both solutions achieve a similar value, indicating that both approaches are able to find near-optimal values for this metric. As it can be seen, Figure 5 emphasizes that the proposed method is able to better allocate the backhaul resources, reducing the number of unsatisfied users as well as their dissatisfaction rates. By delivering for each user only what is requested, $E'_{u,\mu} \rightarrow E_{u,\mu}$, the amount of resources allocated to over satisfied users is reduced, freeing backhaul resources and reducing the number of unsatisfied users and their dissatisfaction rates. However, this comes at a slightly expense in terms of cumulative throughput and number of users in outage (when compared to the fixed 12dB CREO solution).

Lastly, Table III shows how users associated to the macro cell and SCs in and out of the CREO regions contribute to the total of unsatisfied users and dissatisfaction rates. As it can be seen, the proposed solution is able to achieve the minimum dissatisfaction amongst CREO users, at the expense of a higher dissatisfaction rate of users connected to the SCs. Also, the user-specific solution associates the second most amount of users to SCs (when accounting both SC and CREO regions), only behind the 12dB approach. However, the user-specific solution associates more users to the macro cell than the 12dB solution. This highlights the objective of the proposed solution, in which depending on the combination of CREOs, user weights and requirements, users are redirected to the most fitting cell, minimizing network resource wastage.

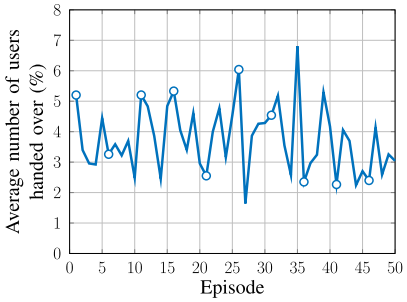
Regarding the algorithm convergence, Figs. 6a and 6b show the average number of iterations of SCs learning and user learning per episode, respectively. As it can be seen and as expected, in the beginning, as both algorithms do not know enough about the environment, they start by performing plenty iterations in order to find the optimal network and user settings. However, as the number of episode increases, this number decreases and both solutions converge to around 10 and 5 iterations in case of SC learning and user weights learning, respectively. Moreover, it can be seen that the optimization of user weights is more stable because they operate after the SCs have optimized their CREOs. On the other hand, the optimization of CREOs is slightly more unstable, although it still converges, due to network changes and user mobility,



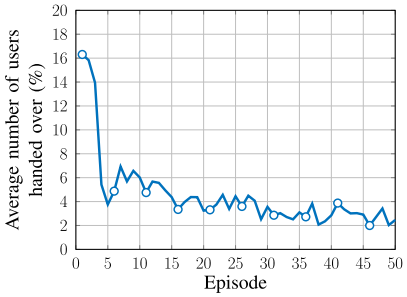
(a) Average number of iterations (SC Learning) per episode.



(b) Average number of iterations (User Weights Learning) per episode.



(c) Average number of users handed over, in percentage, per episode (SC Learning).



(d) Average number of users handed over, in percentage, per episode (User Weights Learning).

Fig. 6. Analysis of the convergence properties of the proposed algorithm.

which varies from one episode to the other. It can also be seen that the proposed solution converges rather fast, as both algorithms converge after around 5 episodes.

Figures 6c and 6d show the average number of users handed over per episode after performing each algorithm (in percentage). As it can be seen, when SC learning is performed an average and constant number of 4% of total users is reallocated every time, while when users learn their weights, this number starts relatively high at around 16% and then converges, after around 20 episodes, to around 3%. This not only shows the convergence of the proposed methods, but also further emphasizes that by only tuning CREOs a constant rate of users is handed over to SCs, while by tuning both CREOs and user weights the algorithm can learn which users to handover and only change the association of the users that it needs to.

VI. Conclusions

In order to achieve the requirements of future cellular networks, such as the ever increasing user demands and also to enable a wide range of applications, it is clear that intelligent and robust solutions need to be deployed. With that in mind, new paradigms of user-cell association need to be considered, in which the end-to-end connectivity is contemplated, instead of current radio interface based solutions. In addition, solutions must also optimize not only parameters of the network, but also user parameters, to achieve user-specific cell association.

In this paper, a RL approach, in which both SC CREOs and user weights were optimized using Q-Learning was proposed. Results show that the proposed method outperforms fixed CREO solutions and another BS-centric approach. Results also demonstrate the importance of tuning both network and user side parameters, as this enables the proposed algorithm to allocate only enough for each user in order for it to be satisfied, while also allowing more backhaul resources to be shared among other users. Thus, by optimizing both network and user parameters a reduction of around 10% in the total number of unsatisfied users could be achieved. One possible extension of this work could be the investigation of a similar scenario, but considering SCs with multiple backhauls with different characteristics. The system could then learn either to choose the best backhaul for each situation or to connect different users to different types of backhauls.

References

[1] G. P. Fettweis, "A 5G wireless communications vision," *Microw. J.*, vol. 55, no. 12, pp. 24–36, Dec. 2012.

