Online Learning for Energy Saving and Interference Coordination in HetNets

Jose A. Ayala-Romero, Juan J. Alcaraz, Andrea Zanella, *Senior Member, IEEE*, and Michele Zorzi, *Fellow, IEEE*

***Abstract*— In heterogeneous cellular networks (HetNets), switching OFF small cells under low user traffic periods has been proved to be an effective energy saving strategy. However, this strategy has strong interactions with interference coordination (IC) mechanisms, making it convenient to address both tasks simultaneously. The motivation of this paper is to develop a self-optimization algorithm capable of jointly controlling energy saving and IC mechanisms using an online learning approach. Our proposal is based on a contextual bandit formulation that, among other challenges, implies discovering the most energy-efficient control actions while satisfying a predefined level of Quality of Service (QoS) for the users. We propose a two-level framework comprising a global controller, in charge of a group of macro cells, and multiple local controllers, one per macro cell. The global controller implements a novel algorithm, referred to as the Bayesian Response Estimation and Threshold Search (BRETS), that is capable of learning, for each control action, its feasibility boundaries in terms of QoS and its energy consumption as a function of the aggregated user traffic. The algorithm comes with a bound on its expected convergence time. The local controllers translate the control actions learned by the global controller into local decisions. Our numerical results show that BRETS is only 1% less efficient than an ideal *oracle* policy, clearly outperforming other benchmark algorithms.**

***Index Terms*— Online learning, conextual multi-armed bandit, green networks, heterogeneous networks, interference coordination.**

I. Introduction

A PROMISING step towards increasing the network capacity is based on the dense deployment of small cells, thus realizing the so-called Heterogeneous Networks (HetNets), considered one of the key technologies in 5G networks [1]. Nevertheless, the densification of HetNets poses two main challenges: the increment of the energy consumption due to the larger number of cells, and the inter-cell interference from the macro to the small cells. For the first issue, the most effective energy saving (ES) strategy is to switch off under-utilized small cell stations, if this is possile, while avoiding noticeable degradation of the Quality of Service (QoS) of the network users [2]. For the second one, interference coordination (IC) mechanisms allow the network to optimize its wireless access capacity by adjusting the interference level and the radio resource allocation between macro and small cells [3]. However, these two mechanisms are intertwined: switching on/off small cells yields a reassignment of the user equipments (UEs) to the active base stations and, consequently, a change in the inter-cell interference. In turn, the IC control mechanisms will adjust the transmission resources assigned to the different UEs and their transmit power, thus impacting on the overall energy consumption of the system. To exemplify the concept, we report in the upper row of Fig. 1 the heat maps of the overall energy consumption and in the lower row the corresponding heat maps of the fraction of UEs with satisfactory QoS (i.e., sufficiently high throughput in this example), when varying the number of active picocells from 0 to 6 (left to right). For each number of active picos, the heat maps are obtained by changing the configuration of the Almost Blank Subframe (ABS) and the Cell Range Expansion (CRE) bias, which are two IC mechanisms that will be explained in detail later. We can observe how energy efficiency and QoS satisfaction jointly depend on the number of active picos and the setting of the IC parameters. Therefore, addressing both tasks simultaneously has been shown in [4] to improve ES while maintaining a desired level of QoS.

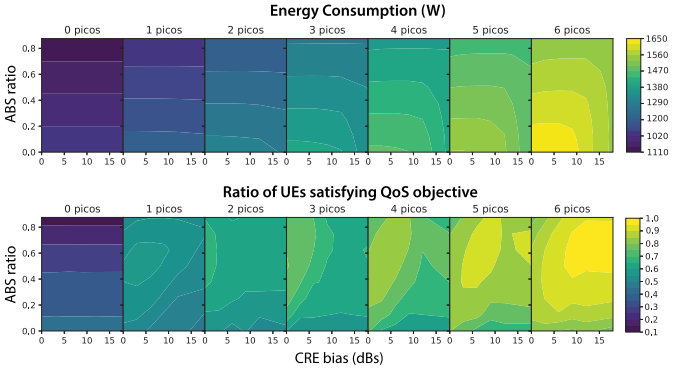

Fig. 1. Energy consumption and QoS satisfaction values in a macro cell for different combinations of IC and ES controls under the same traffic conditions. IC controls are given by the Almost Blank Subframe (ABS) ratio, and Cell Range Expansion (CRE) bias, while ES controls are determined by the number of active picos in the macro cell.

