1-1-2018

A Machine Learning Approach for Power Allocation in HetNets Considering QoS

Roohollah Amiri
Boise State University

Hani Mehrpouyan
Boise State University

Lex Fridman
Massachusetts Institute of Technology

Ranjan K. Mallik
Indian Institute of Technology

Arumugam Nallanathan
Kings College London

See next page for additional authors

 doi: 10.1109/ICC.2018.8422864

A Machine Learning Approach for Power Allocation in HetNets Considering QoS

Roohollah Amiri*, Hani Mehrpouyan*, Lex Fridman ¶,Ranjan K. Mallik†, Arumugam Nallanathan‡, David Matolak§

*Department of Electrical and Computer Engineering, Boise State University - Idaho, USA, {roohollahamiri,hanimehrpouyan}@boisestate.edu

¶Massachusetts Institute of Technology, Cambridge, MA, USA, fridman@mit.edu

†Department of Electrical Engineering, Indian Institute of Technology - Delhi, India, rkmallik@ee.iitd.ernet.in

‡Division of Engineering, Kings College London - London, United Kingdom, nallanathan@ieee.org

§Department of Electrical Engineering, University of South Carolina, Columbia, USA, matolak@cec.sc.edu

***Abstract*—There is an increase in usage of smaller cells or femtocells to improve performance and coverage of next-generation heterogeneous wireless networks (HetNets). However, the interference caused by femtocells to neighboring cells is a limiting performance factor in dense HetNets. This interference is being managed via distributed resource allocation methods. However, as the density of the network increases so does the complexity of such resource allocation methods. Yet, unplanned deployment of femtocells requires an adaptable and self-organizing algorithm to make HetNets viable. As such, we propose to use a machine learning approach based on Q-*learning* to solve the resource allocation problem in such complex networks. By defining each base station as an agent, a cellular network is modeled as a multi-agent network. Subsequently, cooperative Q-*learning* can be applied as an efficient approach to manage the resources of a multi-agent network. Furthermore, the proposed approach considers the quality of service (QoS) for each user and fairness in the network. In comparison with prior work, the proposed approach can bring more than a four-fold increase in the number of supported femtocells while using cooperative Q-*learning* to reduce resource allocation overhead.**

I. Introduction

With an ever increasing density of mobile broadband users, next generation wireless networks (5G) need to support a higher density of users compared to today's networks. One approach for meeting this need is to more effectively share network resources through femtocells [1]. However, lack of guidelines for providing fairness to users and significant interference caused by unplanned deployment of femtocells are important issues that have to be resolved to make heterogeneous networks (HetNets) viable [2]. In this paper reinforcement-learning (more specifically Q-*learning*) as a machine learning method is used in power allocation of a dense femtocell network to maximize the sum capacity of the network while providing quality of service (QoS) and fairness to users.

A. Motivation

Ultra densification is one of the technologies to support the expected huge data traffic required of wireless networks. The idea is to use nested cells comprising small-range low-power access points called femtocells. Femtocells are connected to service providers via a broadband connection (the backhaul connection is supported via DSL or cable). As such, femtocells can be deployed by users anywhere in the cell and the overall cellular network must adapt accordingly. In the last few years, there has been concerted effort by researchers to design different algorithms to optimize the performance of femtocells within next generation wireless network, i.e., 5G. To carry the desired traffic in 5G, most of these methods have aimed for features such as reliability, fairness, and the ability to be distributive, while attempting to maintain a low complexity [3], [4]. However, one important feature that most of these works miss is self-organization and ability to adapt to new conditions of the network.

Reinforcement learning (RL) as a machine learning method, has been developed to optimize an unknown system by interacting with it. The nature of the RL method makes it a perfect solution for scenarios in which statistics of the system continuously change. Further, RL methods can be employed in a distributed manner to achieve even better results in many scenarios [5]. Although RL has been used in many fields, it has been just recently applied in the field of communications with specific applications in areas such as allocation problems [6]–[11], energy harvesting [12], opportunistic spectrum access [13] and other scenarios with distributed nature. With this in mind, this paper tries to apply the RL method to develop a self-organizing dense femtocell network.

B. Prior Work

The selection of a proper reward function in Q-*learning* is essential because an appropriate reward function results in the desired solution to the optimization problem. In this regard, the works in [6]–[9] have proposed different reward functions to optimize power allocation between femto base stations. The works in [6], [7] have used independent learning while the works in [8], [9] have improved the prior art by using cooperative learning. The method in [6] satisfies the QoS of macro users while trying to maximize the sum capacity of the network. However, the QoS and the fairness between femto users (users served by femto base stations) are not considered. In [7], the authors try to improve the throughput of cell-edge users while keeping the fairness between the macro and femto users through a round robin approach. The work in [8] has used cooperative Q-*learning* to maximize the

This research is in part supported by National Science Foundation (NSF) grant on Enhancing Access to Radio Spectrum #1642865 and NASA ULI grant #NNX17AJ94A.

sum capacity of the femto users while keeping the capacity of macro users near a certain threshold. Nevertheless, in both [7] and [8] the QoS of femto users are not taken into consideration. Further, the reward functions in [6]–[8] are not designed for a dense network. The authors in [9] have used the proximity of femto base stations to a macro user as a factor in the reward function, which causes the Q-*learning* method to provide a fair allocation of power between femto base stations. Their proposed reward function keeps the capacity of a macro user above a certain threshold while maximizing the sum capacity of femto users in a dense network. However, by not considering a minimum threshold for the femto users' capacity, the approaches in [6]–[9] fail to support femto users as the density of the network (and consequently interference) increases. Finally, the details of cooperation between femto base stations are not described in [9] and the complexity of their algorithm is not specified.

