Embracing Complexity: Agent-Based Modeling for HetNets Design and Optimization via Concurrent Reinforcement Learning Algorithms

Mostafa Ibrahim, *Student Member, IEEE*, Umair Sajid Hashmi, *Member, IEEE*, Muhammad Nabeel, Ali Imran, *Senior Member, IEEE*, and Sabit Ekin, *Senior Member, IEEE*

***Abstract*—Complexity is an inherent property in wireless heterogeneous networks (HetNets). In this paper, we investigate the application of the agent-based modeling (ABM) tool for optimization of complex and dynamic HetNets. The proposed framework contains a diversity of game-theoretic, machine learning, and rule-based algorithms within the same model. We present and analyze a HetNet ABM model that runs parallel reinforcement learning (RL) algorithms for spectrum deployment, interference management, resource allocation, and load balancing at both micro and macrocell levels. In our proposed model, two RL-based algorithms work jointly to manage the co-tier and cross-tier interferences. The macrocell runs the first algorithm to control the transmission power of the small cells. The second RL algorithm is run by small cells to assign the users to the sub-bands with less interference levels. Simultaneously, the user association is decided by the users depending on the available resources at the cells and user preferences. The model is then evaluated under various network load conditions to deduce relationships between the cell loads, aggregate bit rate, latency, and user association. Moreover, the system is assessed in a dynamic network scenario with moving users and is confirmed to possess the ability to attain convergence with sufficient performance levels.**

***Index Terms*—HetNets, complexity, agent-based-modeling, multi-agent-systems, 5G and beyond.**

I. Introduction

HETEROGENEOUS networks (HetNets) and small cell densification is a pillar technology in 5G and Beyond telecommunication systems. Cell densification aims to continuously improve key network parameters, such as network coverage, capacity, latency, and load distribution. Several technical challenges are in the way of the deployment of small cell networks. The interference management and self-organization in HetNets are the two of the main technical challenges discussed in [1], [2]. Further, the small cell network capabilities of self-organization, self-configuration, and self-analysis affect the system efficiency.

There are several parameters to study and trade-offs to resolve when optimizing a HetNet [3]. The main goal is to cross optimize HetNet parameters and functions; resource allocation [4], [5], interference management [6], [7], [8], latency [9], user association [10] and cell load balancing [11], mobility and handovers [12], energy efficiency [13], [14], costs of deployment [15], and coexistence with other radio access technologies [16]. In such a high dimensional design space these key performance indicators (KPIs) compete with each other, making it difficult to satisfy the peak values for all of them simultaneously [17]. Moreover, in a realistic scenario, different nodes (users or cells) have diverse priorities and goals upon which the optimal solution is defined. Therefore, a suitable modeling paradigm is required within which the problem can be entirely formalized to yield an optimal operating solution.

There are several modeling paradigms for such complex scenarios [18]. One of the main modeling paradigms is the game-theoretic framework that studies strategies and interactions among players who behave rationally towards maximizing their benefits [19]. A game-theoretic analysis can capture the Nash equilibrium conditions and states, but it is limited in showing the system's dynamic behavior. Although game theory is a powerful tool, its application in HetNets faces some challenges, whether in cooperative [20] or non-cooperative schemes [21]. For example, the assumption of purely rational agents is not always reflected in practical networks. Then, when it comes to the utility functions, there is the modeling challenge of how a node assigns values for performance levels and how that would affect the validity and efficiency of the Nash equilibrium. Moreover, wireless networks are both complex and random, which leads to complex nonlinear mathematical analysis.

A widely used machine learning-based paradigm is Multi-agent Reinforcement Learning (RL). This model depends on agents to study how players make decisions in their environment to maximize a utility function [22]. RL is a powerful tool

Manuscript received April 12, 2021; revised August 16, 2021; accepted October 12, 2021. Date of publication October 19, 2021; date of current version December 9, 2021. This work was supported by the National Science Foundation under Grants 1923295 and 1923669. The associate editor coordinating the review of this article and approving it for publication was H. Lutfiyya. *(Corresponding authors: Mostafa Ibrahim; Sabit Ekin.)*

Mostafa Ibrahim and Sabit Ekin are with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078 USA (e-mail: mostafa.ibrahim@okstate.edu; sabit.ekin@okstate.edu).

Umair Sajid Hashmi is with the School of Electrical Engineering and Computer Science, National University of Sciences and Technology, Islamabad 44000, Pakistan (e-mail: umair.hashmi@seecs.edu.pk).

Muhammad Nabeel and Ali Imran are with the AI4Networks Research Center, School of Electrical and Computer Engineering, University of Oklahoma, Tulsa, OK 74135 USA (e-mail: muhmd.nabeel@ou.edu; ali.imran@ou.edu).

Digital Object Identifier 10.1109/TNSM.2021.3121282

that can reach policies and strategies that are beyond simple human decision making strategies [23]. In this paradigm, the agents utilize a comprehensive state-action reward table (or Q-table) that is a mapping of actions with expected rewards. Based on this, the agent chooses one of the available actions given to it. Consequently, it moves to a new state with a different reward. The goal of the agent eventually is to gather as much cumulative reward as possible. There are some challenges that face a multi-agent RL system [24], [25]. The relevant ones are the high-dimensional state and action space in 5G and beyond HetNets [26], which introduces non-practical computational complexity and high learning duration. Another challenge is the proper choice of reward functions, especially when we have different types of agents.

Rule-based modeling differs from the previous paradigms in terms of intelligence and rationale of the agents [27]. The players are not required to maximize some utility function or to follow a learning algorithm. Instead, the players follow a set of well-defined rules by the system designers. The system modelers' focus is to choose a proper set of rules that collectively reach optimality rather than designing a utility or reward function. This approach is practical when the modeled players do not have the computational capacity to act intelligently but lacks other benefits of learning-based approaches.

Our vision in this paper is to present a modeling paradigm that embraces the inherent complexity of heterogeneous networks. We develop a novel agent-based modeling (ABM) method to develop an extensive model where we can integrate a large number of HetNets parameters to investigate and study their complex interactions. We first propose the client-driven ABM-based model followed by a detailed discussion on its mechanism and dynamics. The developed model is then analyzed on multiple KPIs and compared with models from the literature. ABM is a method that models micro-scale interactions among a population of agents to study a complex system emergent behavior on a macro level [28]. ABMs are analyzed in simulation environments, and the players/agents follow rules that do not essentially refer to utility functions. Furthermore, different classes of agents can be defined with a diverse set of rules. Embracing the complexity concept allows us to build and study a more comprehensive model that mimics reality and gives more insight into the system. In the next subsection, we elaborate on the advantages offered by ABM in complex multi-objective optimization paradigms.

A. The Motivation Behind ABM

The study of complex adaptive systems and complexity theory is growing in many fields of science [29], [30], [31]. Realistic systems are non-linear and complex, therefore, complexity theory is applied to a wide range of applications. The growing studies are, for examples, in fields of economics [32], [33], cities management [34], social networks [35], transportation [36], [37], networks congestion [38], social sciences [39], and computation [40].

In [41], Mikulecky states that *"Complexity is the property of a real-world system that is manifest in the inability of any one formalism being adequate to capture all its properties"*. This principle applies quite adequately to the problem of modeling complex wireless HetNets, where base stations and users have a range of interdependent interactions and relationships. ABMs, unlike game theory, allow the designer to model several interacting games within the same model without having to construct an analytical framework. It also supports the testing of different player heuristics without assuming cognitive abilities. As a consequence, it can run real-world business simulations and analyze them across all network parameters. The ABM models incorporate:

- a group of agents,
- a set of rule-based actions that do not have to be rational-based,
- adaptive rules (optional), and
- an environment and a containing network.

There are differences between ABM and simulation-based learning paradigms. The ABM players' actions do not have to follow a reward function. Players can apply simple rules, like following their neighboring agents, which does not require any involved rationale. Moreover, ABMs are flexible to create different games within the same network and experiment. We can summarize the benefits of ABM for HetNets design and optimization in the following points.

1) It helps in building nonlinear complex systems with heterogeneous agents and complex interactions.
2) We can mix rule-based behaviors and machine learning (ML) based adaptive approaches for different types of agents within the same model. The relationship between ABM and complex adaptive systems is discussed in [42].
3) ABM transforms the awareness of the problem from merely solving an analytic optimization problem to studying the system dynamics, oscillations and instabilities, and interactions needed to reach an emergent behavior on the macro level.
4) It bridges the gap between theoretically reduced models and industry deployable models.

B. Contributions and Organization

We list below the salient contributions of our work in this paper.

1) We propose a novel ABM model with comprehensive rules of agent behaviors and interactions at both macro and small cell levels. The model aims at solving a complex HetNets problem of joint optimization of cell association, resource allocation, and interference management.
2) We create a dynamic client-driven paradigm where different agent based processes are defined for user terminals, small cells and macrocells in the architecture. Different processes are proposed at the UEs (user equipments) for real-time information collection and resource request decisions.
3) We model user association as a utility function that incorporates cell throughput and latency. ABM enables us to measure the service queuing in time and calculate load-induced queuing latency as a parameter. We incorporate users with different request rates and feedback

evaluation metrics via heterogeneous modeling of users in ABM.

4) We employ two concurrent reinforcement based algorithms for sub-band power management at the macrocell level and user assignment to sub-bands at the small cell level for efficient resource allocations. Multi-armed bandit problems are formulated for both cases which are tackled using Q-learning algorithms, which offer rewards on actions that increase aggregate signal-to-interference-and-noise ratio (SINR) for the network.
5) The proposed RL + rule-based mechanism is analyzed in terms of aggregate SINR, per cell throughput, inter-tier network load distribution and service latency. Simulation results show that the proposed solution improves the overall spectral efficiency of the network while allowing a better dynamic load balance between small cell and macrocell tier.

The paper's organization is as follows. In Section II, the related literature is reviewed. Then in Section III, the HetNet system model is presented. The proposed agent-based model is discussed in Section IV, followed by Section V which is dedicated to simulations and results. Finally, we conclude the paper in Section VI.

II. Related Work

This section reviews the literature of game-theoretic models and multi-agent reinforcement learning approaches first. Then we review the literature for the optimization methods cell association, resource allocation, and interference management.

A. Game-Theoretic Approaches

The most commonly used game-theoretic methods in HetNets can be classified as centralized or distributed schemes or cooperative or non-cooperative games [43]. In [4], a centralized scheme is proposed where the optimum downlink resource allocation is determined based on channel parameters. The authors investigated the trade-off between power consumption, transmission rate, and service quality. Inter-cell interference (ICI) management game theoretic approaches were presented in [6], [7], [13], [44], [45]. The work in [5] formalizes resource allocation as an auction-based algorithm with power-bandwidth constraints. In [46], a non-cooperative game is presented in which the players strive to minimize transmitted power while maintaining the required data rate. In order to ensure the convergence of the non-cooperative game, a virtual referee was suggested. A base station ON/OFF switching (sleeping) method was suggested in [47] as part of an energy-saving game called the satisfaction game.

B. Multi-Agent Reinforcement Learning Approaches

There have also been several studies on Multi-Agent Reinforcement Learning for HetNets. In [48], ICI is managed in a small cell architecture using a reinforcement learning process, in which a minimum SINR is given to macrocell users, and the SINR of small cells is maximized. Cells in [49] make resource assignment choices in order to improve SINR and QoS. A self-organization approach is paired with reinforcement learning in this study. In [50], macro- and picocells sense their environment and solve the problem of interference management and cell association through the usage of RL. In [51], through a decentralized Q-learning algorithm, small cells optimize their transmission power to maximize their capacity while keeping the interference at macrocell users' within reasonable limits. In [52], using multi-agent RL, downlink power and data rates are adapted. The study in [16] considers the coexistence of WiFi with cellular networks, with different QoS requirements for each HetNet user. In [15], an agent-based bargaining process is used to investigate the economic aspects of small cell deployment and spectrum leasing.

Next, we review the optimization processes for HetNets.

C. User Association and Resource Allocation

The conventional user association policy connects the users to the cell corresponding to the highest downlink received power [53]. Utility maximization association was also proposed in [54], [55], [56]. In the well-known *biased user association*, also called *cell range expansion* [57], cells are assigned bias factors, and users associate to the maximum received power weighted by the bias factor. For spectrum allocation, there are three schemes in literature; i) orthogonal deployment (OD), ii) co-channel deployment (CCD), and iii) partially shared deployment (PSD). In OD, the small cells' spectrum is orthogonal to the macrocells' spectrum, which is not a spectrally efficient deployment. In PSD, the macrocells share part of their spectrum with the small cells tier as in [58], [59]. In CCD, the whole spectrum is shared between the two tiers, which is more spectrally efficient but creates the problem of interference management. In [60], OD and CCD are evaluated with the assumption of conventional user association, and it shows that the CCD scheme improves the system throughput. Some studies aimed at jointly optimizing resource allocation and user association [61], [62], [63], [64].

D. Interference Management

As mentioned, sharing the spectrum between the macrocell tier and small cells tier introduces interference that should be managed. There are several studies and solutions in the literature on how to manage cross-tier and co-tier interference. The power of transmission at small cells is adjusted to reach specific performance levels without degrading the macrocell users' SINR levels. The following studies presented solutions for the cross-tier interference. In [48], two performance metrics were considered for small cells, the individual Shannon transmission rate, and the aggregate transmission rates of all the small cells. The study in [65] proposed a non-convex optimization method to maximize the aggregate throughput. Also, in [66] a distributed multi-agent learning approach is used to achieve maximum transmission link throughput. In [67], a distributed iterative power control scheme (uplink) is proposed, with dynamic pricing set by the interfered macrocells. The work in [68] offers a game-theoretic approach to jointly manage spectrum access, user scheduling, and power allocation to maximize the users' satisfaction.

For co-tier interference management, in [69], a distributed method is studied by managing the intra-cluster channel allocation. In [70], co-tier interference is managed through a coalition game between the small cells. In [71], inter-cell interference is mitigated by maximizing the available throughput ratio to the user's required data rate. In [72], an optimization problem is formulated for resource management where multiple QoS classes can be supported for different categories of users. In [73], [74], small cells use orthogonal resources initially, then later, they attempt to reuse the resources by coordinating with the neighboring small cells.

Since it is not straightforward to exploit state-of-the-art approaches in practical networks due to challenges related to complex non-linear mathematical analysis and infinitely high computation complexity with uncontrollable delay, in this work, we exploit the agent-based modeling method that not only reduces the complexity but is also practical and, hence, it is possible to incorporate in future wireless networks. Contrary to the existing literature, we propose a model that analyzes and optimizes the concurrent multi-agent processes, interference management, cell association resource allocation, spectrum deployment, and load balancing, all at the same time. Our model also runs comprehensive rules of agent behaviors and interactions for resource allocation, spectrum deployment, and load balancing at both micro and macrocell levels.