Manuscript received July 16, 2018; revised December 21, 2018; accepted March 3, 2019. Date of publication March 11, 2019; date of current version May 15, 2019. This work was supported in part by the Grant AEI/FEDER TEC2016-76465-C2-1-R (AIM) and in part by the program Supporting Talent in Research@University of Padua: STARS Grants, through the project Cognition-Based Networks: Building the Next Generation of Wireless Communications Systems Using Learning and Distributed Intelligence. The work of J. A. Ayala-Romero was supported by the Grant FPU14/03701. *(Corresponding author: Jose A. Ayala-Romero.)*

J. A. Ayala-Romero and J. J. Alcaraz are with the Department of Information and Communications Technologies, Technical University of Cartagena, 30202 Cartagena, Spain (e-mail: josea.ayala@upct.es; juan.alcaraz@upct.es).

A. Zanella and M. Zorzi are with the Department of Information Engineering, University of Padova, 35131 Padua, Italy (e-mail: zanella@dei.unipd.it; zorzi@dei.unipd.it).

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

Digital Object Identifier 10.1109/JSAC.2019.2904362

On the other hand, the self-optimization of network control tasks is also considered a key feature of 5G networks [5], [6], enabling the network to autonomously find the most efficient configuration for each functionality without need for human intervention. To attain this objective, an online learning approach is especially interesting since it aims at learning the most effective configurations using observations taken from the real operating network. Offline learning algorithms, in contrast, need to be trained before their implementation in the real system, requiring either a simulated model of the network, or a data set obtained from the system, involving additional costs. Moreover, the resulting control policies would be effective as long as the real network behaves as predicted during the offline training phase. Inaccuracies of the simulation model, biases in the data set, or changes in the network, might reduce the effectiveness of the policy. Notably, although there exist previous online learning proposals for IC [7], [8] and ES [9] separately, this approach has not yet been applied to the joint control of IC and ES, which is the motivation of this work.

Designing an online learning scheme for IC-ES is a challenging task because of several reasons. First, the performance degradation associated to the exploration of poor performing controls should be kept at minimum. Second, the dimension of the control space can be very large because in principle it comprises all the combinations of IC and ES control values. Third, the optimal IC-ES configuration depends on the user traffic intensity in the network, which changes over time. Fourth, our problem involves keeping the QoS perceived by the users above a certain value.

The online learning approach fits a multi-armed bandit (MAB) problem where the state of the system *contexts* changes independently of the controls, i.e., a contextual bandit problem [10], whose objective is to learn the best configuration at each possible intensity of the network traffic (context). Nevertheless, while classical contextual bandits only consider one performance metric, we must consider two: energy saving and QoS fulfillment. The latter is introduced in the problem as a constraint, resulting in a novel variant referred to as *constrained* contextual bandit problem. Besides, the contexts take values from a continuous set, unlike the usual case, where a finite set is considered.

Our application scenario is an HetNet composed of a set (or cluster) of contiguous macro eNodeBs (eNBs), and multiple pico eNBs overlapping the coverage area of the macro eNBs. The IC functionality considered is the enhanced Inter Cell Interference Coordination (eICIC) mechanism proposed by the 3GPP for LTE-A Networks [3]. Our framework comprises two decision levels, global and local, corresponding to the cluster and the individual macro eNBs, respectively. At the global level, a centralized entity (global controller) makes IC-ES decisions for the whole cluster and obtains performance observations from the eNBs of the cluster. The online learning algorithm operates in the global controller, allowing it to progressively learn how to select better controls based on the history of past decisions and observations. At the local level, the local controllers decide how to effectively translate each global configuration prescribed by the global controller into a local configuration for each macro cell. This approach, introduced in our earlier work ClassMAB [11] relies on the results from previous works [7], [12], according to which it is more effective to use the same IC control in sets of contiguous eNBs (*synchronized muting*), instead of using different controls for each eNB.

However, the learning algorithm proposed in this paper is substantially different from ClassMAB and follows a novel strategy consisting of associating each control action with two concurrent learning processes. The first one, referred to as *Threshold Search* (TS) is a new mechanism based on the premise that, for each control action, the QoS objective is fulfilled only when the UE traffic intensity is below some threshold. The traffic threshold for each control is initially unknown, and the efficient discovery of all the thresholds is the objective of TS. The second learning process, *Response Estimation*, aims at estimating the function that maps traffic intensity to network energy consumption (the network *response*) for each control. Response Estimation uses a Bayesian approach, in particular a Gaussian Process, which generates response functions estimations with relatively few samples and provides the uncertainty of the estimation at each traffic intensity. This allows us to apply the principle of *optimism in the face of uncertainty* used by classic bandit algorithms [13], [14] in a novel way: each control action is associated to a function instead of a scalar. This concurrent learning strategy allows our algorithm to obtain information about multiple context values at each decision stage, increasing its sampling efficiency and therefore its learning rate with respect to other alternatives such as ClassMAB.