C. Contribution

In the present work, we propose a new Q-*learning* approach that provides better fairness throughout the whole network. Our contributions can be categorized as follows:

- A new reward function is developed which satisfies the required QoS for each macro and femto user as the density of the network increases.
- New details are provided of how to achieve cooperative Q-*learning* through sharing specific rows of learning tables assigned to femto base stations to carryout power allocation between them. The proposed details clearly indicate that by using cooperative Q-learning the complexity of the machine learning approach can be significantly reduced.
- We carry out a complexity analysis and investigation to indicate the advantage of the proposed Q-learning approach in solving resource allocation in dense HetNets.

D. Organization

The paper is organized as follows. In Section II the system model is presented. Section III introduces the optimization problem and its resulting solution. Section IV presents simulation results. Finally, Section V concludes the paper.

II. System Model

In this paper we consider a single cell of a HetNet that consists of a single macro base station (MBS) and M femto base stations (FBSs). Each FBS serves one user, i.e., a femto user equipment (FUE) and the MBS is assumed to serve a macro user equipment (MUE). We focus on the power allocation in the downlink of a dense HetNet, in which the density results in significant interference. All users transmit in the same spectrum, and narrowband signaling is assumed, or equivalently, results pertain to a single subcarrier of a wideband multicarrier signal. The overall network configuration is presented in Fig. 1. Note that although we consider that both the MBS and FBS server a single user, the proposed approach can easily be adapted to scenarios when more users are served.

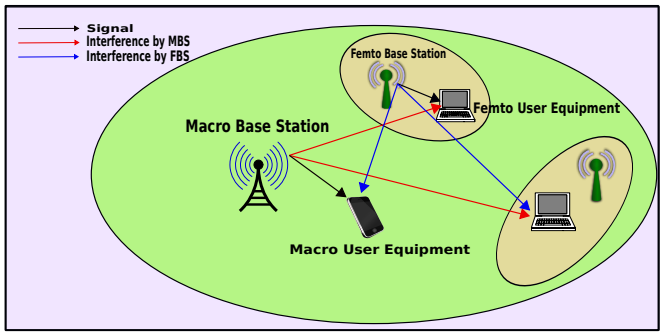


Fig. 1: Femtocell network

The received signal in the downlink at the MUE receiver includes interference from the FBSs and also thermal noise. Hence, the signal-to-interference-noise-ratio (SINR) at the MUE is calculated as follows

$$\mathrm{SINR}_{\mathrm{MUE}} = \frac{P_{\mathrm{BS}} h_{\mathrm{BS,MUE}}}{\sum_{i=1}^{M} P_i h_{\mathrm{FBS}_i,\mathrm{MUE}} + \sigma^2}, \tag{1}$$

where P_{BS} is the power transmitted by MBS, $h_{\mathrm{BS,MUE}}$ is the channel gain from the MBS to the MUE, P_i is the power transmitted by the ith FBS, $h_{\mathrm{FBS}_i,\mathrm{MUE}}$ is the channel gain from the ith FBS to the MUE, and σ^2 denotes the variance of the additive white Gaussian noise.

Similarly, the SINR at the ith FUE is calculated as follows:

$$\mathrm{SINR}_{\mathrm{FUE}_i} = \frac{P_i h_{\mathrm{FBS}_i,\mathrm{FUE}_i}}{P_{\mathrm{BS}} h_{\mathrm{BS,FUE}_i} + \sum_{j=1,j\neq i}^{M} P_j h_{\mathrm{FBS}_j,\mathrm{FUE}_i} + \sigma^2}, \tag{2}$$

where $h_{\mathrm{FBS}_i,\mathrm{FUE}_i}$ is the channel gain between the ith FBS and the ith FUE, $h_{\mathrm{BS,FUE}_i}$ is the channel gain between the MBS and the ith FUE, P_j is the power transmitted by the jth FBS and $h_{\mathrm{FBS}_j,\mathrm{FUE}_i}$ is the channel gain between the jth FBS and the ith FUE. All channel parameters are assumed to be known by the FBS, which is consistent with prior works such as [9], [11]. This is also practically justifiable since the channel information can be fedback to the femtocells through the bakchaul network. Finally, the normalized capacity at any user equipment is calculated as follows

$$C_{\mathrm{MUE}} = \log_2(1+\mathrm{SINR}_{\mathrm{MUE}}). \tag{3}$$

$$C_{\mathrm{FUE}_i} = \log_2(1+\mathrm{SINR}_{\mathrm{FUE}_i}),\ i=1,\ldots,M. \tag{4}$$

III. Problem Formulation and Proposed Solution

In this section, the optimization problem is defined and the Q-*learning* approach to solve this problem is provided. Subsequently, the convergence of the proposed approach and cooperation between femtocells are presented.