III. System Model

In this section, a distributed system model is proposed to study a complex, realistic HetNet, where the decision-making is partly assigned to the edges of the network (users and small cells). The two-tier network is composed of three agents; i) macrocells (MCs), ii) small cells (SCs), and iii) user equipment (UEs). The network is serving a group of UEs with different characteristics, i.e., having different applications for each UE. The spectrum is shared between macrocells and small cells and reused several times within the same macrocell to increase the network spectral efficiency. In the following subsections, a description of the system elements and assumptions are detailed.

A. Network Model

The modeled HetNet is a 2-tier network with macrocells forming the main network and small cells used as the second tier cells, as shown in Fig. 1. The MCs are distributed in a way that they have low overlapping areas, and the full network spectrum is reused orthogonally between them. The system is based on the LTE (long-term evolution) time-frequency resource block numerology. The spatial distribution of the UEs follows a stationary Poisson point process (SPPP) $\Phi_{UE} \in \mathbb{R}^2$ with intensity λ_{UE}. While for SCs, the spatial distribution is based on a repulsive point process, which maintains a minimum separation distance d_{min} between them. We consider the Matern hard core (MHC) type II [75] for the repulsive point process. The MHC point process is generated by dependent thinning [76] of a SPPP $\Phi_p \in \mathbb{R}^2$ with intensity λ_p. The MHC point process $\Phi_m \in \mathbb{R}^2$, will then have the intensity:

$$\lambda_m = \frac{1 - e^{-\lambda_p \pi d_{min}^2}}{\pi d_{min}^2}. \tag{1}$$

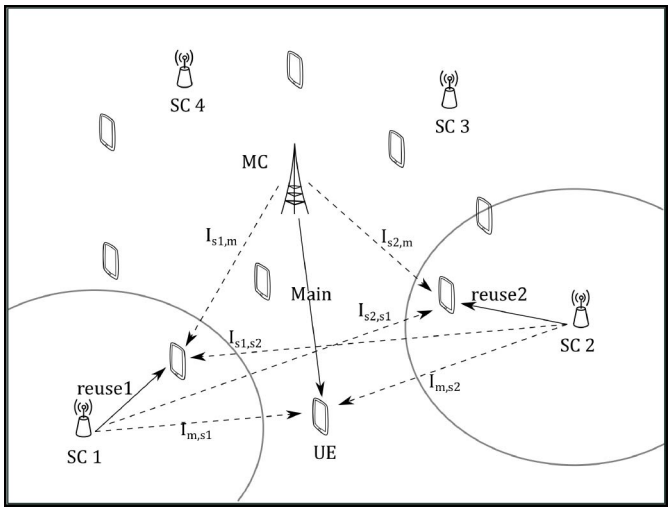


Fig. 1. Two tier network architecture, representing the main (desired) link as a solid line and the interferers with dotted lines.

B. Cell Association

UEs have different preferences regarding SINR, latency, and the number of requested resource blocks (RBs). Affected by what the cells are offering, the UEs decide the cell association. The cells have the responsibility of coordinating and distributing the spectrum between each other. Also, they manage the network load balancing and interference levels at the UEs, aiming to satisfy the different UEs' satisfaction levels and reach specific cell performance levels.

C. Channel Model

The large scale path loss *PL* used in our model is the simplified free space model [77]:

$$PL(dB) = \kappa + 10\zeta \log_{10}(d), \tag{2}$$

where ζ is the path loss exponent, and κ is a unitless factor that depends on the average channel attenuation, frequency of operation, and antenna characteristics. d is the distance between the UE and the serving cell. The macrocell spectrum organization ensures orthogonality between MC links; therefore, the inter-cell interference on the first tier level is assumed to be null.

In the presented downlink scheme, the interferences induced by spectrum reuse are cross-tier interference and co-tier interference. The cross-tier interference is caused by macrocell *m* at the user of small cell s_i: is given as $I_{s_i,m}$, and by a small cell at a macrocell user is given as I_{m,s_i}. In comparison, the co-tier interference from a small cell to a user of another small cell is given as I_{s_i,s_j}. The main (desired) link is represented as a solid line in Fig. 1, while the interference links are represented with dotted lines. The SINRs $\gamma_{n,m}$, and $\gamma_{n,s}$ at the *n*th user served by macrocell *m* and the small cell *s*, on

the *r*th resource block, are formalized respectively as:

$$\gamma_{n,m}^{(r)} = \frac{|h_{n,m}^{(r)}|^2 p_m^{(r)}}{N_{n,m}^{(r)} + \sum_{s \in S} |h_{n,s}^{(r)}|^2 p_s^{(r)}}, \tag{3}$$

and

$$\gamma_{n,s}^{(r)} = \frac{|h_{n,s}^{(r)}|^2 p_s^{(r)}}{N_{n,s}^{(r)} + \sum_{m \in M} |h_{n,m}^{(r)}|^2 p_m^{(r)} + \sum_{j \in S, j \neq s} |h_{n,s}^{(r)}|^2 p_s^{(r)}}, \tag{4}$$

where *S* is the set of small cells and *M* is the set of macrocells, $N_n^{(r)}$ is the noise variance, and $h_{i,m}$, and $h_{i,s}$ are the channel coefficients from the macrocell and small cells, respectively, to user *n*. The channel coefficients are determined from the large scale model via $h = 10^{-PL/20}$. $p_m^{(r)}$, and $p_s^{(r)}$ are the transmit powers of the macrocell and small cells over resource block *r* respectively.

The transmit power levels will be managed to decrease the interference and maintain an aggregate satisfaction level all over the network.

D. User Requests

The user *n* creates *u* requests per unit time *t*, with rate λ_r. The random variable *u* follows the Poisson process

$$P(u) = \frac{(\lambda_r t)^u e^{-\lambda_r t}}{u}. \tag{5}$$

For each user, the number of requested resource blocks *x* is a truncated normal distribution over the interval $0 < x < \infty$, with mean $\mu_x^{(n)}$ and standard deviation $\sigma_x^{(n)}$. The values $\mu_x^{(n)}$, and $\sigma_x^{(n)}$ change from one user to another depending on the user application. Hence, they can be considered as random variables of another process with distribution for a set of users. For simplicity, we assume the same values $\mu_x^{(n)}$, and $\sigma_x^{(n)}$ for all the users.

E. Spectrum Allocation

The proposed model describes a client-driven HetNet, where the users' requests and feedback drive the power and spectrum management processes. Unlike the traditional design approach, where a central authority makes decisions for the users based on a global objective (e.g., cell association decisions). The client-centric approach is a distributed system that makes use of the ongoing advances in the intelligence capabilities of network edge devices [78]. Instead of loading a centralized agent with an exponentially growing optimization complexity, the clients can coordinate or compete over the network resources (bandwidth, SINR, latency, etc.). UEs in real scenarios have different preferences and decision criteria depending on their applications. This motivates a client-driven scheme where each UE tries to maximize its satisfaction, and cells coordinate with each other to maintain their service levels. UEs decide which cell to associate with, depending on the cell's level of service and the UEs' local utility functions. Therefore, in our design, the UE-cell negotiations and interactions define how the system behaves.

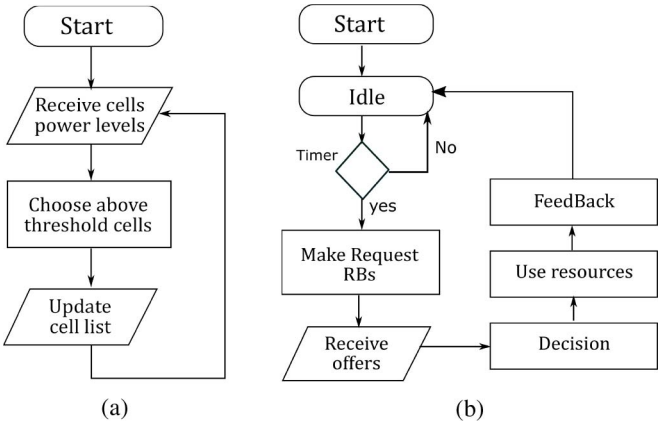


Fig. 2. User equipment flowchart; a) Collecting information about cells, b) Service request and usage.

IV. Proposed Agent Based Architecture

The proposed system architecture is described with several processes performed by each agent (UE, MC or SC) and a set of interactions between those agents. Each process is formalized with a flowchart, and an agent's behavior can be summed by several processes running asynchronously and in parallel. As in any ABM, the transitions between the flow chart states are dependent on the current state and the inputs from other processes.

A. Agent Based Processes at the User Terminals

A UE is assumed to be exchanging information with several nearby cells (macrocells or small cells) over the control channel, but it receives the service only from one of those cells. UEs make requests for service over the control channels, and they choose to be served by the best offeror. The service is characterized by three main variables: 1) the number of resource blocks (RBs), 2) signal-to-noise ratio (SNR), and 3) latency. All RBs occupy the same number of sub-carriers during a given time duration. The expected SNR value is calculated at the receiver. Latency in our work's context is the time it takes from the cell receiving the UE decision until the RB reaches the UE, which only depends on the queuing of the UE requests.

A UE agent is composed of two flow charts, as shown in Fig. 2. The first one manages the list of cells the UE is communicating with over the control channels. The second flow chart represents service requests' flow, receiving offers, decision-making, and reporting satisfaction levels. The cell list management flow chart runs when the UE is in idle mode. This certainly means that, at any time instant, either the first or the second flow chart is running.

The states can access a list of properties / local variables for each UE. The variables are position (geograhical location), mean request rate λ_r, mean requested *RBs*, the standard deviation of the amount of requested RBs, a list of nearest *N* cells, and utility function weights. The behaviors, inputs, and outputs of each state are described in Fig. 2. The flow chart in Fig. 2(a), is composed of a loop that, first, measures the receive

power levels P_{rx} over the control channels of the nearby cells:

$$P_{rx,c} = |h_{u,c}|^2 . P_{tx,c}, \tag{6}$$

where the subscript *c* corresponds to cells, P_{tx} is the transmit power level, and $h_{u,c}$ is the channel fading between user *u* and the cell c. Then it sorts the nearby cells by their $P_{rx,c}$ values and selects the first *N* corresponding cells. Finally, the UE updates the list values of the nearest N cells.

The flow chart in Fig. 2(b) describes a resource block request and utilization cycle. The UE is initially at the *'Idle'* state. Then it moves to the *'Make Request RBs'* state when a countdown timer reaches zero. The timer value is a random variable τ that corresponds to the interval time between two requests. It is assigned a new value after the timer expires, following the exponential distribution

$$f(\tau) = \lambda_r^{(n)} e^{-\lambda_r^{(n)} \tau}, \tag{7}$$

where $\lambda_r^{(n)}$ is the request rate for user *n*. Using an exponential distribution for τ ensures a Poisson distribution for the number of requests per unit time, shown in eq. (5).

The cells then send offers that depend on their transmit power levels, as we will observe in the following sections. The offer *G* is the vector

$$G_c = (RBs, f_1, f_{end}, t_1, t_{end}, P_{tx}), \tag{8}$$

composed of the number of offered resource blocks *RBs*, the start and end in the frequency domain given by f_1 and f_{end} respectively, the start and end in the time domain given by t_1 and t_{end} respectively, and the cell transmit power P_{tx}. The UE then collects all the offers at the *'Receive offers'* state, chooses the best offer, and sends an accept response to the corresponding cell.

Now depending on the diverse UE applications, the top offers could be quite contrasting for different UEs. For instance, some users may accept the lowest latency offer even if it has lower data rates; while others accept the service with a higher data rate, not caring about latency. The UE chooses the best offer based on the following mathematical utility function:

$$U(c) = \arg\max_c \left(\frac{RBs_c \times w_r \log_2(1 + 10^{(\gamma_c/10)})}{1 + w_d(t_{end} - t)} \right). \tag{9}$$

This function sets the UE service preferences by assigning the weights: w_r for the expected throughput at the receiver, and w_d for latency. The SNR value is estimated from the transmission power and path loss information between each cell and the user ($\gamma_c = P_{tx} - PL$). The w_r and w_d values are proportional to the importance of each corresponding factor. Note that the value $(t_{end} - t)$ represents how long it takes for the RBs to reach the UE. During the *'Use resources'* state, the UE measures the quality of service affected by the interference levels. Then it is shared with the serving cell in the *'Feedback'* state. Finally, before the UE returns to the *'Idle'* state, it reports its feedback to the serving cell. The feedback holds information about the interference levels, the SINR, the delay, and any satisfaction parameters that the cell uses to optimize the network.

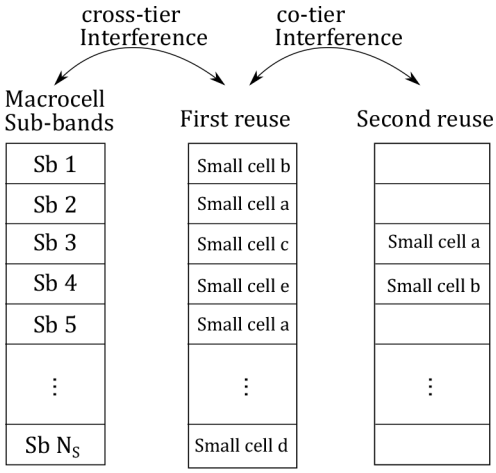
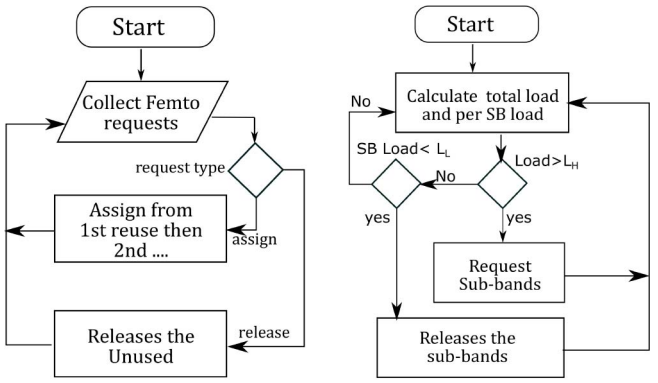


Fig. 3. Cells sub-band allocation and reuse.

Additionally, in order to incorporate handover processes and strategies, we must include processes that interact over the service duration and make decisions based on handover strategies. However, it will make the model more complex. Therefore, in this work, we assume that the UEs move only during their idle phase. We also assume that after the RBs are assigned by the serving cell, there will be no messaging between the UE and the cell until the feedback reporting step. Consequently, the flowchart that collects information about the cells is assumed to run only while the UE is idle.