[2] J. G. Andrews *et al.*, "What will 5G be," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1065–1082, Jun. 2014.

[3] 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 Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2392–2431, 4th Quart. 2017.

[4] S. Chia, M. Gasparroni, and P. Brick, "The next challenge for cellular networks: Backhaul," *IEEE Microw. Mag.*, vol. 10, no. 5, pp. 54–66, Aug. 2009.

[5] H. Galeana-Zapién and R. Ferrus, "Design and evaluation of a Backhaul-aware base station assignment algorithm for OFDMA-based cellular networks," *IEEE Trans. Wireless Commun.*, vol. 9, no. 10, pp. 3226–3237, Oct. 2010.

[6] J. J. Olmos, R. Ferrus, and H. Galeana-Zapien, "Analytical modeling and performance evaluation of cell selection algorithms for mobile networks with Backhaul capacity constraints," *IEEE Trans. Wireless Commun.*, vol. 12, no. 12, pp. 6011–6023, Dec. 2013.

[7] M. Jaber, M. A. Imran, R. Tafazolli, and A. Tukmanov, "A distributed son-based user-centric Backhaul provisioning scheme," *IEEE Access*, vol. 4, pp. 2314–2330, 2016.
[8] C. Ran, S. Wang, and C. Wang, "Balancing backhaul load in heterogeneous cloud radio access networks," *IEEE Wireless Commun.*, vol. 22, no. 3, pp. 42–48, Jun. 2015.
[9] H. Elshaer, F. Boccardi, M. Dohler, and R. Irmer, "Load Backhaul aware decoupled downlink/uplink access in 5G systems," in *Proc. IEEE Int. Conf. Commun. ICC*, Jun. 2015, pp. 5380–5385.
[10] A. D. Domenico, V. Savin, and D. Ktenas, "A Backhaul-aware cell selection algorithm for heterogeneous cellular networks," in *Proc. IEEE 24th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2013, pp. 1688–1693.
[11] F. Pantisano, M. Bennis, W. Saad, and M. Debbah, "Cache-aware user association in Backhaul-constrained small cell networks," in *Proc. 12th Int. Symp. Modeling Optim. Mobile, Ad Hoc, Wireless Netw. (WiOpt)*, May 2014, pp. 37–42.
[12] Q. Han, B. Yang, G. Miao, C. Chen, X. Wang, and X. Guan, "Backhaul-aware user association and resource allocation for energy-constrained HetNets," *IEEE Trans. Veh. Technol.*, vol. 66, no. 1, pp. 580–593, Jan. 2017.
[13] M. Feng, S. Mao, and T. Jiang, "Joint frame design, resource allocation and user association for massive MIMO heterogeneous networks with wireless Backhaul," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1937–1950, Mar. 2018.
[14] H. Ma, H. Zhang, X. Wang, and J. Cheng, "Backhaul-aware user association and resource allocation for massive MIMO-enabled HetNets," *IEEE Commun. Lett.*, vol. 21, no. 12, pp. 2710–2713, Dec. 2017.
[15] Y. L. Lee, T. C. Chuah, A. A. El-Saleh, and J. Loo, "User association for Backhaul load balancing with quality of service provisioning for heterogeneous networks," *IEEE Commun. Lett.*, vol. 22, no. 11, pp. 2338–2341, Nov. 2018.
[16] M. Jaber, M. Imran, R. Tafazolli, and A. Tukmanov, "An adaptive Backhaul-aware cell range extension approach," in *Proc. IEEE Int. Conf. Commun. Workshop (ICCW)*, Jun. 2015, pp. 74–79.
[17] M. Jaber, M. A. Imran, R. Tafazolli, and A. Tukmanov, "A multiple attribute user-centric Backhaul provisioning scheme using distributed SON," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2016, pp. 1–6.
[18] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, vol. 1. Cambridge, MA, USA: MIT Press, 1998.
[19] G. Zhang, T. Q. Quek, M. Kountouris, A. Huang, and H. Shan, "Fundamentals of heterogeneous Backhaul design—Analysis and optimization," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 876–889, Feb. 2016.
[20] N. Bhushan *et al.*, "Network densification: The dominant theme for wireless evolution into 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 82–89, Feb. 2014.
[21] P. Wang, W. Song, D. Niyato, and Y. Xiao, "QoS-aware cell association in 5G heterogeneous networks with massive MIMO," *IEEE Netw.*, vol. 29, no. 6, pp. 76–82, Nov./Dec. 2015.
[22] "Heterogeneous network deployments in LTE—The soft-cell approach," Ericsson, Stockholm, Sweden, White Paper, Dec. 2011. [Online]. Available: https://pdfs.semanticscholar.org/b252/4d4a278fec094d52e626cf1e38393bf6c431.pdf
[23] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Intell. Res.*, vol. 4, pp. 237–285, May 1996.
[24] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, May 1992. doi: 10.1007/BF00992698.
[25] E. Almeida *et al.*, "Enabling LTE/WiFi coexistence by LTE blank subframe allocation," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2013, pp. 5083–5088.
[26] T. S. Rappaport *et al.*, *Wireless Communications: Principles and Practice*, vol. 2. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.
[27] S. Koenig and R. G. Simmons, "Complexity analysis of real-time reinforcement learning applied to finding shortest paths in deterministic domains," School Comput. Sci., Carnegie-Mellon Univ. Pittsburgh, PA, USA, Tech. Rep. CMU-CS-93-106, 1992.
[28] *On Range Extension in Open-Access Heterogeneous Networks*, document, R1-103181, 3GPP TSG RAN WG1 Meeting, Montreal, QC, Canada, May 2010, vol. 61.
[29] *Technical Specification Group Radio Access Network, Further Advancements for E-UTRA Physical Layer Aspects (Release 9)*, document TR 36.814, 3rd Generation Partnership Project, Mar. 2017, vol. 9.2.0.
[30] *R1-103264: Performance of eICIC with Control Channel Coverage Limitation*, document 3GPP TSG RAN WG1 Meeting, NTT DoCoMo, Montreal, QC, Canada, May 2010, vol. 61.
[31] N. Alliance, "Small cell backhaul requirements," Next Gener. Mobile Netw. (NGMN) Alliance, Frankfurt, Germany, White Paper, Jun. 2012. [Online]. Available: https://www.ngmn.org/fileadmin/user_upload/NGMN_Whitepaper_Small_Cell_Backhaul_Requirements.pdf