A. Optimization Problem

The goal of the optimization problem is to allocate power to the FBSs to maximize the sum capacity of the FUEs, while supporting all users (MUE and FUEs) with their required QoS. By defining $\bar{p} = \{P_1, P_2, ..., P_M\}$ as the vector containing

the transmit powers at the FBSs, the optimization problem can be formulated as

$$\underset{\bar{p}}{\text{maximize}} \quad \sum_{k=1}^{M} C_{\text{FUE}_k} \tag{5a}$$

$$\text{subject to} \quad P_i \leq P_{max},\ i = 1,\ldots,M \tag{5b}$$

$$C_{\text{FUE}_i} \geq \tilde{q}_i,\ i = 1,...,M \tag{5c}$$

$$C_{\text{MUE}} \geq \tilde{q}_{\text{MUE}}. \tag{5d}$$

Here, the objective (5a) is to maximize the sum capacity of the FUEs while providing the MUE with its required QoS in (5d). The first constraint, (5b), refers to the power limitation of every FBS. The terms $\tilde{q}_i$ in (5c) and $\tilde{q}_{\text{MUE}}$ in (5d) refer to the minimum required capacity for the FUEs and the MUE, respectively. Constraints (5c) and (5d) ensure that the QoS is satisfied for all users. Considering (2), (4), and (5), it can be concluded that the optimization in (5) is a non-convex problem for dense HetNets. This follows from the SINR expression in (2) and the objective function (5). More specifically, the interference term due to the neighboring femtocells in the denominator of (2), ensures that the optimization problem in (5a) is not convex. This interference term may be ignored in low density networks but cannot be ignored in dense HetNets consisting of a large number of femtocells.

In the next section, a Q-*learning* based approach to solve this problem is proposed.

B. Reinforcement Learning

The RL problem consists of an environment and a single or multiple agents which based on a chosen policy take actions to interact with the environment. After each interaction, the agent receives a feedback (reward) from the environment and updates its state. An agent can be any intelligent member of the problem, for example in a cellular network it could be an FBS. The goal of this approach is to maximize the cumulative received rewards during an infinite number of interactions. Fig. 2 shows the RL procedure. Most of the RL problems can be considered as Markov Decision Processes (MDPs).

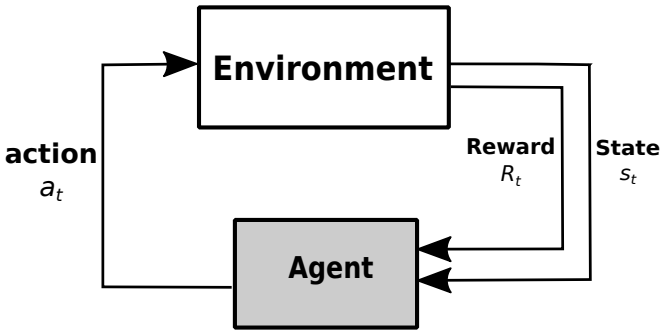


Fig. 2: Reinforcement Learning, Agent and Environment.

C. Proposed Q-learning Approach

Q-*learning* is a model-free RL method that attacks MDP problems with dynamic programming [14]. Q-*learning* can be considered as a function approximator in which the values of the approximator, Q, depend on the state (x_t) and action (a_t) at time step t. The dynamic programming equation for computing a function approximator Q (also known as Bellman equation) is as follows

$$Q(x_t, a_t) = \max_a(E[R_t + \gamma Q(x_{t+1}, a)]), \tag{6}$$

where E denotes the expectation operator and R_t is the received reward at time step t and $0 \leq \gamma \leq 1$ is the discount factor. Eq. (6) has a unique strictly concave solution and the solution is approached by limit as $t \to \infty$ by iterations [15].

The novelty of Q-*learning* is attributed to the use of *temporal-difference* (TD) to approximate a Q-function [16], [17]. The simplest form of *one-step* Q-*learning* approach, is defined by

$$Q(x_t, a_t) \leftarrow (1-\alpha)Q(x_t, a_t) + \alpha \max_a(R_t + \gamma Q(x_{t+1}, a)), \tag{7}$$

where α is the learning rate. Algorithm 1 specifies the Q-*learning* in procedural form [16].

Algorithm 1 Q-Learning algorithm

```
Initialize Q(x_t, a_t) arbitrarily
for all episodes do
    Initialize x_t
    for all steps of episode do
        Choose a_t from set of actions
        Take action a_t, observe R_t, x_{t+1}
        Q(x_t, a_t) ← (1−α)Q(x_t, a_t) + α max_a(R_t + γQ(x_{t+1}, a))
        x_t ← x_{t+1};
    end for
end for
```

In the context of a femtocell network, FBS acts as an agent in the Q-*learning* algorithm, which means each FBS runs Algorithm 1, separately. The Q-*learning* approach consists of three main parts as follows

*1) **Actions**:* Each FBS chooses its transmit power from a set $A = \{a_1, a_2, ..., a_{N_{power}}\}$, which covers the space between P_{min} and P_{max}. In general, there is no particular information from environment, so the FBS chooses actions with the same probability. Therefore, equal step sizes are chosen between P_{min} and P_{max}.