B. Sub-Band Management Agent Based Processes at the Macro- and Small Cells

In our design, the cells are responsible for two main tasks: interference management and cell load balancing. In a spectrally efficient system, the small cells share the spectrum with the upper-tier (macrocells). Inspired by the previous studies mentioned in the introduction section [58], [59], [61], [62], [63] and in the context of our ABM modeling flexibility, in this subsection, we first propose interactions among the cells to manage the interference and the load balancing at the same time, with the help of the UEs feedback. We then propose comprehensive agent-based flowcharts for the proposed interactions. In the downlink scheme, the macrocells' bands are divided into sub-bands *Sb* higher in granularity than a resource block. The sub-bands are meant to be reused several times in a way that satisfies the cross-tier interference and co-tier interference thresholds, and they should also be allocated in consistency with the cell load and the traffic around that cell. The diagram in Fig. 3 shows two reuse levels where the sub-bands are to be allocated to the small cells over several phases. In the first reuse phase, the small cells and the macrocells coordinate to minimize the cross-tier interference by adjusting the small cells' transmit power levels. The macrocell users' feedback on the interference levels is used to adjust the small cells' transmission powers. The decisions for power level allocations are taken at the MCs level, while the SC requests initiate the sub-band assignment due to load requirements.



(a) Macrocell process. (b) Small cell process.

Fig. 4. Sub-band assignment to small cells flow charts.

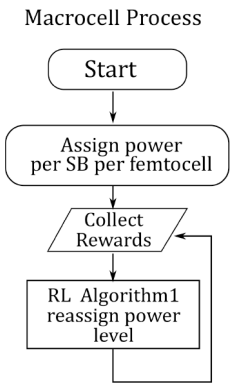


Fig. 5. Power management RL flow chart.

When sub-bands of the first reuse phase are full, the macrocell can start leasing from the second reuse sub-band. In this phase, the MC manages the co-tier interference levels reported by the SC connected UEs. This is accomplished by adjusting the transmit power levels of the reused sub-bands.

1) Small Cells Sub-Band Management: The processes in Fig. 4 manages the sub-bands assignment and release. A SC keeps monitoring its total load and the per-sub-band load as shown in Fig. 4(b). When it increases above a specific threshold L_H a request for sub-band assignment is sent to the parent MC. On the other hand, if the total load of the small cell decreases below a specific threshold L_L the SC sends a permission to the MC to release the least utilized sub-band.

Then the MC reacts to the sub-band assignment request, as shown in Fig. 4(a), by assigning one of the free sub-bands. The MC assigns the first reuse sub-bands then reassigns the same sub-band when the first reuse pool is occupied, as mentioned before. Moreover, it responds to the release permission by releasing the sub-band, and making it free to be reassigned to another SC upon request. The two processes in Fig. 4 complement each other and manage the sub-band assignment and release behavior.

C. Reinforcement Learning Processes

Reinforcement learning (RL) is used for two processes; first, to adjust the power levels in the reuse schemes; second, to assign the users to the sub-bands with highest performance level. Due to the nature of the algorithms where we have a list of actions that we need to choose from, the multi-armed bandit method is used as our model-free reinforcement learning method [79].

Multi-armed bandit is equivalent to a one-state Markov Decision Process [80]. This version of RL is chosen because of its simplicity and low computational complexity. Instead of having a state-action space, a multi-armed bandit algorithm has only one action space to choose from, hence the term 'arm'. Learning is done over rounds; in each round, an arm is chosen, and the corresponding rewards are collected during the round duration.

The proposed two RL algorithms have two different action spaces. The algorithm RL1, responsible for adjusting the power levels for the reused sub-bands, has the action space of power transmit levels for each sub-band. Whereas algorithm RL2, which is responsible for assigning the users to the sub-bands with the highest performance level, has the action space of choosing one of the serving cell's sub-bands. In the following subsections, we discuss these RL processes in more detail.

1) Small Cell Transmit Power Management: This process is running under the macrocell agents. The flow chart in Fig. 5 starts by assigning initial power levels for the small cells. Then it enters a loop of collecting rewards and updating the small cell power values.

The usage of a multi-armed bandit allows passing rewards and punishments (negative rewards) to learn the small cells' optimum power levels. As shown in the UE processes given in Fig. 2, the users create feedback information that is collected by the cells. This information is then reformulated by the cells as rewards for the MCs' RL process. The eventual goal of the algorithm is to maximize the rewards. The macrocell's multi-armed bandit algorithm components are as follows.

- *Action:* $A_i = \{a_i^{(p)}\}_{p\in\{P_1,P_2,...,P_k\}}$, where $a_i^{(p)}$ represents the power transmit level for the reused *i*th sub-band SB_i, from a set of transmit power levels, and *k* is the number of the power levels.
- Rewards R_i.
- Value function *Q*: Holds an evaluation for the expected reward for each action.
- Explorer factor ϵ.

The value function is updated via the recursive equation:

$$Q_{t+1}(A_i) = (1-\alpha)Q_t(A_i) + \alpha(R_i), \tag{10}$$

where α is a discount factor.

The reward function used for the proposed model is the aggregate SINR for all RBs in sub-band SB_i, over the last learning episode T_e:

$$R_i = \sum_{t_1>t-T_e} \sum_{RB\in SB_i} \gamma_{RB,t_1}. \tag{11}$$

The power management RL algorithm is shown below in Algorithm RL1. Deploying this algorithm determines the proper reuse power levels to achieve the maximum reward over each sub-band.

Algorithm 1: RL1 Small Cell Sub-Band Power Management Learning Algorithm

initialization $Q(.) = 0$;
initialize the reuse power levels $P(Sb)$
for each Sub-band *Sb* define power levels list
$A_i \in [P_1, P_2 \cdots P_k]$
while *1* **do**
  for $i \in$ *Sub-bands* **do**
    if *rand(.)*$< \epsilon$ **then**
      Explore: choose action from the $A_i()$ list randomly;
    else
      Exploit: choose action $A_i(t+1) = \arg\max_{a_i} Q_{t+1}$;
    end
    Receive rewards $R_i(t+1)$;
    Update Q table: $Q_{t+1}(A_i) = (1-\alpha)Q_t(A_i) + \alpha(R_i)$
  end
end

The multi-armed bandit algorithm first initiates the Q-table, the actions list of power levels, and the initial transmit powers. The algorithm then loops over the sub-bands by exploring the rewards of random power levels or exploiting the previously learned experiences.

In the second reuse case, two small cells transmit different power values for each sub-band. Therefore, the action space is two dimensional:

$$A_{i,j} \in \Big[(P_{F_{i_1}}, P_{F_{j_1}}), (P_{F_{i_1}}, P_{F_{j_2}}) \ \cdots \ (P_{F_{i_1}}, P_{F_{j_K}}) \ \cdots \ (P_{F_{i_K}}, P_{F_{j_K}})\Big]. \tag{12}$$

where *i* and *j* are the notation of the same sub-band for two different small cells, F_i and F_j. Finally, it is worth mentioning that, the utilization of the sub-band at the small cell is a measure of a successful power allocation by Algorithm RL1.

2) User to Sub-Band Association: Another factor that should be considered is that the UEs can have different performance levels for different sub-bands at the same cell. This is affected by the distribution of the set of users served by the cell and their distances from the interfering cell. Therefore, there should be a method to allocate each UE on the sub-band that suits its position with respect to the other agents (cells and UEs) in the network. The proposed process is based on an RL method, and like before, we choose the multi-armed bandit algorithm for that purpose. The process flow chart is shown in Fig. 6. The process starts by assigning the served users to the available sub-bands randomly. Then it keeps collecting the service feedback from the UEs and formulating the rewarding function from them. Each UE has its own Q-table that gets updated from the reward functions. The Q-table holds the values reflecting the learned performance per sub-band.

Note that if the process's computational overhead is an issue at the SCs, it can offload the learning process to each UE. In that case, the UE always informs the serving cell of its preferences instead of keeping a record for each UE at each cell. The learning algorithm components for the *n*th user are as follows:

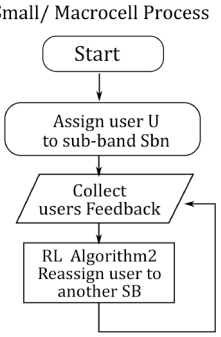


Fig. 6. UE assignment to sub-band RL flow chart.

Algorithm 2: RL2 User Sub-Band Choice Learning Algorithm

initialization $Q(.) = 0$;
define list of sub-bands at this cell
while *1* **do**
  if *rand(.)*$< \epsilon$ **then**
    Explore: Chose action from the $A_n()$ list randomly;
  else
    Exploit: Choose action $A_n(t+1) = \arg\max_{a_n} Q_{t+1}$;
  end
  Receive rewards $R_n(t+1)$;
  Update Q table: $Q_{t+1}(A_n) = (1-\alpha)Q_t(A_n) + \alpha(R_n)$
  Update list of sub-bands
end

- *Actions:* $A_n = \{a_n^{(s)}\}_{s\in\{1,\ldots,N_S\}}$, where $a_n^{(s)}$ represents the action of switching to one of the cell sub-bands.
- Rewards R_n: The received SINR level or satisfaction vector.
- Value function *Q*: Holds an evaluation for the expected reward for each action.
- Explorer factor ϵ.

The proposed reward function for this algorithm is:

$$R_n = \frac{\gamma}{1 + w_d t_d}, \tag{13}$$

where γ is the SINR, and t_d represents the delay experienced by the UE during the last served RBs. The factor $(1 + w_d t_d)$ normalizes the SINR level by the latency level to ensure that the users associate to the sub-bands, not only based on the SINR but also the sub-band load induced latency. The learning algorithm is described in Algorithm RL2.

D. Resource Blocks Allocation

We close the system model with the small cell RB assignment process illustrated in Fig. 7, which basically elaborated on the mechanism of the SC response to UE requests. The SC process keeps listening to the UE requests. Once it receives a request, it finds the sub-band which is suitable for this UE. The suitable sub-band is already determined in the RL process described in Section IV-C2.

Now based on the SC current load, an offer is formulated. Ideally, if the SC is not congested, the offered RBs will be the same number as the requested RBs. However, if the SC is congested, a discounted offer with fewer RBs can be made.

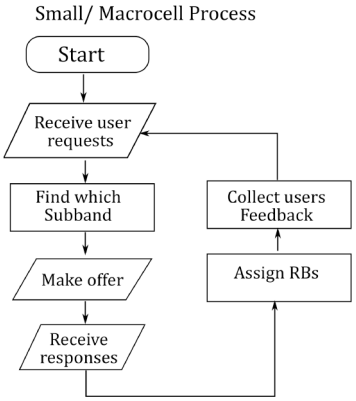


Fig. 7. RB assignment flow chart.

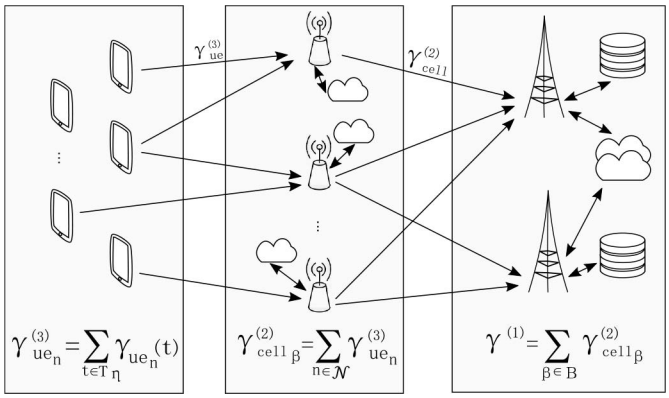


Fig. 8. RL Algorithm 1 computational framework.

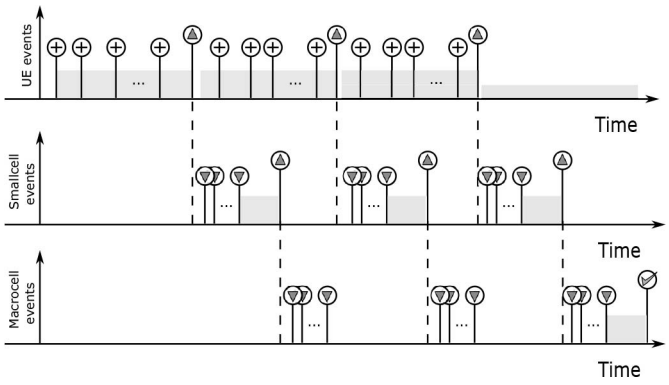


Fig. 9. RL1 Algorithm Events timing over one learning episode, where ⊕, ▲, ▼, and ✓represent $\gamma^{(1)}_{ue_n}$ calculation, sending to the upper tier, receiving from the lower tier, and updating the value function respectively.

The analysis of discounted offers is out of the scope of this study and will be analyzed in future versions of this work. The offer, previously expressed in eq. (8), has information about transmit power P_{tx}, latency (t_1, t_{end}), and band of service (f_1, f_{end}).

It is also worth mentioning that the selected RBs will join a queue if those RBs are being utilized by another UE in the current time instance. After the offers from several cells are sent to the UE, and the UE makes a decision. The selected SC receives the response and assigns the requested RBs to the UE. In comparison, the other SCs will time-out and terminate the request. Finally, after the UE has finished using the RBs, the SC receives the users' feedback. The feedback contains values for the reward functions given in eqs. (11) and (13). Therefore, it has the SINR γ, the delay t_d, and the delay weight factor w_d.

E. On Convergence of Multi-Armed Bandit Algorithm

The multi-armed bandit was first presented under the concept of sequential sampling in [81], and strategies were discussed to attain convergence. Later, bounds for the expected regret were studied in [82], where the regret, as a function of time, is the difference between the recent reward and the ideal maximum reward. In [83], the multi-armed bandit problem was studied under the probably approximately correct model. It was shown that it is sufficient for *n* arms to be sampled $O(\frac{n}{\varepsilon^2} log(\frac{1}{\delta}))$ times in order to reach an ε-optimal arm with probability $(1-\delta)$. The same lower bound for the expected number of trials was also presented by [84]. In [85], successive elimination and median elimination processes were proposed showing improved expected error bounds.

F. Distributed Computing

In this section, we discuss a distributed computing framework for the reinforcement learning algorithms RL1 and RL2. We propose a practical deployment for reward collection and value function updates, though this should not be the only way to deploy the aforementioned algorithms. First, we discuss the rewards distributed computation and its timing schedule for the RL1 power management algorithm. Then we do the same for the RL2 UE sub-bands preferences algorithm.

From the value function, eq. (10), the operations needed to calculate Q_{t+1} are: a database read to fetch the Q_t values, rewards R_i summation operations in eq. (11) and eq. (13), and pair of multiplication operations with α. The final *R* is a single value that corresponds to an action in A_i, therefore, the multiplication operations by α are not as intensive as calculating R_i. To determine R_i, the summation operations required to collect info from all the UEs can be massive for dense networks. The number of summations required is the number of users in the network multiplied by the average number of feedbacks in one learning episode. So, here we focus on distributing the rewards calculations.