In summary, the main contributions of this work are:

- An online learning framework for the joint control of energy saving and interference coordination mechanisms in HetNets based on a new variant of contextual bandit problems which comprises a constraint.
- A novel approach to the above problem based on associating two learning processes to each control, one per performance metric. Each process aims at learning a *function* mapping the context variable (network traffic intensity) to one performance metric.
- A new algorithm (Threshold Search) implementing the learning process associated to the QoS response of the system. This algorithm is characterized in terms of convergence time.

The remainder of the paper is organized as follows. In Section II the related work and contribution summary are given. In Section III we describe the interference management mechanism and the system model allowing us to formulate the contextual bandit problem. In Section IV we describe our joint coordination framework and our proposed exploration algorithm, which is analyzed in Section V. Finally, the numerical results are given in Section VI and the conclusions are provided in Section VII.

II. Related Work

Both ES and IC in HetNets have been widely investigated during the last years, generally as separate problems. One of the most usual approaches in ES is to formulate the

TABLE I
COMPARISON OF ENERGY SAVING RELATED WORKS

	[15], [16]	[17]	[18]–[23]	[24]	[25]	[26]	[27]	[28]	[29]	[30], [31]	[9]	[7], [8]	[4]	**Ours**
On-Off switch.	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	No	Yes	No	Yes	**Yes**
User Associat.	No	Yes	Yes	No	No	Yes	Yes	Yes	No	Yes	No	Yes	Yes	**Yes**
Power Control	No	Yes	No	No	Yes	No	Yes	No	No	Yes	No	Yes	Yes	**Yes**
Interf. Manag.	No	No	No	No	Yes	No	No	No	No	Yes	No	Yes	Yes	**Yes**
Approach	O	O	O	SG	O	SG	SG	RL	MDP	O	OL	OL	DP	**OL**

problem as a Markov Decision Process (MDP) [28], [29]. The inherent computational complexity of the MDPs implies the use of approximate dynamic programming approaches such as Reinforcement Learning (RL) [28]. The complexity and scalability of a learning approach is directly related to the dimensionality of the state and control spaces (curse of dimensionality). Other works make a compact representation of the state space using a function approximation [28], [32], [33]. Specifically, Comsa *et al.* [32] and Comşa [33] propose a RL approach to determine the scheduling rules under QoS constraints. But even with this strategy, RL algorithms rely on offline learning before being implemented in a real network. This is because of the slow convergence properties of RL. The main drawback of this strategy is that it requires a very accurate simulation model for each specific network deployment. Our previous work [4] addressed the IC-ES control also from an offline learning perspective, using dynamic programming and certainty equivalence control. In contrast, our current proposal follows an online learning approach and is designed specifically to operate on the real network without previous training.

Other works have addressed IC in HetNets using learning algorithms [6]–[8], [34]. Simsek *et al.* [34] propose Q-learning algorithms for learning ABS ratio and CRE bias. Our previous works [7] and [8] propose online learning algorithms for IC configuration control. While [8] applies a multi-armed bandit strategy, [7] is based on Response Surface Methodology. However, these proposals are not applicable to the IC-ES problem considered in this paper, because of the diverse new challenges: the higher dimension of the problem, the absence of the main property exploited by [7] and [8] (unimodality of the system response), the presence of a second performance metric in the form of a constraint, and the inclusion of the network traffic intensity (context) in the decision making process.

Virdis *et al.* [25] and Zheng *et al.* [31] address the problem of energy saving in HetNets exploiting the ABS configuration and show that the ABS configuration has a significant impact on the power consumption. Nevertheless, eNB on-off switching is not considered.

The problem of eNB on-off switching has been also addressed as an optimization problem [15], [17]–[23], [25]. These problems are usually addressed using iterative algorithms aimed at finding suboptimal solutions since their computational complexity is NP-Hard in most cases. Moreover, the solutions of these problems have to be recomputed whenever the network state (e.g., traffic intensity) changes. In contrast, our proposal learns efficient configurations for any network state. Some of these works [17]–[23] also take into account the user association problem, but none of them considers interference management which, as our work shows, has a notable impact on the performance of on-off switching algorithms in HetNets.

Some works address the eNB switching problem using Stochastic Geometry [24], [26], [27], [35], [36]. Nevertheless, this approach is based on a network model with some simplifications (e.g., eNBs deployed following a Poisson Point Process, path loss as the channel model). In contrast, our proposal is able to learn using real data from the network and does not require any simplification nor assumption in the network model. Table I summarizes the main aspects of the previous works most related to ours.

Other works have used contextual bandit algorithms in cellular networking problems [9], [37], [38]. In [37] an algorithm for content caching based on contextual bandits is proposed. This algorithm learns the context-dependent popularity profiles in order to update the cache content efficiently. In [38] a contextual bandit algorithm addresses the beam alignment problem in millimeter wave systems. Maghsudi and Hossain [9] propose a multi-armed bandit framework for energy-efficient small cell activation but, in contrast to our work, they do not take into account the state of the network (traffic intensity), and do not consider the influence of the interference coordination mechanisms on the global network performance. To the best of our knowledge, our work is the first to apply a contextual bandit formulation for IC-ES control in HetNets.