*2) **States**:* States are chosen based on the vicinity of the FBS to the MBS and the MUE. In order to specify the state of an FBS, we define two parameters for each FBS:

- $D_{\text{MBS}} \in \{0, 1, 2, ..., N_1\}$: The value of D_{MBS} defines the location of an FBS compared to N_1 rings centered on the MBS. The radius of layers are d_{BS1}, d_{BS2}, ... , $d_{\text{BS}_{N_1}}$.
- $D_{\text{MUE}} \in \{0, 1, 2, ..., N_2\}$: The value of D_{MUE} defines the location of an FBS compared to N_2 rings centered on the MUE. The radius of layers are d_{FBS1}, d_{FBS2}, ... , $d_{\text{FBS}_{N_2}}$.

By considering the above definitions, the state of the FBS i at time step t is defined as $s_t^i \in \{D_{\text{MBS}}, D_{\text{MUE}}\}$. Each FBS, constructs a table for itself, which comprises all possible states as its rows and actions as its columns called a Q-table. By the state definition, in the proposed model, the FBS state

remains constant as long as its location is fixed. This feature brings an advantage in sharing Q-tables between the FBSs with the same state, where only a single row of each FBS's Q-table needs to be shared.

3) ***Reward****:* The definition of the reward function is essential because it targets the objective of the Q-*learning* method. According to the optimization problem in (5), the goal of the optimization process is to maximize the sum capacity of femto users in the network while maintaining QoS for each one of them. In order to translate this objective to a reward function, the following points are taken into account:

- The objective of the optimization problem is to maximize the capacity of the network, so a higher capacity for FUE or MUE should result in a higher reward.
- To satisfy the QoS requirements of users, capacity deviation of users from their required QoS ($\tilde{q}_i$ or $\tilde{q}_{\mathrm{MUE}}$) should decrease the reward.

By considering the above points, the proposed reward function (RF) for the ith FBS at time step t is defined as

$$R_t^i = \underbrace{\beta_i}_{(d)} \underbrace{C_{\mathrm{FUE}_{i,t}} C^2_{\mathrm{MUE}_t}}_{(a)} - \underbrace{\frac{1}{\beta_i}}_{(e)} \underbrace{(C_{\mathrm{MUE}_t} - \tilde{q}_{\mathrm{MUE}})^2}_{(b)} - \underbrace{\left(C_{\mathrm{FUE}_{i,t}} - \tilde{q}_i\right)^2}_{(c)}, \quad (8)$$

which is derived based on the above points. In (8), $C_{\mathrm{FUE}_{i,t}}$ and C_{MUE_t} are the capacities of the ith FUE and the MUE at time step t, respectively. According to (8), the reward function comprises three main terms *(a), (b), and (c)*. The first term (a) implies that a higher capacity for the ith FUE or the MUE results in a higher reward. In the same term, the capacity of the MUE is squared. The power assigned to C_{MUE_t} in (a), is supported by our simulation in Section IV and to also give a higher priority to MUE with respect to the FUE by allocating a higher reward to its capacity value. The terms (b) and (c) are deviations of the ith FUE and the MUE from their required threshold. Hence, terms (b) and (c) are reduced from term a to decrease the reward. Terms (d) and (e) provide fairness to the algorithm. β_i (term d) is defined as the distance of the ith FBS to the MUE normalized by d_{th}. d_{th} is a constant distance, which indicates whether the FBS is in the vicinity of the MUE or not. For example, if the distance of the ith FBS and the MUE is less than d_{th}, the interference of the ith FBS affects the MUE more than any other FBS with distance more than d_{th}. Then the ith FBS should be given less reward, which means reducing term (a) by multiplying it by β_i (or (d)) and increasing term b by multiplying it by the inverse of β_i (or (e)).

D. Convergence

According to [14], in Algorithm 1, if all actions are repeatedly sampled in all states, (7) will be updated until the value of Q converges to the optimal value (Q^*) with probability 1. In practice, the number of updates is limited. Hence, the final value of Q may be suboptimal. Q-*learning* itself is a greedy policy since it finds the action which derives the maximum Q-*value* on each iteration. Greedy policies have the disadvantage of being vulnerable to environmental changes, and they can be trapped or biased in a limited search area which causes the algorithm to converge slower. One reasonable solution is to act greedy with probability $1-\varepsilon$ (exploiting) and act randomly with probability ε (exploring). Different values for ε provide a trade-off between exploitation and exploration. Algorithms that try to explore and exploit fairly are called SARSA or ε-greedy [16]. In [16] it is shown that in a limited number of iterations, the ε-greedy policy has a faster convergence rate and closer final value to the optimal one, compared to the greedy policy. As such, the ε-greedy policy has been used in the rest of this paper. Further, our investigations show that ε values of 0.1 or 0.01 provide a reasonable trade off between exploitation and exploration.