1) RL1 Computational Framework: For RL1, the rewards in eq. (11) are distributed as shown in Fig. 8 between the users and the two tiers of the SCs, and MC. In a specific learning episode, scheduled by the MC, the UEs sum the SINR experienced on a sub-duration T_η. The summation process for this tier at user *n* is formalized as follows:

$$\gamma^{(3)}_{ue_n} = \sum_{t \in T_\eta} \gamma_{ue_n}(t), \tag{14}$$

and shown in the timing diagram in Fig. 9 with the symbol ⊕ on the UEs' timeline. The superscript indicates the tier level, where 1 is for the MCs, 2 is for the SCs or a MC that supports service, 3 is for the UEs. At the end of this sub-duration, the summed rewards are uploaded to the upper tier of the small cells, this is represented by symbol ▲. After a propagation delay and when all the summed values arrive,

represented by ⓥ, the SC sums those values. The summation operation in this tier is formalized as:

$$\gamma^{(2)}_{cell_\beta} = \sum_{n \in \mathcal{N}} \gamma^{(3)}_{ue_n} \ , \tag{15}$$

where user n is a member of the SC set of users $\mathcal{N}$. The summation duration leaves a gap in the timeline, after which the values are uploaded to the MC, represented by ⓐ. On the MC timeline, the rewards are received from the small cells in groups after each sub-duration, represented with ⓥ on the MC timeline. The purpose of the sub-duration is to introduce pipelining between the collection of data durations and computation durations, hence increasing the rewards sampling duration. At the end of the learning episode, the MC uses all the received rewards and calculates the vector total rewards over each SB using:

$$\gamma^{(1)} = \sum_{\beta \in B} \gamma^{(2)}_{cell_\beta}, \tag{16}$$

where the cell β belongs to the set of serving cells B under the upper MC, with the MC as a serving cell included.

As a result, the RL1 summations are distributed between the agents as follows; a UE on average has a number of summations that is equal to the value of feedbacks per learning episode, a 2nd tier node (serving cell) has a number of summations that is equal to the served UEs, and a 1st tier node (MC) has a number of summations that is equal to the cells being managed under it.

We notice from the timelines that in a practical network, with information propagation delays and computational durations, the reward sampling duration is a subset from the whole learning episode duration. However, as long as the reward sampling duration is same for each episode, the ratio between the value function elements will be the same. This is under the assumption of constant or slow varying network statistical characteristics over the learning episode duration. The reward sampling duration will act as a scaling factor for Q but it does not affect the algorithm decisions.

2) RL2 Computational Framework: For RL2, it runs between the UEs' tier and the serving cells', as shown in Fig. 10. The rewards are collected from the UEs following eq. (13), then it is uploaded to the serving cell as shown in the timeline in Fig. 11. The reward collection over this tier is formulated as:

$$R^{(3)}_{ue_n} = \sum_{t \in T_{\zeta n}} \frac{\gamma(t)}{1 + w_d t_d(t)}, \tag{17}$$

where $T_{\zeta n}$ is the RL2 learning episode duration of user n. We can also observe that this algorithm is meant to run in a faster manner than the power management algorithm, and that the learning episode duration is shorter in order to have several Q value updates within the RL1 episode. In other words, every time the power levels are changed by RL1, the users try to update their favorite sub-bands via RL2. In the timeline in Fig. 11, symbol ⊕ represents the summation operation is eq. (17). Symbols ①, ②, and ③ represent the uploads to cell 1, cell 2, and cell 3 respectively. The symbol ✓ represents a value function update.

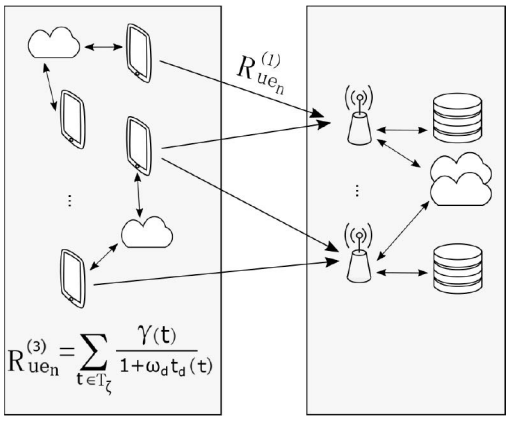


Fig. 10. RL Algorithm 2 computational framework.

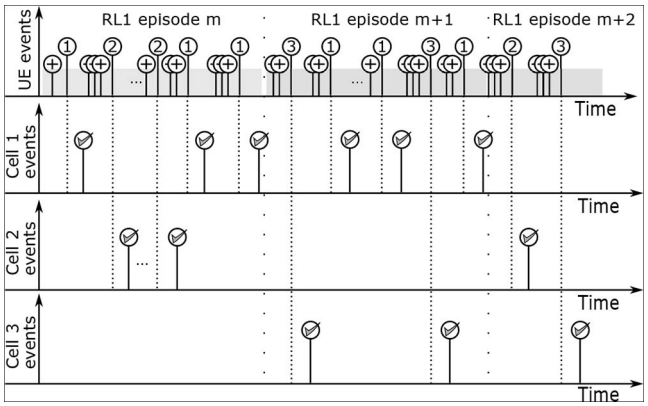


Fig. 11. RL Algorithm 2 Events timing, where ⊕ represents the collection of UE rewards. ①, ②, and ③ are the info upload operation to the consecutive cells. ✓ is the value function update operation.

3) Energy Concerns: For value function update operations, depending on the UE energy capabilities, there are several choices for which node should update the value functions, the duration of the learning episode, and the number of rewards gathered before each upload. For this algorithm, UEs can act independently with different Q update rates, hence, different learning episode duration and summations to upload operations ratios. Each strategy will affect energy consumption differently, and there will be a compromise between energy consumption and the conversion rate for the whole system. The detailed analysis of this is beyond the scope of this work and is left for a future study.

A reward gathering node (UE or cell) spends its power on summation operations and upload to the upper-tier operations. For the UEs, we propose to save the energy consumption of uploading through the air by buffering the rewards and summing their values, as in eq. (14) and eq. (17), then uploading them. Further energy optimization can be achieved by adjusting the ratio of the summation operations to the upload operations. This will require the introduction of energy consumption models of the UE agents. Increasing the ratio of summation operations to upload operations should save energy at the UEs, but it may affect the conversion rate of the whole network. The compromise between energy consumption and network conversion rate is also worth investigating in a future study.

4) Cloud Computing: In our architecture, intensively computational operations can have one of the following two options, depending on its power and computational capabilities. If the node is able to handle the reward gathering computations or Q function updates, the operation can take place at the cell. Otherwise the computations can be offloaded to a cloud service or cloudlets. A cloudlet is a small-scale cloud data center positioned at the edge of the network, intending to support computing resources with low propagation delays.

When the rewards calculations of RL2 are run by cloud services instead of running by the UE, the UEs will only have to upload the values of γ and t_d. This will assume a direct link between the UEs and the cloudlets as shown in Fig. 10.

The value function updates can be saved on the MCs or they can be uploaded to remote datastores, depending on the MC capabilities and the amount of Q function history that needs to be saved. Theoretically, only the final Q value needs to be stored, but the history is useful for monitoring and debugging purposes.

The UE decision operation in the "service request and usage" flow chart, has a utility function that is calculated. In this step, the UE evaluates the received offers using eq. (9). This computation can be transferred to a cloudlet for a low computational capability UE. However, unlike the previous cases of RL calculations, delegating this computation will increase the latency of the service. This is because this operation belongs to the service plane, while the RL computations belong to the control plane.

Moreover, several deployment technologies can be utilized to implement our proposed architecture. The telecommunication industry is evolving toward highly decentralized systems like distributed network Function virtualization (NFV). Our proposed processes can also be implemented as virtual network functions (VNF) that get deployed as virtual machines VMs or containers in a network functions virtualization infrastructure (NFVI), or on the cloud, as mentioned in the previous section. The network topologies and connections for the cloud or the NFVI should take into consideration the propagation delays of the computations, and the virtualization overheads and their effect on the RL convergence rates [86].

NFV can be build using the microservices paradigm to deploy their functions as services. The services in a microservices architecture are fine-grained, and the protocols are light. Asynchronous protocols like Advanced Message Queuing Protocol (AMQP) or synchronous protocols like HTTP/REST are used to communicate between services. Services are built and deployed independently from one another, and each service has its own database. Another deployment architecture that can be considered is the novel multi-agent-based autonomic network management system MANA-NMS [87].

In the next section, we simulate the proposed system with different scenarios and utility functions.

V. Simulations and Discussions

This section presents simulations of the proposed ABM architecture, and evaluates its behavior in different scenarios and modes. For the simulator, we used MATLAB to build an object-oriented program. Agents were defined as classes with methods corresponding to the agents' procedures. Moreover, input and output objects were included for each class to handle the communication between agents. An environment code was written to instantiate the agent classes and assign their positions and characteristics from predefined statistical properties.

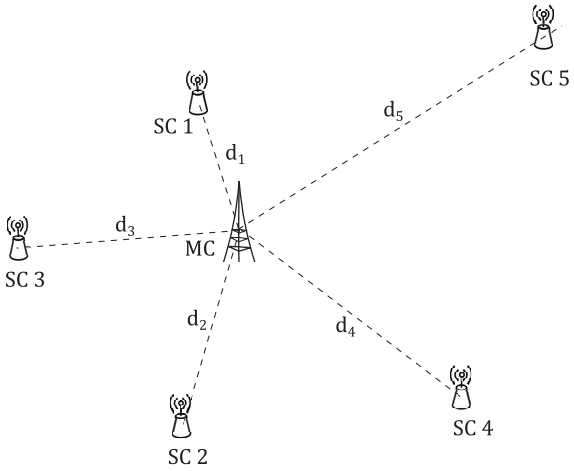


Fig. 12. Simulation environment.

In real life, the network entities/agents run in parallel while in a computer program codes run sequentially. Therefore, a scheduler was created to mimic the system parallelism. The scheduler loops over all agents every single time unit and the called agent will read its input ports and do the procedure related to the received message. The order of the agents in the loop is randomized in every iteration to avoid any biases.

The scheduler works with the asynchronous update mode [88]. It manages the cell agents that should run with higher rates than the UE agents, because it has one-to-many relationships. Finally, post processing codes are responsible for parsing the history objects in each agent to plot the results shown in this section.

We evaluate one network structure with several numbers of users. This creates several scenarios with different numbers of sub-band reuse. The simulated network contains a macrocell, and under it, there are five small cells, as shown in Fig. 12. The small cell distances from the macrocell are in an increasing order; $d_1 < d_2 < d_3 < d_4 < d_5$. For the sake of comparison we fix the positions of the cells for the entire simulations. This section is divided into two subsections. In the first, we simulate a static deployment of the spectrum. Then, in the second, we simulate a scenario where the dynamics of spectrum deployment are demonstrated.

A. Static Spectrum Deployment

In this subsection, different reuse cases are considered by fixing the macrocell sub-band reuse to specific numbers. Then we evaluate the network for each case with different numbers of users.

1) Single Reuse Case: First, for the single reuse scheme, the macrocell has its spectrum divided into six sub-bands, and

TABLE I
SIMULATION PARAMETERS

Parameters	Values
Cells positions	macrocell(x,y) = (5, 5) km small cell1(x,y) = (7.8, 7) km small cell2(x,y) = (4.4, 4.2) km small cell3(x,y) = (6.1, 2.6) km small cell4(x,y) = (1.3, 8.5) km small cell5(x,y) = (1.4, 8.4) km
Users positions	PSSS distributed in the area [x=[0, 10] km, y=[0, 10] km]
Number of users	single reuse case [800, 1400] users second resuse case [1000, 1800] users 5th reuse case [1000, 2700] users
Macrocell tx power	30 dBm
Small cell tx power range	[10, 25] dBm
Pathloss model	$25\log_{10}(d) + 40$
Number of RBs per sub-band	10 RBs
Request rate for UEs	$\lambda_r = 0.012$ request per tick
Requested RBs statistics	Avg. RBs per request= 4 , Std. dev. RBs per request= 2
Rewards function of RL Algorithm 1	Eq. (11)
RL1 Learning episode length	200 T
Rewards function of RL Algorithm 2	$(\gamma/(1 + w_d \times delay))$
RL1 explore factor	decreasing from 0.8 to 0
RL2 Learning episode length	1 UE-request cycle
RL2 explore factor	fixed 0.3
Load request and release thresholds are set to the extreme values so that the corresponding processes are off during static spectrum deployment.	

it is sharing one sub-band with each small cell. The sixth sub-band is not reused, and it is left for the macro users who cannot share their spectrum with any of the small cell users. This corresponds to the partial spectrum deployment (PSD) scheme mentioned in the literature review section. Second, a scenario where spectrum is reused twice is considered, then finally, we simulate a co-channel spectrum deployment where spectrum is reused five times (the number of small cells). The environment and agent parameters are tabulated in Table I.

After running the simulation environment with 1000 UEs for a duration of 5000 T, where T is one resource block duration, the network converges to the results shown in Fig. 13(f). The distribution of the users over the first four sub-bands is shown in each of the sub-figures. The marker color and shape corresponding to the cell association is explained in the figure legend. One can observe that the small cell users are clustered around their cells. At the same time, the numbers of users associated to a small cell are proportional to its distance from the macrocell due to interference levels. Also, the macrocell users on a specific sub-band have lower densities around the small cell using this sub-band due to Algorithm 2.

The results in Fig. 14 show the number of users associated with the small and macrocells for different numbers of network users. A higher number of users is associated with the macrocells, and as their number increases, the load becomes more balanced between the two tiers.

2) Second Reuse Case: We then increase the number of users above 1600 user. With the rate of requests and number of RBs in Table I, the single reuse case is not sufficient for catering to user demands, and the small cells start requesting for more spectrum from the macrocell. At this point, the second reuse is enabled, and more sub-bands are offered for the small cells. The users' distribution over the sub-bands and cells

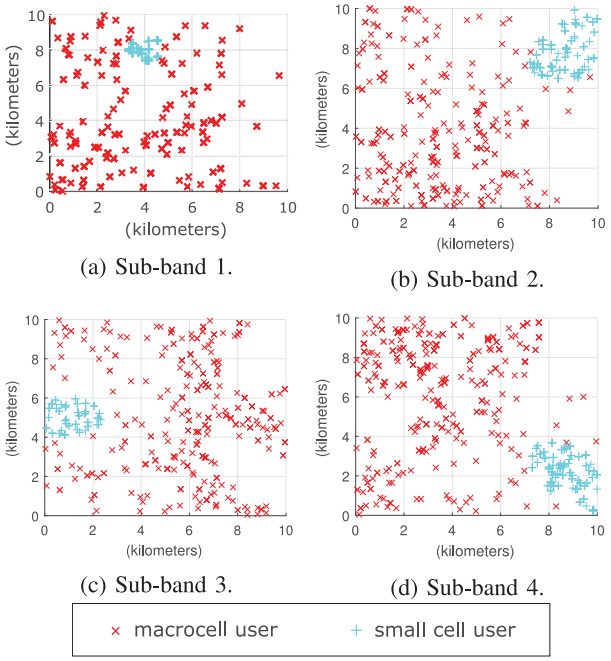


Fig. 13. Users distribution, single reuse case.