Paulo Valente Klaine (S'17) received the B.Eng. degree in electrical and electronic engineering from the Federal University of Technology–Paraná (UTFPR), Brazil, in 2014, and the M.Sc. degree (Hons.) in mobile communications systems from the University of Surrey, Guildford, U.K., in 2015. He is currently pursuing the Ph.D. degree with the School of Engineering, University of Glasgow. In 2016, he spent the first year of his Ph.D. working in the 5G Innovation Centre (5GIC), University of Surrey. His main interests include self-organizing cellular networks and the application of machine learning in cellular networks.

Mona Jaber received the B.E. degree in computer and communications engineering and the M.E. degree in electrical and computer engineering from the American University of Beirut, Beirut, Lebanon, in 1996 and 2014, respectively, and the Ph.D. degree from the 5G Innovation Centre, University of Surrey, in 2017. Her Ph.D. research was on 5G backhaul innovations. She was a telecommunication consultant in various international firms with a focus on the radio design of cellular networks, including GSM, GPRS, UMTS, and HSPA. She has been leading the IoT Research Group, Fujitsu Laboratories on Europe, since 2017, where she focuses in particular on automotive applications. Her research interests include cyber-physical systems, data-driven digital twins, and AI/ML applications in the automotive industry.

Richard Demo Souza (S'01–M'04–SM'12) was born in Florianópolis, Brazil. He received the B.Sc. and D.Sc. degrees in electrical engineering from the Federal University of Santa Catarina (UFSC), Brazil, in 1999 and 2003, respectively. In 2003, he was a Visiting Researcher with the Department of Electrical and Computer Engineering, University of Delaware, USA. From 2004 to 2016, he was with the Federal University of Technology–Paraná (UTFPR), Brazil. Since 2017, he has been with the Federal University of Santa Catarina (UFSC), Brazil, where he is currently an Associate Professor. His research interests are in the areas of wireless communications and signal processing. He is a Senior Member of the Brazilian Telecommunications Society (SBrT). He was a co-recipient of the 2014 IEEE/IFIP Wireless Days Conference Best Paper Award, the Supervisor of the awarded Best Ph.D. Thesis in electrical engineering in Brazil in 2014, and the 2016 Research Award from the Cuban Academy of Sciences. He has served as an Editor-in-Chief of the SBrT *Journal of Communication and Information Systems* and an Associate Editor for the IEEE COMMUNICATIONS LETTERS, the *EURASIP Journal on Wireless Communications and Networking*, and the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY.

Muhammad Ali Imran (M'03–SM'12) has been a consultant to international projects and local companies in the area of self-organized networks. He is currently a Professor of wireless communication systems. He is also the Head of the Communications, Sensing and Imaging CSI Research Group, University of Glasgow. He is also an Affiliate Professor with The University of Oklahoma, USA, and also a Visiting Professor with the 5G Innovation Centre, University of Surrey, U.K. He has more than 18 years of combined academic and industry experience with several leading roles in multi-million pounds funded projects. He has authored or coauthored more than 400 journals and conference publications and holds 15 patents. His research interests are in self organized networks, wireless networked control systems, and wireless sensor systems. He was a fellow of IET and a Senior Fellow of HEA. He was an editor of two books and authored more than 15 book chapters, has successfully supervised more than 40 postgraduate students at Doctoral level.