E. Cooperative Q-Learning

The time complexity of an RL algorithm depends on three main factors: the state space size, the structure of states, and the primary knowledge of the agents [5], [18]. If priori knowledge is not available to an agent or if environment changes and the agent has to adapt, the search time can be excessive [5]. Considering the above, decreasing the effect of state space size on learning rate and providing agents with priori knowledge has been a subject of significant research [5], [18]–[20].

One approach to deal with this problem is by transferring information from one agent to another instead of expecting agents to discover all the necessary information. In fact, by using a multi agent RL network (MARL), agents can communicate and share their experiences with each other, and learn from one another [5]. The reason that cooperation can reduce the search time for RL algorithms can be attributed to the different information that the agents can gather regarding their experiences in the network. By sharing information between experienced and new agents, a priori knowledge is provided for new agents to reduce their search time. It is worth mentioning that even in a MARL network that consists of a large number of new agents, cooperation and information sharing among these agents can reduce search time for the optimum power allocation solution [5]. Another reason why cooperation enhances search speed is the inherent parallelism in cooperation between agents [5], [20]. In other words, by sharing their information, the agents search different choices in parallel which decreases the search time greatly. In [5] it is shown that by intelligent sharing of information between agents, search time can be executed as a linear function of the state-space size.

Sharing Q-values in MARL networks for resource allocation and management is still an open research problem. The main challenge lies in the fact that the agents must be able to acquire the required information from the shared Q-values [18]. As a result, in a large MARL network it is not yet clear what Q-values must be shared among the agents to reduce search time and reach the optimum power allocation solution. Moreover, cooperation comes at the cost of communication. Agents can share their information to help each other to learn faster while adding more overhead to the network by passing on their Q-values through the backhaul network. Nevertheless, it is important to note that these Q-values can be significantly quantized to reduce this overhead.

In a femtocell network, each FBS gathers information regarding the network. The nature of this information for each FBS may be different and directly related to its active time in the network. Accordingly, we propose a cooperative Q-learning approach where the Q-tables of FBSs that are in the same state, i.e., the FBSs that are located in the same vicinity (rings) relative to the MBS and the MUE, are shared with one another. The latter is proposed since our results show that only the FBSs with the same state have useful information for one another. Moreover, the proposed approach reduces the communication overhead among the FBSs.

Accordingly, the proposed method for the femtocell network consists of two modes: individual learning and cooperative learning. The individual learning starts by initializing the Q-values of a small subset of FBSs, e.g., four, to zero. These FBSs execute the proposed RL algorithm independently. After convergence, new agents are added to the problem one by one and cooperative Q-*learning* takes place. In this mode, the MARL network consists of experienced FBSs and one new FBS. The new FBS takes its priori knowledge from the FBSs with the same state and all FBSs execute the RL algorithm. The FBSs with the same state share their Q-tables (just one row) after each iteration. To form a new Q-table from the shared Q-tables, we have used the method in [19], where the shared Q-tables are averaged over. Although this method is suboptimal [18] and to perform accurate sharing, a weighted averaging between Q-tables should be used, we have chosen to select the simple averaging method to achieve a lower overall complexity.

IV. Simulation Results

In this section the simulation setup is detailed and then the results of the simulations are presented.

A. Simulation Setup

A femtocell network is simulated with a single MBS, one MUE, and M number of FBSs, where each FBS supports one FUE, see Fig. 3.

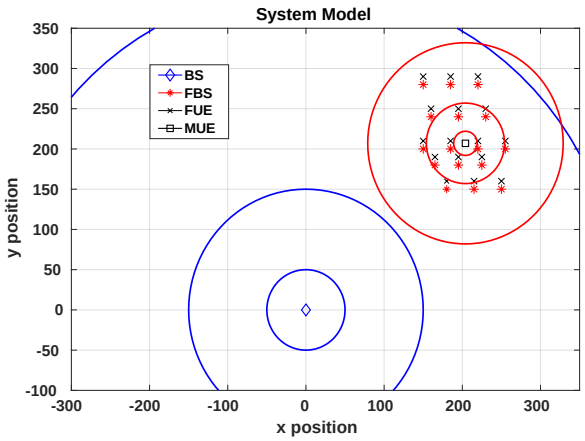


Fig. 3: Locations of MBS, FBSs, MUE and FUEs.

To simulate a residential neighborhood, the FBSs are located 35 m apart from each other. Each FUE is located in a 10 m radius from its serving FBS. To simulate a high interference scenario in a dense network, the MUE is located among 15 number of FBSs. The locations of the MBS, FBSs, and MUE are shown in Fig. 3. In these simulations the number of layers around the MBS and the MUE are assumed to be three ($N_1 = N_2 = 3$). Although, as the density increases, more rings with smaller diameters can be used to more clearly distinguish between the FBSs. The blue and red rings indicate the states of the FBSs with respect to the MBS and the MUE, respectively.