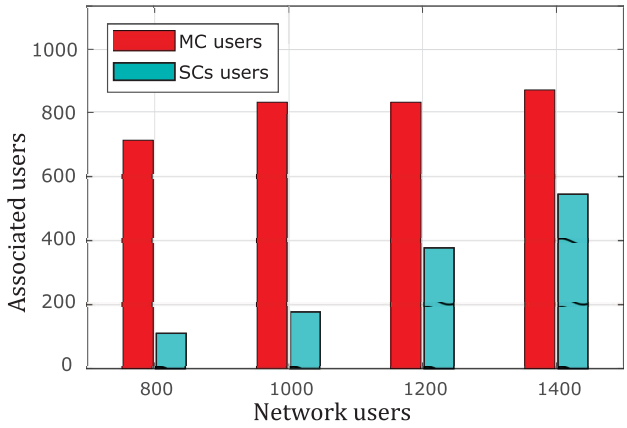


Fig. 14. User association, in a single reuse case.

are shown in Fig. 15(f). Here we can observe that the learning algorithms of the small cells that are closer to the macrocell settled at a transmission power with lower coverages. This allows for lower co-tier interference and better coexistence with the macrocell tier.

The progression of aggregate network SINR for 1800 users is shown in the curve in Fig. 16. During the duration from $t = 0$ to $t = 5000T$, the exploration factor of Algorithm 1 decreases linearly from unity to null. After that, the algorithm starts exploiting the knowledge it gained during the exploration phase. Therefore, we observe a rise in the network SINR during the exploration phase, and it saturates during exploitation. The other two curves show the aggregate SINR over the macrocell and the small cells in the same figure. The throughput is exchanged between the macrocell and the small cells until it reaches a combination that maximizes the sum of both. The number of active users is shown in Fig. 17. For the simulated request rate, the number of active users at any

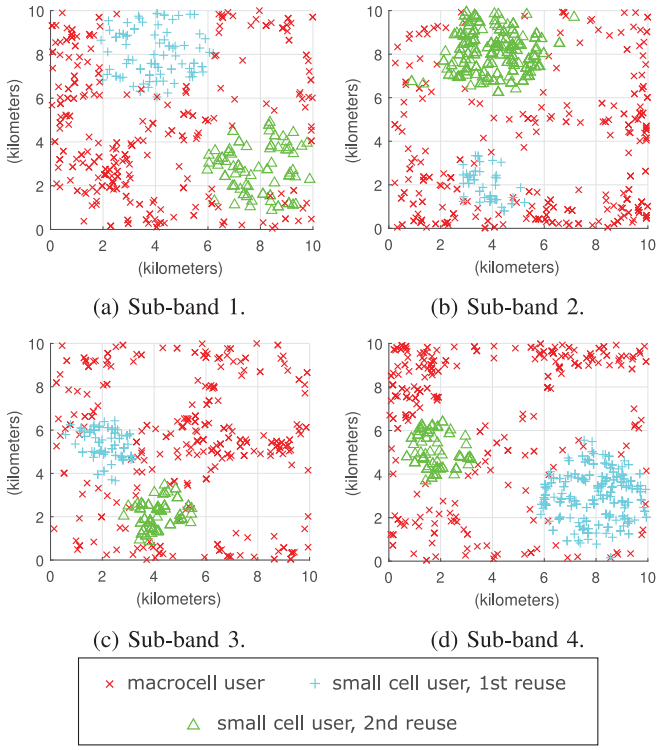


Fig. 15. Users distribution, 2nd reuse enabled.

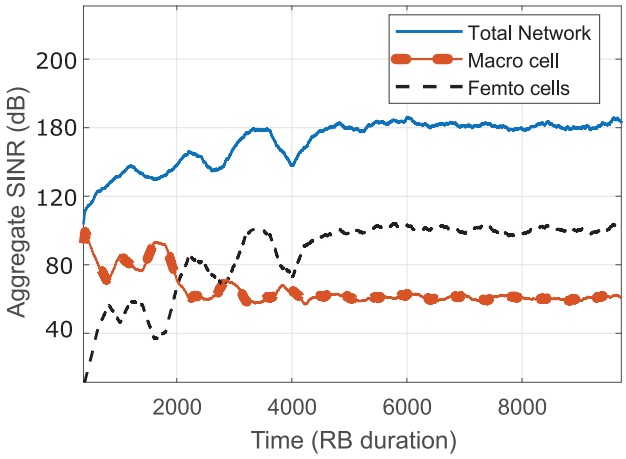


Fig. 16. Aggregate SINR progression in time (1800 user).

point in time is around 12. Still, during the learning phase, the users association keeps moving between the first tier and the second tier until the transmission powers finally converge to the solution that maximizes the utility function given in Algorithm 1.

To evaluate the load balancing in the second reuse case, we simulate different numbers of network users and determine the number of associated users in the two tiers, as shown in Fig. 18. Here, due to the increase of the small cells sub-bands, the users' association is more tilted towards the small cells' tier. As the number of users increases, more users are associated with the macrocell.

3) Full Reuse Case: We skip the third and fourth reuse cases and enable the sub-band reuse five times. This

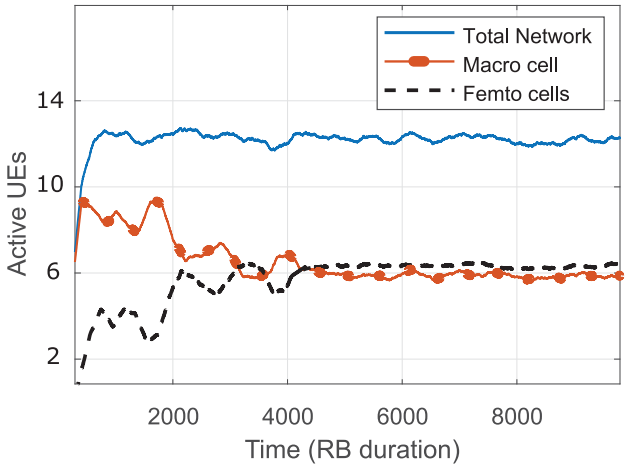


Fig. 17. Number of active users.

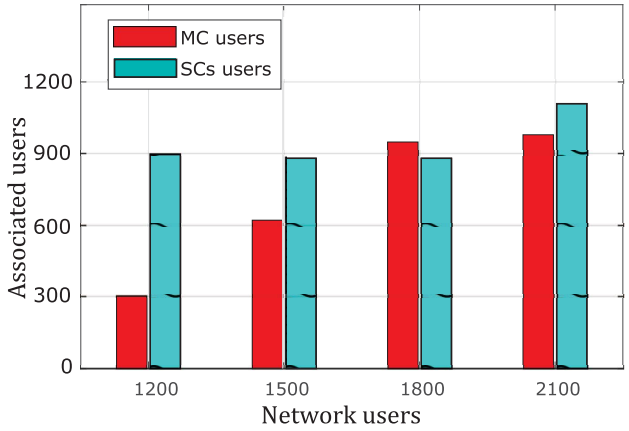


Fig. 18. User association, in a second reuse case.

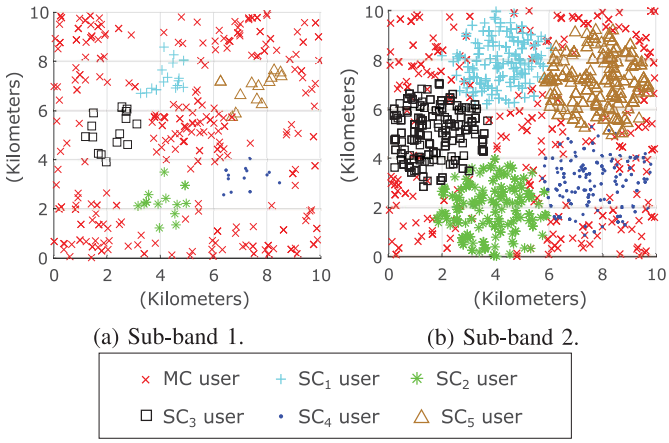


Fig. 19. Users distribution, 5th reuse enabled.

corresponds to CCD deployment. The user distribution for the fifth reuse case is shown in Fig. 19(d) for only the first two sub-bands, as it looks almost the same for the other sub-bands. One can observe that the macrocell users are associated in large proportion with the first sub-band, where they experience less inter-tier interference. This is due to the low transmission power of small cells over this sub-band. Hence, we observe a lower number of small cell users in this sub-band. The opposite

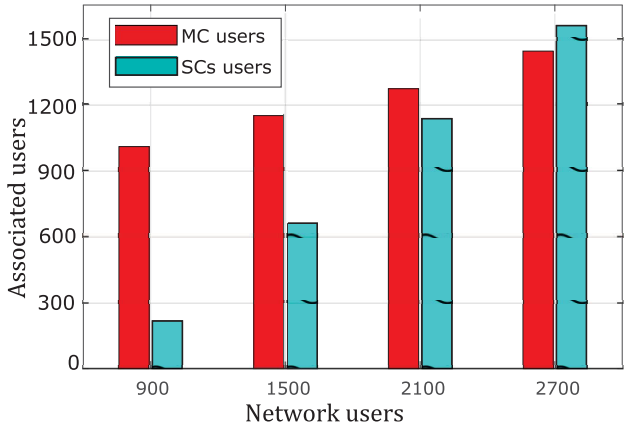


Fig. 20. User association, a full reuse case.

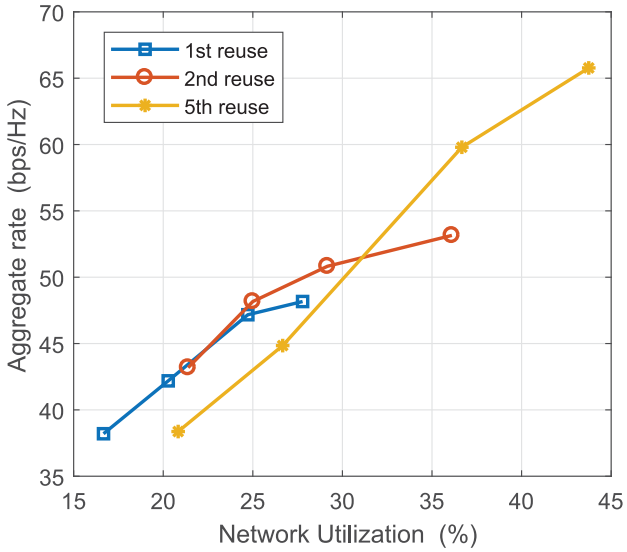


Fig. 21. Network aggregate throughput.

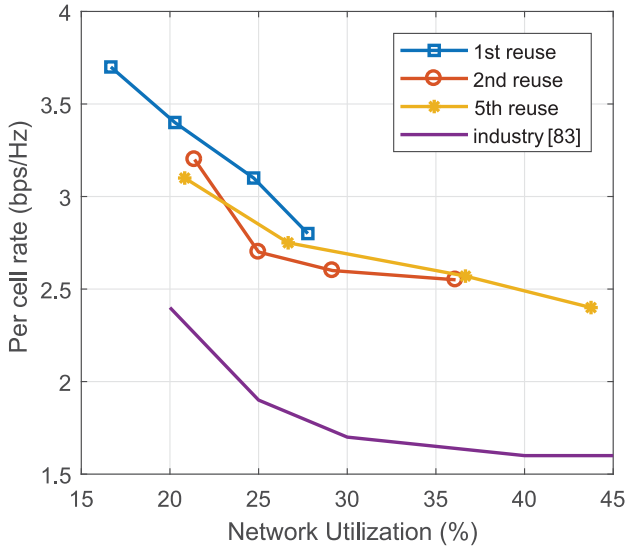


Fig. 22. Per-cell rate.

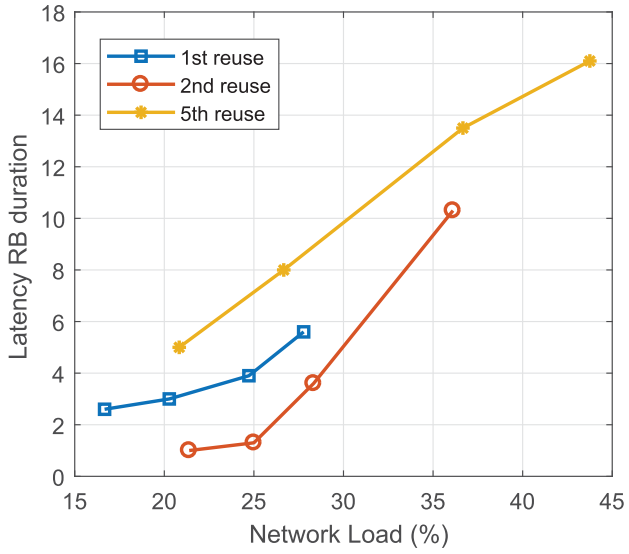


Fig. 23. Average user latency.

behavior is observed for the second sub-band, where a larger proportion of users are associated with small cells.

User association is re-evaluated for this full reuse case in Fig. 20. Despite the abundance of small cell sub-bands, the network reaches a solution where fewer users are associated with the small cell tier due to the high co-tier interference. But as the number of users increases, more users become associated with the small cells tier.

Next, we sweep over the number of users to analyze the aggregate network rate versus the number of users and the per-cell rate versus the corresponding network utilization for several reuse cases. The results are shown in Fig. 21. Several observations can be deduced from the figure. First, during the single reuse operation, there is a maximum number of users that can be served before the increase in aggregate rate starts to decline. At that point, second reuse should be enabled, and the small cells with high traffic should be assigned sub-bands that are reused for the second time. By moving to the second curve at the second reuse case, the users are distributed over more sub-bands which increases the aggregate throughput. However, the increase is nonlinear due to higher interference levels. The second reuse also saturates at a specific network load. The reasoning is the same as the single reuse curve. For the full reuse case (5th reuse), aggregate throughput is lower as compared to 1st and 2nd reuse scenarios for corresponding network utilization. This is due to the increase in interference levels at full reuse. However, higher reuse is essential to serve more users as evident by higher network utilization values in the figure.

We can also extract the relationship between the whole network load and the per-cell spectral efficiency plotted in Fig. 22. The numbers reported in the industry [89] are also shown in the figure as a reference. The gain over the reported industry rates is due to the emergent coordination between the small cell power assignment and sub-band user association algorithms.

Latency is also affected by the network load at different reuse cases. Fig. 23 shows the average latency per user in

RB duration T, versus network load. As the utilization of the resources increases in the single reuse case, the requests' queuing results in an increase in latency. The same happens to the second and the full reuse cases. It is also worth mentioning that more users were using the macrocell than the small cells for the first and the full reuse cases. Therefore, latency values for both cases were higher than the second reuse case, where the user distribution between the two tiers was more balanced.