It is assumed that the FBS and the MBS are both operating over the same channel bandwidth at $f = 2.4$ GHz. The path loss model of the link between the MBS and the MUE, and the one between the FBS and its serving FUE is given by

$$PL = PL_0 + 10n\log_{10}(d/d_0), \tag{9}$$

where PL_0 is the constant path loss value, and n is the path loss exponent. The parameters of the model are set to: d_0 = 5 m, PL_0 = 62.3 dB and n = 4 [21], as an example of a model for a residential area. The path loss of the link between each FBS and the MUE, and the link between each FBS and the FUE of other FBSs are modeled using an empirical indoor-to-outdoor model suitable for femtocells from [22]. Using (6), (7), and Table I from [22], the path loss can be written as

$$PL = PL_i + PL_0, \tag{10}$$

$$PL_i = -1.8f^2 + 10.6f + 6.1, \tag{11}$$

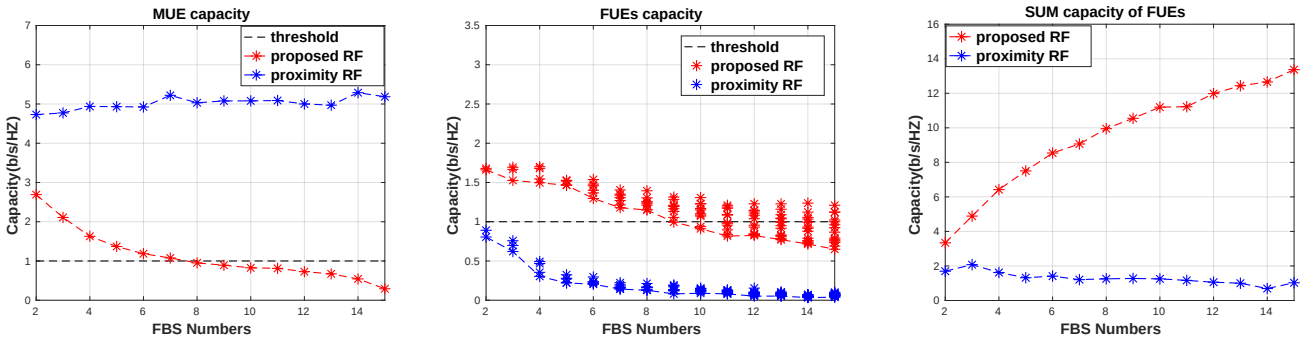
$$PL_0 = 62.3 + 32\log_{10}(d/5), \tag{12}$$

where f denotes the operating frequency in GHz. The remaining parameters are given in Table I.

TABLE I: Simulation Parameters

Parameter	Value	Parameter	Value
Pmin	-20 dBm	Pmax	25 dBm
N_{power}	31	Step Size	1.5 dBm
d_{bs_1}	50 m	d_{fbs_1}	15 m
d_{bs_2}	150 m	d_{fbs_2}	50 m
d_{bs_3}	400 m	d_{fbs_3}	125 m
dth	25 m		

The QoS requirements for the MUE and FUEs are defined as the capacity needed for each to support their user's application. For simulation the values of $\tilde{q}_{MUE} = 1$ (b/s/Hz) and $\tilde{q}_i = 1$ (b/s/Hz), $i = 1,..,15$ are considered for the MUE and FUEs, respectively. By knowing the MAC layer parameters, the values of the required QoS can be calculated using [23, Eqs. (20) and (21)]. To perform Q-*learning* the following values are used: learning rate $\alpha = 0.5$, discount factor $\gamma = 0.9$. The ε-greedy algorithm is used for the first 80% of iteration with random $\varepsilon = 0.1$ and the maximum number of iterations is set to 50,000. The agents are randomly added to the network. For each number of agents, the algorithm goes through all iterations and the agents share their Q-tables according to the proposed algorithm in Section III-E.



a: MUE capacity. b: Capacity of FUEs. c: Sum capacity of FUEs

Fig. 4: Performance of the proposed reward function.

B. Results

In this section we show the results of the proposed method compared to the results of the approach in [9]. The method in [9] is based on a proximity based reward function, which we call it *proximity RF*. To have a fair comparison between the two algorithms, three measurements are plotted: the MUE capacity, the capacity of each one of the FUEs for every number of FBSs operating in the network, and the sum capacity of the FUEs. As it is shown in Fig. 3, the position of the MUE is an example of a dense network which results in a high interference scenario. The results are shown in Figs. 4a, 4b, and 4c, which indicate that the approach in [9] is successful in satisfying the required QoS of the MUE for all number of FBSs (Fig. 4a), while it fails to support the FUEs as the density of the network increases (Fig. 4b). In fact, after adding the sixth FBS for the approach in [9], the FUE capacity decreases to almost zero. Hence, the QoS of the MUE is satisfied at the expense of no service for some of the FUEs. However, Fig. 4a shows that the proposed approach satisfies the QoS for the MUE and the FUEs up to the point where 8 FBSs are operating simultaneously in close vicinity of the MUE. Further, after adding more FBSs the capacity of the MUE does not fall to zero and it is still close to the required threshold whenever 11 FBSs are operating in close vicinity. At the same time, the majority of FUEs are still meeting their required QoS. According to Fig. 4b the capacity of FUEs are fairly close to each other regardless of their position, which demonstrates the algorithm's fairness. Finally, Fig. 4c shows the sum capacity of the network which has an increasing trend for all number of FBSs and is consistently higher than that of the approach in [9].