4) Comparison With Benchmark in the Literature: For comparison with solutions in the literature, we compare between the proposed utility functions, having the aggregate SINR reward function for Algorithm RL1 and eq. (13) with $\omega_d = 0.05$ for Algorithm RL2, and the following utility functions from the literature:

- Maximum SINR for small cell users given that the macrocell users do not drop below a specific SINR (10 dB simulated) due to interference, by Bennis [48].
- Inter-cell interference coordination (ICIC) [50], [90], which is similar to our work, composed of two parts: sub-channel allocation and power assignment algorithms.

These frameworks have been adapted in our architecture to be comparable with our proposed algorithms. For the first study, the utility function of Algorithm 1 is as in [48], with the assumption that the interference information of macrocell users is shared with the small cells. Algorithm 2 is deactivated as it has no relevance in this study. The utility function used for Algorithm 1 is as follows:

$$U_1 = \arg\max_{p_i \in P} \sum_{t_1 > t - T_e} \sum_{k \in K} \log_2(1 + \gamma_k^{(RB)}) \mathbb{1}_{\{\gamma_m^{(RB)} > \Gamma_{th}\}}, \tag{18}$$

where the indicator function $\mathbb{1}_{\{condition\}}$, is 1 when the condition is true and 0 otherwise. Γ_{th} is the minimum allowed macrocell user SINR.

For the ICIC, in the context of our architecture, power assignment is managed by RL Algorithm 1, with the utility function as minimum aggregate interference over a sub-band, while the sub-channel allocation is managed by Algorithm 2 using the utility function to minimize the interference level per user. A delay factor $w_{d_{ICIC}}$ was added to manage the load distribution between the sub-band and to manage the latency. The utility functions used for Algorithm 1 and 2 are as follows:

$$V_1 = \arg\min_{p_i \in P} \sum_{t_1 > t - T_e} \sum_{RB \in SB_i} I_{RB,t_1} + (w_{d_{ICIC}} t_d), \tag{19}$$

$$V_2 = \arg\min_{s_n \in S} (I_{RB} + (w_{d_{ICIC}} t_d)). \tag{20}$$

Without the w_d factor, the load is not guaranteed to be balanced between the sub-bands; hence, high per-user latency values can occur.

The results of the comparisons are shown in Fig. 24 and Fig. 25, for the per-user throughput cumulative distribution function (CDF), and per-user latency complementary cumulative distribution function (CCDF). The average throughput for the simulated number of users (1000 users) is found to be higher for the maximum aggregate SINR utility function case. On the other hand, the ICIC framework has a slightly lower number of low throughput users. This is due to the focus of

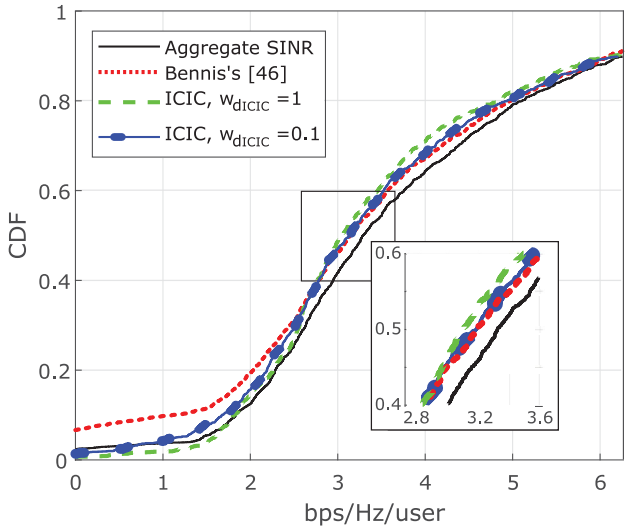


Fig. 24. Per-user throughput CDF.

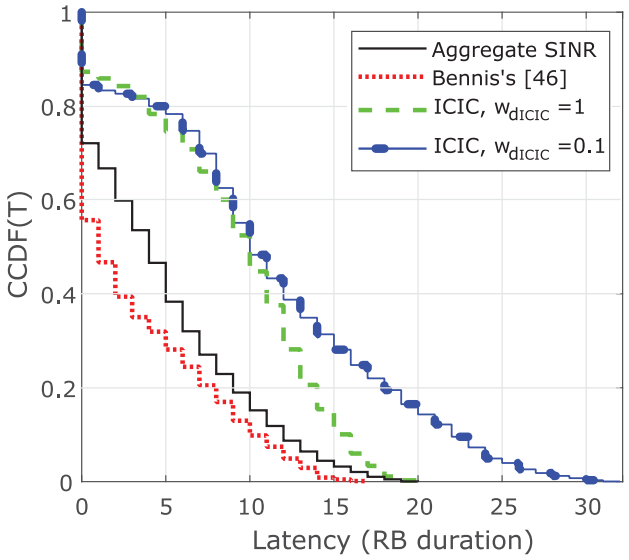
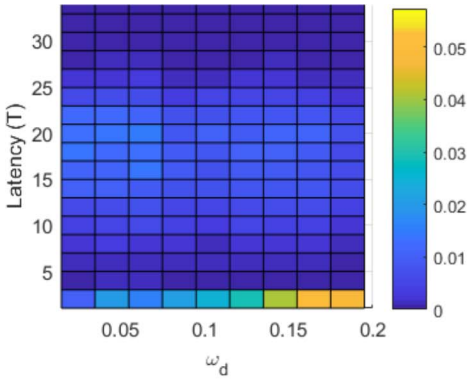


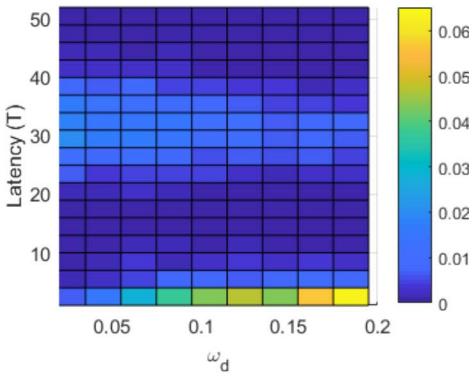
Fig. 25. Latency CCDF.

the ICIC algorithm on minimizing the interference. The work in [48] has a higher number of low throughput users due to not deploying a sub-band or a sub-channel algorithm, as in Algorithm 2.

Finally, the per-user latency CCDF is a measure of the efficiency of load distribution between the macrocell and small cells; and amongst small cells. The two aggregate SINR based methods achieved lower latency values than the minimum interference-based method. The usage of Algorithm 2 added latency due to the behavior of the users of preferring to utilize sub-bands that are not reused more than the reused sub-bands. This can result in some non-uniformity in sub-band utilization, hence the slight increase in latency. We can observe this effect more prominent in the case of the ICIC algorithm, where this imbalance can be managed by modifying the utility function in Algorithm 2 to take sub-band association decisions based on the delay. Hence, we observe a prominent effect of w_d on



(a) 900 user.



(b) 1400 user.

Fig. 26. Latency vs. ω_d joint distribution, given that UEs' ω_d is uniformly distributed over the range [0.01 0.2] (Full reuse case).

the latency results. In h next section we give more insight on the latency distribution.

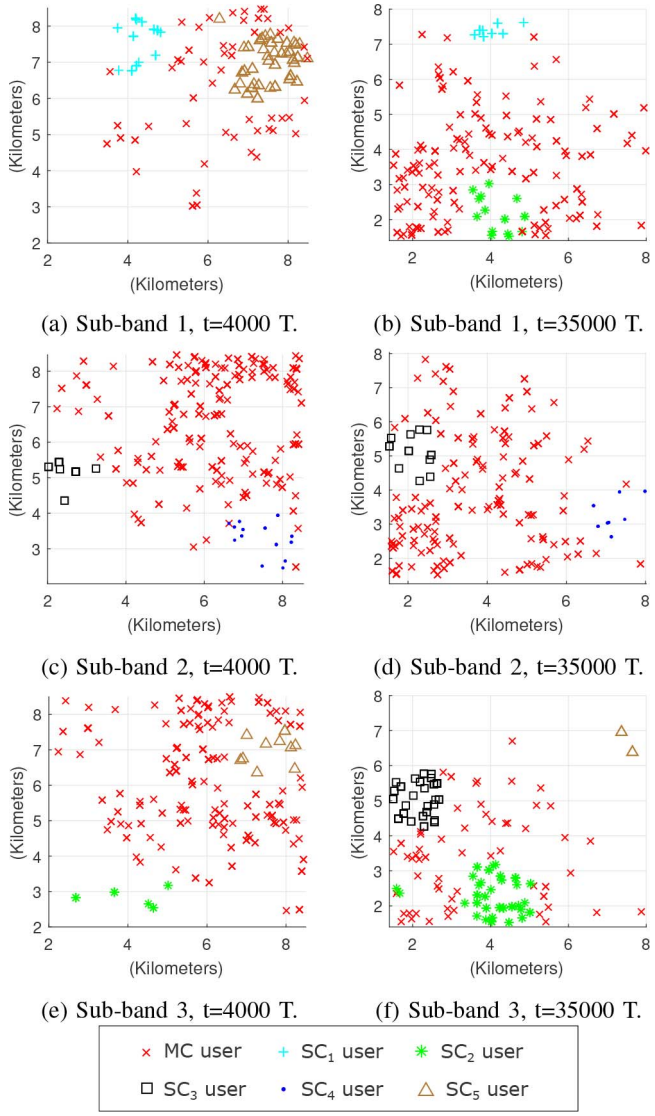
5) Queuing Latency for Different UEs With Different ω_d Values: Here we shed more light on the latency distribution between the users and its relationship with the latency weight value ω_d. This value was presented in two cases, the first is the decision operation in eq. (9) and the second is in the reward eq. (13) and eq. (17). For both cases, the value ω_d represented the UE's preference of low latency or tolerance for high latency.

For the full reuse case scenario, we create a number of users with ω_d as a random variable distributed uniformly from 0.01 to 0.2. The measured queuing latency is in T. Here we evaluate the joint distribution between user latency and the user's ω_d parameter for two different traffic loading. For this simulation, we use the decision function in eq. (9) with $\omega_r = 1$, and the rewards in eq. (13) and eq. (17).

We can observe in Fig. 26 that the users with low ω_d experience higher latency, with an average that depends on the network load. On the other hand, the users with higher ω_d values has higher chances of being served with low latency RBs. Hence, devices with stringent latency requirements can be assigned to relatively higher ω_d values to meet their latency demands.

B. Dynamic Spectrum Deployment

In the previous subsection, we fixed a different number of reuses for sub-bands to characterize the network



(a) Sub-band 1, t=4000 T. (b) Sub-band 1, t=35000 T.

(c) Sub-band 2, t=4000 T. (d) Sub-band 2, t=35000 T.

(e) Sub-band 3, t=4000 T. (f) Sub-band 3, t=35000 T.

Fig. 27. Users distribution for three sub-bands.

behavior for each scenario. In this subsection, we enable the sub-band assignment and release requests at the small cells by assigning upper and lower load limits. The upper load limit $L_u = 45\%$ defines the threshold above which the small cell requests sub-band assignment from the macrocell. The lower load limit $L_l = 8\%$ is the threshold below which the small cell releases the least used sub-band. The positions of the cells and channel characteristics are the same as in Table I. For this scenario, users' distribution is nonuniform to examine how the network balances the load and runs its learning algorithms in an inherently unbalanced network. The 300 users are initially condensed in the upper right corner of the simulated area shown in Fig. 27(a), Fig. 27(c), and Fig. 27(e).

We give the network a duration of 10000 T to train and stabilize. Then, the users are gradually allowed to move from the upper-right corner to the lower-left corner as shown in Fig. 27(b), Fig. 27(d), and Fig. 27(f). Note that, we assume in this simulation that the UEs move only during their idle mode.

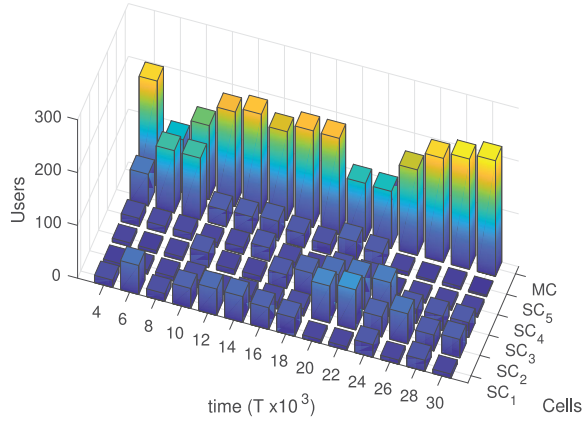


Fig. 28. Cell association progress in time.

TABLE II
INITIAL SUB-BAND ASSIGNMENT

Time (T)	Event	Action	Explore factor
1	Start	SB_1 assigned to $SC_1, SC5$ SB_2 assigned to $SC_3, SC4$ SB_3 assigned to SC_2	1
4000	SC_5 load >L_u	SB_3 added to SC_5	1
6000		exploitation	0
8000	SC_5 load >L_u	SB_2 added to SC_5	1
10000		exploitation	0
20000	SC_5 load <L_l	SB 2 removed from SC_5	1
21000		exploitation	0
25000	SC_2 load >L_u SC_3 load >L_u	SB_2 added to SC_2 , SB_3 added to SC_3	1
27000		exploitation	0
35000		End	

As a UE changes its position, it measures the received signal from the surrounding entities and updates its cell list. During this migration, we examine the network load, aggregate SINR, cell association, and sub-band deployment. The users change their positions at a constant rate, and the last user arrives at $t = 30000$ *T*.

Three sub-bands are considered for this scenario, which are initially assigned as shown in Table II. The exploitation to exploration duration ratio increases as the network stabilizes. During the exploration episodes, the exploration factor is $\epsilon = 1$, while during the exploitation episodes, it is $\epsilon = 0$. The macrocell processes sub-band assignment or release requests at the end of each episode. After the sub-band deployment map changes, the network starts an exploration episode to update the Q-tables of the new map. Algorithm 2 has a fixed exploration factor of $\epsilon = 0.3$ for the complete simulation.

The events and actions that take place in the network are listed in Table II. The resultant user cell association plots, aggregate SINR, and small cell load curves are shown in Fig. 23, Fig. 24, and Fig. 25, respectively. Below we explain those results side by side. The network starts with the aforementioned initial conditions. Then, at $t = 4000$ *T*, the macrocell assigns SB_3 to SC_5 due to high load. Then, another exploration episode starts and ends at $t = 6000$ *T*. As a result, the load of SC_5 drops from 90% to 30%, and the user association becomes balanced between *MC*, SC_5, and SC_1. Most users are located in SC_1 at this point in time. Although this re-assignment was beneficial from load balancing perspective, the aggregate SINR dropped 8 dB from the maximum SINR achieved in the previous exploration episode.