C. Convergence Analysis

As it is noted in Section IV-A, the maximum number of iterations to run the algorithm is set to $50,000$, although the algorithm always converges before this number is reached. Fig. 5 provides the number of iterations that it takes for both algorithms to converge with respect to the number of active FBSs in the network. As shown in Fig. 5, the proposed algorithm requires close to 4×10^4 iterations whenever 13 FBSs are active in the network. In contrast, the number of iterations is always lower than the algorithm in [9]. The order of required iterations for convergence is $4\times10^4\approx2^{15}$, which is an extremely small portion of total number of iterations required for exhaustive search, i.e, $32^{15}=2^{75}$.

To provide a better understanding of time duration of the proposed algorithm, Fig. 6 shows the actual run time of the proposed algorithm on a regular processor.

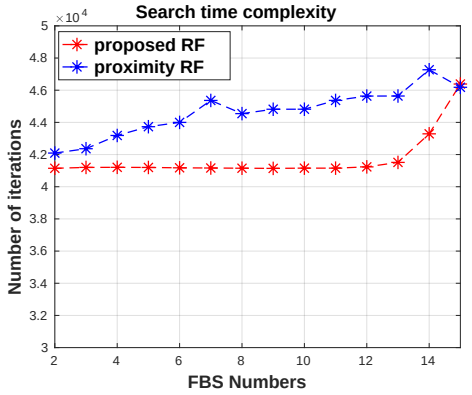


Fig. 5: Average number of iterations for the algorithms to converge.

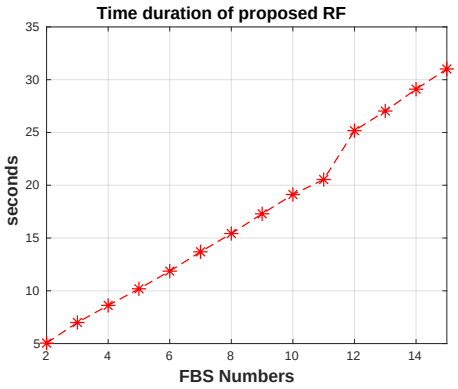


Fig. 6: Average run time of the proposed algorithm on Intel(R) Core(TM) i5-4590 CPU @ 3.30GHz.

D. Fairness

To provide measurement for fairness, Jain's fairness index [24] is used. In this method fairness is defined as $f(x_1,x_2,...,x_n)=\frac{\left(\sum_{i=1}^{n}x_i\right)^2}{n\sum_{i=1}^{n}x_i^2}$, in which $0\leq f(x_1,x_2,...,x_n)\leq 1$, here equality to 1 is achieved when all the FUEs have the

same capacity. As it is shown in Fig. 7, the fairness index is close to one whenever 13 FBSs are active in the network.

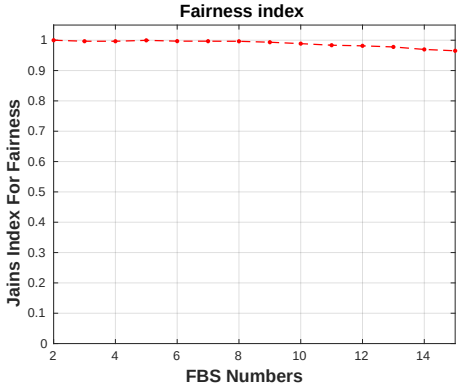


Fig. 7: Jain's fairness index as a function of the number of FBSs.

E. Complexity Analysis

In Section III-E the parameters that affect the search time of Q-*learning* method are discussed. In a single FBS network (single agent network), with a finite number of iterations, and using the ε-greedy policy with fixed ε, and $|S|$ as the size of the state-space, the search time is upper bounded by $\mathcal{O}(|S|\log(|S|)\log(1/\varepsilon)/\varepsilon^2)$ [25]. The cooperation method that is proposed in Section III-E for femtocell networks is a special case of a learning with an external critic (LEC) method proposed by [5] for MARL networks. According to [5] the expected time needed for convergence is upper bounded by $\mathcal{O}(|S|N_{power}\log(1/\varepsilon)/\varepsilon^2)$, where N_{power} is the number of actions (power levels) in each iteration from which the FBS can choose. N_{power} is linear in state-space size. On the other hand, the optimal exhaustive search has a time complexity of $\mathcal{O}(N_{power}^M)$, where M is the number of FBSs in the network.

V. Conclusion

The results of this paper show the application of machine learning to address resource allocation in dense HetNets. In a high interference scenarios, the power optimization in HetNet is a non-convex problem that cannot be solved with reasonable complexity. On the other hand, the proposed method as a distributed approach can solve the optimization problem in dense HetNets, while significantly reducing complexity. Our simulations show that while reducing the overall complexity of resource allocation, the proposed approach serves all users for up to 8 femtocells whereas the approach in [9] was unable to satisfy the FUEs at the expense of satisfying only the MUE. Future work will explore different methods of sharing information to obtain an optimal information sharing algorithm between agents.

References

[1] H. Mehrpouyan, M. Matthaiou, R. Wang, G. K. Karagiannidis, and Y. Hua, "Hybrid millimeter-wave systems: a novel paradigm for hetnets," *IEEE Commun. Mag.*, vol. 53, no. 1, pp. 216–221, January 2015.