At $t = 8000$ *T*, an exploitation episode ends, and SB_2 is assigned to SC_5 as its load is still above L_u. Consequently, the load drops, but the learning algorithms find its maximum utilities at a less balanced cell association for this new sub-band mapping.

At $t = 10000$ *T*, users start to gradually migrate from the top right corner to the bottom left corner of the simulated area. As a result, we can observe a gradual increase in cell association of SC_3, which is less gradual from 10000T to 25000T.

At $t = 20000$ *T*, SB_2 is released from SC_5 due to the decreased load. Once this happens, a new solution is reached where the user association is better balanced between the macrocell and the small cell tier. This is observed at $t = 20000$ *T*, and $t = 22000$ *T*. This harmony is lost at $t = 25000$ *T* when Sb_3 and Sb_2 were assigned to SC_3 and SC_2 respectively, to decrease their loads. It also interrupts the increase in SINR. The final aggregate SINR is 8 dB less than the maximum reached at $t = 24000$ *T*. Soon after, the users settle at their final positions.

We can conclude from the presented analysis that the RL processes can cope with the changes in the network as they maximize their utility functions in a given sub-band deployment setup. Moreover, The sub-band deployment processes manage to keep cell loads below limits, but it can interrupt the aforementioned utility functions' local maximums. Hence, for future work, there is a need to add intelligence to those processes to choose the best sub-band-to-small cell mapping.

VI. CONCLUSION

Accurate network models are crucial for the development of the standards and finding solutions for heterogeneous network design compromises. This paper sheds light on the complex nature of HetNets and proposes an ABM framework through which a complex dynamic network can be formalized. Agent-based modeling is a computational tool that can build an extensive model incorporating diverse levels of rationality at the agents (network nodes), and can model rule-based behaviors along with machine learning algorithms for different agents. Moreover, local interactions between the nodes create an emergent behavior on the network macro level.

A client-driven system model was proposed wherein the cells are responsible for power and spectrum management based on the users' requests and feedback. Resource allocation and load balancing are the results of the interaction between the cells and the users. Small cells transmit powers were managed by training a Q-learning algorithm on a multi-armed bandit problem to maximize the network's aggregate throughput. Another Q-learning algorithm on a different multi-armed bandit problem drove user sub-band association to maximize the SINR and minimize the user's latency. Both these reinforcement learning algorithms were executed concurrently at macro and small cell levels for efficient sub-band power and RB assignment. Moreover, load

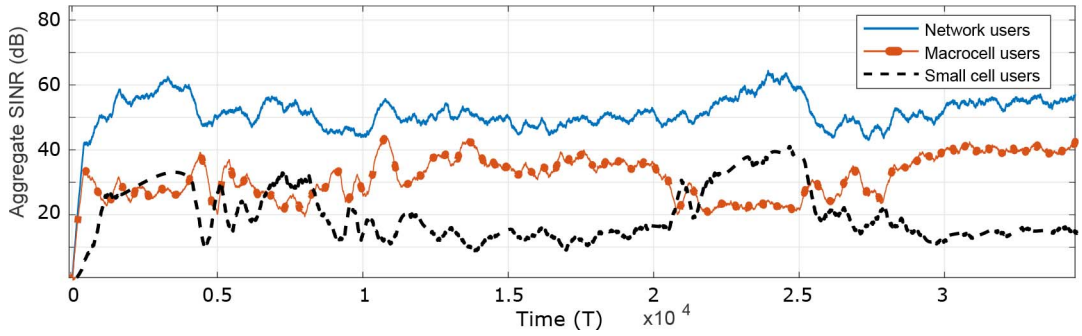


Fig. 29. Aggregate SINR for the proposed scenario versus time.

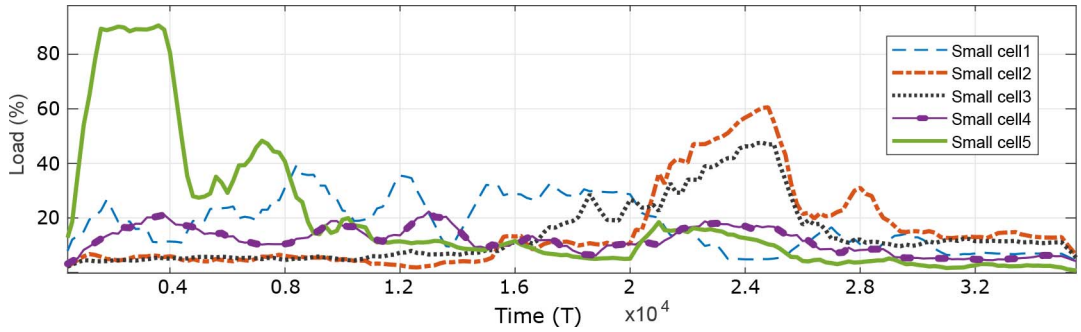


Fig. 30. Small cells loads versus time.

balancing was managed with prescribed rules at the cell level.

In the simulations section, the emergent behavior was shown in the users' distribution within sub-bands and geographical space. The progression of the aggregate SINR by the learning algorithms was analyzed. It was observed that after a certain training duration, the load distribution and aggregate SINR stabilized. We also analyzed the tradeoff between throughput, latency and network utilization, indicating the possible transition points between different reuse cases. A comparison with other literature benchmarks showed that the proposed reinforcement based learning paradigm outperforms in terms of per-user throughput at a minute cost of user latency.

In addition to the aforementioned, a dynamic scenario was considered where the positions of users were changing constantly in time. Consequently, the network traffic was shifting from some cells to others. The dynamic sub-band assignment, cell association, and network throughput were demonstrated and analyzed. We also pointed out that the sub-band assignment processes were acting sub-optimally and that there is room to enhance it by adding more intelligence to the process.

We conclude that agent-based modeling is a versatile tool that should be considered for future complex wireless communication systems development and optimization. It will help with problems, such as moving the intelligence further to the edges of the network, and the design of HetNets assisted with device-to-device communication. Also, the ability to do the computations in a parallel distributed manner enables evaluating the developed solutions on dense networks.

REFERENCES

[1] J. G. Andrews, H. Claussen, M. Dohler, S. Rangan, and M. C. Reed, "Femtocells: Past, present, and future," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 3, pp. 497–508, Apr. 2012.

[2] D. Lopez-Perez, A. Valcarce, G. de la Roche, and J. Zhang, "OFDMA femtocells: A roadmap on interference avoidance," *IEEE Commun. Mag.*, vol. 47, no. 9, pp. 41–48, Sep. 2009.

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

[4] D. Kivanc, G. Li, and H. Liu, "Computationally efficient bandwidth allocation and power control for OFDMA," *IEEE Trans. Wireless Commun.*, vol. 2, no. 6, pp. 1150–1158, Nov. 2003.

[5] K. Yang, N. Prasad, and X. Wang, "An auction approach to resource allocation in uplink OFDMA systems," *IEEE Trans. Signal Process.*, vol. 57, no. 11, pp. 4482–4496, Nov. 2009.

[6] N. Ksairi, P. Bianchi, P. Ciblat, and W. Hachem, "Resource allocation for downlink cellular OFDMA systems—Part I: Optimal allocation," *IEEE Trans. Signal Process.*, vol. 58, no. 2, pp. 720–734, Feb. 2010.

[7] R. Y. Chang, Z. Tao, J. Zhang, and C.-C. J. Kuo, "Multicell OFDMA downlink resource allocation using a graphic framework," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3494–3507, Sep. 2009.

[8] S. K. Kasi, U. S. Hashmi, M. Nabeel, S. Ekin, and A. Imran, "Analysis of area spectral & energy efficiency in a CoMP-enabled user-centric cloud RAN," *IEEE Trans. Green Commun. Netw.*, early access, Jun. 29, 2021, doi: 10.1109/TGCN.2021.3093390.

[9] M. Bennis, M. Debbah, and H. V. Poor, "Ultrareliable and low-latency wireless communication: Tail, risk, and scale," *Proc. IEEE*, vol. 106, no. 10, pp. 1834–1853, Oct. 2018.

[10] S. Cetinkaya, U. S. Hashmi, and A. Imran, "What user-cell association algorithms will perform best in mmWave massive MIMO ultra-dense HetNets?" in *Proc. IEEE 28th Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, 2017, pp. 1–7.

[11] J. G. Andrews, S. Singh, Q. Ye, X. Lin, and H. S. Dhillon, "An overview of load balancing in HetNets: Old myths and open problems," *IEEE Wireless Commun.*, vol. 21, no. 2, pp. 18–25, Apr. 2014.

[12] M. Tayyab, X. Gelabert, and R. Jäntti, "A survey on handover management: From LTE to NR," *IEEE Access*, vol. 7, pp. 118907–118930, 2019.

[13] Y. Liu, C. S. Chen, C. W. Sung, and C. Singh, "A game theoretic distributed algorithm for FeICIC optimization in LTE-A HetNets," *IEEE/ACM Trans. Netw.*, vol. 25, no. 6, pp. 3500–3513, Dec. 2017.

[14] F. Albiero, F. H. P. Fitzek, and M. Katz, "Cooperative power saving strategies in wireless networks: An agent-based model," in *Proc. 4th Int. Symp. Wireless Commun. Syst.*, 2007, pp. 287–291.

[15] J. Zausinova, M. Zoricak, V. Gazda, G. Bugar, and J. Gazda, "An agent-based model of adaptive pricing in HetNets," in *Proc. 3rd Int. Conf. Adv. Inf. Commun. Technol. (AICT)*, 2019, pp. 31–35.

[16] M. Yan, G. Feng, and S. Qin, "Multi-RAT access based on multi-agent reinforcement learning," in *Proc. IEEE Global Commun. Conf.*, 2017, pp. 1–6.

[17] M. Nabeel, U. S. Hashmi, S. Ekin, H. Refai, A. Abu-Dayya, and A. Imran, "SpiderNet: Spectrally efficient and energy efficient data aided demand driven elastic architecture for 6G," *IEEE Netw.*, early access, Sep. 15, 2021, doi: 10.1109/MNET.101.2000635.

[18] S. E. Page, "Uncertainty, difficulty, and complexity," *J. Theor. Polit.*, vol. 20, no. 2, pp. 115–149, 2008.

[19] O. Morgenstern and J. Von Neumann, *Theory of Games and Economic Behavior*. Princeton, NJ, USA: Princeton Univ. Press, 1953.

[20] U. S. Hashmi, S. A. R. Zaidi, and A. Imran, "User-centric cloud RAN: An analytical framework for optimizing area spectral and energy efficiency," *IEEE Access*, vol. 6, pp. 19859–19875, 2018.

[21] U. S. Hashmi, A. Islam, K. M. Nasr, and A. Imran, "Towards user QoE-centric elastic cellular networks: A game theoretic framework for optimizing throughput and energy efficiency," in *Proc. IEEE 29th Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, 2018, pp. 1–7.

[22] 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.

[23] D. Silver *et al.*, "Mastering the game of go without human knowledge," *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[24] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 3826–3839, Sep. 2020.

[25] G. Dulac-Arnold, D. Mankowitz, and T. Hester, "Challenges of real-world reinforcement learning," 2019. [Online]. Available: arXiv:1904.12901.

[26] Z. Xiong, Y. Zhang, D. Niyato, R. Deng, P. Wang, and L. Wang, "Deep reinforcement learning for mobile 5G and beyond: Fundamentals, applications, and challenges," *IEEE Veh. Technol. Mag.*, vol. 14, no. 2, pp. 44–52, Jun. 2019.

[27] L. A. Chylek, L. A. Harris, C.-S. Tung, J. R. Faeder, C. F. Lopez, and W. S. Hlavacek, "Rule-based modeling: A computational approach for studying biomolecular site dynamics in cell signaling systems," *Wiley Interdiscipl. Rev. Syst. Biol. Med.*, vol. 6, no. 1, pp. 13–36, 2014.

[28] L. Gustafsson and M. Sternad, "Consistent micro, macro and state-based population modelling," *Math. Biosci.*, vol. 225, no. 2, pp. 94–107, 2010.

[29] J. H. Holland, *Complexity: A Very Short Introduction*. Oxford, U.K.: Oxford Univ. Press, 2014.

[30] M. M. Waldrop, *Complexity: The Emerging Science at the Edge of Order and Chaos*. New York, NY, USA: Simon Schuster, 1993.

[31] M. Mitchell, *Complexity: A Guided Tour*. Oxford, U.K.: Oxford Univ. Press, 2009.

[32] R. Axtell, "The complexity of exchange," *Econ. J.*, vol. 115, no. 504, pp. F193–F210, 2005.

[33] W. B. Arthur, *The Economy As An Evolving Complex System II*. Boulder, CO, USA: CRC Press, 2018.

[34] M. Batty, *Cities and Complexity: Understanding Cities with Cellular Automata, Agent-Based Models, and Fractals*. Cambridge, MA, USA: MIT Press, 2007.

[35] C. Cioffi-Revilla, "A methodology for complex social simulations," *J. Artif. Soc. Social Simulat.*, vol. 13, no. 1, p. 7, 2010.

[36] I. Chabini, "Discrete dynamic shortest path problems in transportation applications: Complexity and algorithms with optimal run time," *Transp. Res. Rec.*, vol. 1645, no. 1, pp. 170–175, 1998.

[37] O. Woolley-Meza *et al.*, "Complexity in human transportation networks: A comparative analysis of worldwide air transportation and global cargo-ship movements," *Eur. Phys. J. B*, vol. 84, no. 4, pp. 589–600, 2011.

[38] B. Danila, Y. Yu, S. Earl, J. A. Marsh, Z. Toroczkai, and K. E. Bassler, "Congestion-gradient driven transport on complex networks," *Phys. Rev. E, Stat. Phys. Stat. Nonlinear Soft Matter Phys.*, vol. 74, no. 4, 2006, Art. no. 046114.

[39] J. M. Epstein and R. Axtell, *Growing Artificial Societies: Social Science from the Bottom Up*. Washington, DC, USA: Brookings Inst. Press, 1996.

[40] S. Wolfram, *A New Kind of Science*, vol. 5. Champaign, IL, USA: Wolfram Media, 2002.

[41] D. C. Mikulecky, "Complexity science as an aspect of the complexity of science," in *Worldviews, Science and Us*. Hackensack, NJ, USA: World Sci. Publ., 2007, pp. 30–52. [Online]. Available: http://dx.doi.org/10.1142/9789812707420_0003

[42] J. H. Holland, *Hidden Order How Adaptation Builds Complexity*. Reading, MA, USA: Addison Wesley Longman Publ., 1995.

[43] Q. D. Lã, Y. H. Chew, and B.-H. Soong, *Potential Game Theory*. Cham, Switzerland: Springer, 2016.

[44] N. Ksairi, P. Bianchi, P. Ciblat, and W. Hachem, "Resource allocation for downlink cellular OFDMA systems—Part II: Practical algorithms and optimal reuse factor," *IEEE Trans. Signal Process.*, vol. 58, no. 2, pp. 735–749, Feb. 2010.

[45] U. S. Hashmi, S. A. R. Zaidi, A. Darbandi, and A. Imran, "On the efficiency tradeoffs in user-centric cloud RAN," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2018, pp. 1–7.

[46] Z. Han, Z. Ji, and K. J. R. Liu, "Non-cooperative resource competition game by virtual referee in multi-cell OFDMA networks," *IEEE J. Sel. Areas Commun.*, vol. 25, no. 6, pp. 1079–1090, Aug. 2007.

[47] A. H. Arani, M. J. Omidi, A. Mehbodniya, and F. Adachi, "Minimizing base stations' ON/OFF switchings in self-organizing heterogeneous networks: A distributed satisfactory framework," *IEEE Access*, vol. 5, pp. 26267–26278, 2017.

[48] M. Bennis, S. M. Perlaza, P. Blasco, Z. Han, and H. V. Poor, "Self-organization in small cell networks: A reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 12, no. 7, pp. 3202–3212, Jul. 2013.

[49] F. Bernardo, R. Agusti, J. Pérez-Romero, and O. Sallent, "Intercell interference management in ofdma networks: A decentralized approach based onreinforcement learning," *IEEE Trans. Syst., Man, Cybern. C, Appl. Rev.*, vol. 41, no. 6, pp. 968–976, Nov. 2011.

[50] M. Simsek, M. Bennis, and I. Güvenc, "Learning based frequency- and time-domain inter-cell interference coordination in HetNets," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4589–4602, Oct. 2015.

[51] A. Galindo-Serrano, L. Giupponi, and G. Auer, "Distributed learning in multiuser OFDMA femtocell networks," in *Proc. IEEE 73rd Veh. Technol. Conf. (VTC Spring)*, 2011, pp. 1–6.

[52] E. Ghadimi, F. D. Calabrese, G. Peters, and P. Soldati, "A reinforcement learning approach to power control and rate adaptation in cellular networks," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2017, pp. 1–7.

[53] T. Nakamura *et al.*, "Trends in small cell enhancements in LTE advanced," *IEEE Commun. Mag.*, vol. 51, no. 2, pp. 98–105, Feb. 2013.

[54] Q. Ye, B. Rong, Y. Chen, M. Al-Shalash, C. Caramanis, and J. G. Andrews, "User association for load balancing in heterogeneous cellular networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 6, pp. 2706–2716, Jun. 2013.

[55] K. Shen and W. Yu, "Distributed pricing-based user association for downlink heterogeneous cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 32, no. 6, pp. 1100–1113, Jun. 2014.

[56] R. Madan, J. Borran, A. Sampath, N. Bhushan, A. Khandekar, and T. Ji, "Cell association and interference coordination in heterogeneous LTE-A cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 9, pp. 1479–1489, Dec. 2010.

[57] K. Okino, T. Nakayama, C. Yamazaki, H. Sato, and Y. Kusano, "Pico cell range expansion with interference mitigation toward LTE-advanced heterogeneous networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, 2011, pp. 1–5.

[58] V. Chandrasekhar and J. G. Andrews, "Spectrum allocation in tiered cellular networks," *IEEE Trans. Commun.*, vol. 57, no. 10, pp. 3059–3068, Oct. 2009.

[59] W. C. Cheung, T. Q. S. Quek, and M. Kountouris, "Throughput optimization, spectrum allocation, and access control in two-tier femtocell networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 3, pp. 561–574, Apr. 2012.

[60] A. Benmimoune, F. A. Khasawneh, and M. Kadoch, "User association for HetNet small cell networks," in *Proc. 3rd Int. Conf. Future Internet Things Cloud*, 2015, pp. 113–117.

[61] D. Fooladivanda and C. Rosenberg, "Joint resource allocation and user association for heterogeneous wireless cellular networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 1, pp. 248–257, Jan. 2013.

[62] Y. Lin, W. Bao, W. Yu, and B. Liang, "Optimizing user association and spectrum allocation in HetNets: A utility perspective," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 6, pp. 1025–1039, Jun. 2015.

[63] F. Wang, W. Chen, H. Tang, and Q. Wu, "Joint optimization of user association, subchannel allocation, and power allocation in multi-cell multi-association OFDMA heterogeneous networks," *IEEE Trans. Commun.*, vol. 65, no. 6, pp. 2672–2684, Jun. 2017.

[64] Q. Kuang, W. Utschick, and A. Dotzler, "Optimal joint user association and multi-pattern resource allocation in heterogeneous networks," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3388–3401, Jul. 2016.
[65] M. Kim, H. W. Je, and F. A. Tobagi, "Cross-tier interference mitigation for two-tier OFDMA femtocell networks with limited macrocell information," in *Proc. IEEE Global Telecommun. Conf.*, 2010, pp. 1–5.
[66] Junfei Qiu, Y. Yang, and J. Chen, "Load-aware self-organizing spectrum access for small cell networks," in *Proc. IEEE 16th Int. Conf. Commun. Technol. (ICCT)*, 2015, pp. 699–704.
[67] J. Zheng, Y. Wu, N. Zhang, H. Zhou, Y. Cai, and X. Shen, "Optimal power control in ultra-dense small cell networks: A game-theoretic approach," *IEEE Trans. Wireless Commun.*, vol. 16, no. 7, pp. 4139–4150, Jul. 2017.
[68] N. Zhang, S. Zhang, J. Zheng, X. Fang, J. W. Mark, and X. Shen, "QoE driven decentralized spectrum sharing in 5G networks: Potential game approach," *IEEE Trans. Veh. Technol.*, vol. 66, no. 9, pp. 7797–7808, Sep. 2017.
[69] J. Qiu *et al.*, "Hierarchical resource allocation framework for hyper-dense small cell networks," *IEEE Access*, vol. 4, pp. 8657–8669, 2016.
[70] R. Langar, S. Secci, R. Boutaba, and G. Pujolle, "An operations research game approach for resource and power allocation in cooperative femtocell networks," *IEEE Trans. Mobile Comput.*, vol. 14, no. 4, pp. 675–687, Apr. 2015.
[71] Y. Meng, J.-D. Li, H.-Y. Li, and P. Liu, "Graph-based user satisfaction-aware fair resource allocation in OFDMA femtocell networks," *IEEE Trans. Veh. Technol.*, vol. 64, no. 5, pp. 2165–2169, May 2015.
[72] Y.-S. Liang, W.-H. Chung, G.-K. Ni, I.-Y. Chen, H. Zhang, and S.-Y. Kuo, "Resource allocation with interference avoidance in OFDMA femtocell networks," *IEEE Trans. Veh. Technol.*, vol. 61, no. 5, pp. 2243–2255, Jun. 2012.
[73] R. An, X. Zhang, G. Cao, R. Zheng, and L. Sang, "Interference avoidance and adaptive fraction frequency reuse in a hierarchical cell structure," in *Proc. IEEE Wireless Commun. Netw. Conf.*, 2010, pp. 1–5.
[74] F. Hu, K. Zheng, L. Lei, and W. Wang, "A distributed inter-cell interference coordination scheme between femtocells in LTE-advanced networks," in *Proc. IEEE 73rd Veh. Technol. Conf. (VTC Spring)*, 2011, pp. 1–5.
[75] F. Baccelli and B. Błaszczyszyn, *Stochastic Geometry and Wireless Networks*, vol. 1. Boston, MA, USA: Now Publ., 2009.
[76] D. Stoyan and H. Stoyan, "On one of Matérn's hard-core point process models," *Mathematische Nachrichten*, vol. 122, no. 1, pp. 205–214, 1985.
[77] A. Goldsmith, *Wireless Communications*. New York, NY, USA: Cambridge Univ. Press, 2005.
[78] M. Wang, J. Chen, E. Aryafar, and M. Chiang, "A survey of client-controlled HetNets for 5G," *IEEE Access*, vol. 5, pp. 2842–2854, 2017.
[79] T. Lattimore and C. Szepesvári, *Bandit Algorithms*. Cambridge, U.K.: Cambridge Univ. Press, 2020.
[80] J. Vermorel and M. Mohri, "Multi-armed bandit algorithms and empirical evaluation," in *Proc. Eur. Conf. Mach. Learn.*, 2005, pp. 437–448.
[81] H. Robbins, "Some aspects of the sequential design of experiments," *Bull. Trans. Amer. Math. Soc.*, vol. 58, no. 5, pp. 527–535, 1952.
[82] T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Adv. Appl. Math.*, vol. 6, no. 1, pp. 4–22, 1985.
[83] E. Even-Dar, S. Mannor, and Y. Mansour, "PAC bounds for multi-armed bandit and Markov decision processes," in *Proc. Int. Conf. Comput. Learn. Theory*, 2002, pp. 255–270.
[84] S. Mannor and J. N. Tsitsiklis, "The sample complexity of exploration in the multi-armed bandit problem," *J. Mach. Learn. Res.*, vol. 5, pp. 623–648, Dec. 2004.
[85] E. Even-Dar, S. Mannor, and Y. Mansour, "Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems," *J. Mach. Learn. Res.*, vol. 7, pp. 1079–1105, Dec. 2006.
[86] N. Zhang *et al.*, "Software defined networking enabled wireless network virtualization: Challenges and solutions," *IEEE Netw.*, vol. 31, no. 5, pp. 42–49, 2017.
[87] S. T. Arzo, R. Bassoli, F. Granelli, and F. H. P. Fitzek, "Multi-agent based autonomic network management architecture," *IEEE Trans. Netw. Service Manag.*, vol. 18, no. 3, pp. 3595–3618, Sep. 2021.
[88] U. Wilensky and W. Rand, *An Introduction to Agent-Based Modeling: Modeling Natural, Social, and Engineered Complex Systems with NetLogo*. Cambridge, MA, USA: MIT Press, 2015.
[89] W. Cardoso, "5G opportunities and challenges," in *Proc. IEEE 5G Summit Rio*, Nov. 2018.
[90] H. Zhang, Y. Wang, and H. Ji, "Resource optimization-based interference management for hybrid self-organized small-cell network," *IEEE Trans. Veh. Technol.*, vol. 65, no. 2, pp. 936–946, Feb. 2016.

Mostafa Ibrahim (Student Member, IEEE) received the B.Sc. degree in electronics & electrical communication engineering from Ain-Shams University, Egypt, in 2010, the M.Sc. degree in electrical engineering from Istanbul Medipol University, Turkey, in 2017. From 2018 to 2020, he led the implementation and development of a Cell Broadcast Center (Project initiator: BTK Turkish Information and Communication Technologies Authority). He took part in the cell broadcast service deployment in two of Turkey's mobile network operators. He is currently working as a Graduate Research Assistant with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK, USA. His research interests include distributed management in wireless communication systems, beyond 5G waveform design, and air-ground channel modeling and measurements.

Umair Sajid Hashmi (Member, IEEE) received the B.S. degree in electronics engineering from the GIK Institute of Engineering Sciences and Technology, Pakistan, in 2008, the M.Sc. degree in advanced distributed systems from the University of Leicester, U.K., in 2010, and the Ph.D. degree in electrical and computer engineering from the University of Oklahoma, OK, USA, in 2019. During his Ph.D., he worked as a Graduate Research Assistant with the AI4Networks Research Center. He also worked with AT&T, Atlanta, GA, USA, and Nokia Bell Labs, Murray Hill, NJ, USA, on multiple research internships and co-ops. Since Fall 2019, he has been serving as an Assistant Professor with the School of Electrical Engineering and Computer Science, National University of Sciences and Technology, Pakistan, where he is working in the broad area of 5G wireless networks and application of artificial intelligence toward system-level performance optimization of wireless networks, and health care applications. He has published over 15 technical papers in high impact journals and proceedings of IEEE flagship conferences on communications. He has been involved in four NSF funded projects on 5G self organizing networks with a combined award worth of $4 million. He is currently also contributing as a Co-PI on an Erasmus+ consortium project titled "Capacity Building for Digital Health Monitoring and Care Systems in Asia—DigiHealth Asia" with a grant worth of about 1 00 000 Euros and including research teams from five different countries. Since 2020, he has been serving as a Review Editor for IoT and Sensor Networks stream in the *Frontiers in Communications and Networks*.

Muhammad Nabeel received the B.Sc. degree in electrical engineering with majors in communications from the University of Engineering and Technology, Peshawar, Pakistan, in 2011, the M.Sc. degree in electrical engineering from the National University of Computer and Emerging Sciences, Islamabad, Pakistan, in 2013, and the Ph.D. degree in computer science from the Paderborn University, Paderborn, Germany, in 2019. He is currently working as a Postdoctoral Researcher with the Leibniz University Hannover, Hannover, Germany. He is also serving as a Lecturer with the University of Oklahoma, Tulsa, OK, USA. His current research interests include testing and experimentation of beyond 5G networks.

Ali Imran (Senior Member, IEEE) received the B.Sc. degree in electrical engineering from the University of Engineering and Technology Lahore, Pakistan, in 2005, and the M.Sc. degree (Hons.) in mobile and satellite communications and the Ph.D. degree from the University of Surrey, Guildford, U.K., in 2007 and 2011, respectively. He is a Presidential Associate Professor of ECE and the Founding Director of the Artificial Intelligence (AI) for Networks Research Center and TurboRAN Testbed for 5G and Beyond, University of Oklahoma. His research interests include AI and its applications in wireless networks and healthcare. His work on these topics has resulted in several patents and over 100 peer-reviewed articles, including some of the most influential papers in domain of wireless network automation. On these topics, he has led numerous multinational projects, given invited talks/keynotes and tutorials at international forums and advised major public and private stakeholders and cofounded multiple start-ups. He is an Associate Fellow of the Higher Education Academy, U.K. He is also a member of the Advisory Board to the Special Technical Community on Big Data, the IEEE Computer Society.

Sabit Ekin (Senior Member, IEEE) received the B.Sc. degree in electrical and electronics engineering from Eskişehir Osmangazi University, Turkey, in 2006, the M.Sc. degree in electrical engineering from New Mexico Tech, Socorro, NM, USA, in 2008, and the Ph.D. degree in electrical and computer engineering from Texas A&M University, College Station, TX, USA, in 2012. He is a Jack H. Graham Endowed Fellow and an Assistant Professor of Electrical and Computer Engineering with Oklahoma State University (OSU), where he is the Founding Director of OSU Wireless Lab. He has four years of industrial experience as a Senior Modem Systems Engineer with Qualcomm, Inc., where he has received numerous Qualstar awards for his achievements/contributions on cellular modem receiver design. His research interests include the design and analysis of wireless systems including mmWave and terahertz communications in both theoretical and practical point of views, visible light sensing, communications and applications, non-contact health monitoring, and Internet of Things applications.