[2] Z. Lu, T. Bansal, and P. Sinha, "Achieving user-level fairness in open-access femtocell-based architecture," *IEEE Trans. Mobile Comput.*, vol. 12, no. 10, pp. 1943–1954, Oct 2013.

[3] C. Niu, Y. Li, R. Q. Hu, and F. Ye, "Fast and efficient radio resource allocation in dynamic ultra-dense heterogeneous networks," *IEEE Access*, vol. 5, pp. 1911–1924, 2017.

[4] V. N. Ha and L. B. Le, "Fair resource allocation for ofdma femtocell networks with macrocell protection," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 3, pp. 1388–1401, March 2014.

[5] S. D. Whitehead, "A complexity analysis of cooperative mechanisms in reinforcement learning." in *AAAI*, 1991, pp. 607–613.

[6] A. Galindo-Serrano and L. Giupponi, "Distributed Q-learning for interference control in OFDMA-based femtocell networks," in *Proc. IEEE Veh. Technol. Conf.*, May 2010, pp. 1–5.

[7] B. Wen, Z. Gao, L. Huang, Y. Tang, and H. Cai, "A Q-learning-based downlink resource scheduling method for capacity optimization in LTE femtocells," in *Proc. IEEE. Int. Comp. Sci. and Edu.*, Aug 2014, pp. 625–628.

[8] H. Saad, A. Mohamed, and T. ElBatt, "Distributed cooperative Q-learning for power allocation in cognitive femtocell networks," in *Proc. IEEE Veh. Technol. Conf.*, Sept 2012, pp. 1–5.

[9] J. R. Tefft and N. J. Kirsch, "A proximity-based Q-learning reward function for femtocell networks," in *Proc. IEEE Veh. Technol. Conf.*, Sept 2013, pp. 1–5.

[10] Z. Feng, L. Tan, W. Li, and T. A. Gulliver, "Reinforcement learning based dynamic network self-optimization for heterogeneous networks," in *IEEE Pacific Rim Conf. on Commun., Comp. and Signal Process.*, Aug 2009, pp. 319–324.

[11] A. Galindo-Serrano and L. Giupponi, "Self-organized femtocells: A fuzzy Q-learning approach," *Wirel. Netw.*, vol. 20, no. 3, pp. 441–455, Apr. 2014. [Online]. Available: http://dx.doi.org/10.1007/s11276-013-0609-6

[12] M. Miozzo, L. Giupponi, M. Rossi, and P. Dini, "Distributed Q-learning for energy harvesting heterogeneous networks," in *Proc. IEEE. Int. Commun. Workshop*, June 2015, pp. 2006–2011.

[13] B. Hamdaoui, P. Venkatraman, and M. Guizani, "Opportunistic exploitation of bandwidth resources through reinforcement learning," in *IEEE GLOBECOM*, Nov 2009, pp. 1–6.

[14] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3, pp. 279–292, 1992. [Online]. Available: http://dx.doi.org/10.1007/BF00992698

[15] L. Ljungqvist and T. J. Sargent, *Recursive Macroeconomic Theory, Third Edition*. Cambridge, MA, USA: MIT Press, 2012.

[16] R. S. Sutton and A. G. Barto, *Introduction to Reinforcement Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.

[17] C. Watkin, "Learning from delayed rewards," Ph.D. dissertation, King's College, Cambridge, 1989.

[18] M. N. Ahmadabadi and M. Asadpour, "Expertness based cooperative Q-learning," *IEEE Trans. Syst., Man, Cybern. B*, vol. 32, no. 1, pp. 66–76, Feb 2002.

[19] M. Tan, "Multi-agent reinforcement learning: Independent vs. cooperative agents," in *In Proc. ICML*. Morgan Kaufmann, 1993, pp. 330–337.

[20] L. Buoniu, R. B. *hats*ka, and B. D. Schutter, "A comprehensive survey of multiagent reinforcement learning," *IEEE Trans. Syst., Man, Cybern. C*, vol. 38, no. 2, pp. 156–172, March 2008.

[21] E. Hossain, V. K. Bhargava, and G. P. Fettweis, *Green Radio Communication Networks*, 1st ed. New York, NY, USA: Cambridge University Press, 2012.

[22] A. Valcarce and J. Zhang, "Empirical indoor-to-outdoor propagation model for residential areas at 0.9 -3.5 GHz," *IEEE Antennas Wireless Propag. Lett.*, vol. 9, pp. 682–685, 2010.

[23] C. C. Zarakovitis, Q. Ni, D. E. Skordoulis, and M. G. Hadjinicolaou, "Power-efficient cross-layer design for OFDMA systems with heterogeneous qos, imperfect csi, and outage considerations," *IEEE Trans. Veh. Technol.*, vol. 61, no. 2, pp. 781–798, Feb 2012.

[24] R. Jain, D. Chiu, and W. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," *CoRR*, 1998. [Online]. Available: http://arxiv.org/abs/cs.NI/9809099

[25] M. J. Kearns and S. P. Singh, "Finite-sample convergence rates for Q-learning and indirect algorithms," in *Advances in Neural Information Processing Systems 11*. MIT Press, 1999, pp. 996–1002.