III. SYSTEM DESCRIPTION AND PROBLEM FORMULATION

In this section, we first detail the IC mechanism and the consumption model associated with this technology. Then, we present the system model and formulate the problem addressed.

A. Interference Management in LTE-A: eICIC

The 3GPP Release 10 specifies eICIC [3] as the interference coordination mechanism for LTE-A. To minimize inter-cell interference, eICIC schedules the radio resources for pico and macro eNBs in different time periods (subframes). It comprises two main features: *Cell Range Expansion (CRE)* and *Almost Blank Subframe (ABS)*.

The CRE increases the pico eNBs footprint by adding a bias to their Received Signal Reference Power (RSRP). It is intended to balance the offloading (from macro to pico eNBs) in the network. To select an eNB to associate with, the UE adds the CRE bias to the pico RSRP but not to the macro RSRP, and then selects the eNB with maximum (corrected) RSRP. Thus, the higher the CRE bias, the larger the footprint of the

pico eNBs. However, the UEs located in the extended region (CRE UEs) will generally have a poor channel quality due to the high interference received from the macro eNB. Note that a CRE UE receives a stronger signal from the macro eNB than from the pico eNB to which it is currently associated. ABS is motivated by the need to improve the performance of CRE UEs and consists of reserving certain subframes for pico cell traffic only, muting data transmission from the macro eNB on some radio subframes (Almost Blank Subframes). The ABS ratio defines the portion of muted subframes over the total number of subframes (muted and not). We consider synchronized muting, as recommended by the 3GPP [12]. That is, the eICIC controls are applied globally to a cluster of macro eNBs with homogeneous traffic profile. This implies that ABS subframes are free from interference of nearby macro eNBs since these eNBs are muting their ABS subframes simultaneously. The SINR at UE receiver i served by eNB j is given by:

$$\text{SINR}_i = \frac{P_j^{\text{tx}} \cdot g_{i,j}}{\sum_{m\in\mathcal{M}_i} P_m^{\text{tx}} \cdot g_{i,m} + \sum_{p\in\mathcal{P}_i} P_p^{\text{tx}} \cdot g_{i,p}} \tag{1}$$

where:

- P_j^{tx} is the transmission power of eNB j;
- $g_{i,j}$ is the channel gain between eNB j and UE i;
- $\mathcal{M}_i$ is the set of interfering macro eNBs of UE i;
- $\mathcal{P}_i$ is the set of interfering pico eNBs of UE i.

Note that in ABS subframes $\mathcal{M}_i = \emptyset$, because of synchronized muting. The CRE bias also affects the SINR since it determines UE association decisions.

B. eNB Power Consumption Model

The eNB consumption model used in this work is based on 3GPP guidelines [39]. The power consumption of some of the components of an eNB depends on its load. Thus, it is common to assume a linear relationship between RF output power and power consumption of eNB transceivers (TRXs) [39]. The power consumption model of a pico eNB j is given by:

$$C_{\text{p}}^j = e^j \cdot N_{\text{TRX}} \cdot (P_0 + R^j \cdot P_{\text{max}}) + (1 - e^j) \cdot N_{\text{TRX}} \cdot P_{\text{sleep}} \tag{2}$$

where:

- $e^j = 1$ when the pico eNB j is active and $e^j = 0$ otherwise;
- N_{TRX} is the number of TRXs;
- P_{max} is the TRX power consumption when the eNB transmits at maximum RF output;
- P_0 represents the TRX power consumption when the eNB is active but not transmitting;
- $R^j \in [0, 1]$ is the load factor of the pico eNB j and depends on the ABS ratio, the CRE bias, the traffic intensity and the location of UEs;
- P_{sleep} is the power consumption of TRX components in sleep mode.

The power consumption of the macro eNB i is given by

$$C_{\text{m}}^i = N_{\text{TRX}} \cdot (P_0^{\text{m}} + R^i \cdot P_{\text{max}}^{\text{m}}) \cdot (1 - \gamma) + N_{\text{TRX}} \cdot P_0^{\text{m}} \cdot \gamma \tag{3}$$

where γ denotes the ABS ratio, $P_{\text{max}}^{\text{m}}$ is the maximum power output of the macro eNB and P_0^{m} is the power consumption at zero RF output power of the macro eNB. Given the influence of the ABS ratio and the CRE bias on C_{p}^j and C_{m}^i, our proposal includes these parameters in the control of the energy consumption.

C. System Model

We consider a set of M macro eNB sectors denoted by $\mathcal{M}$. Let $\mathcal{P}^m$ be the set of pico eNBs overlapping the macro sector m. We denote the ABS ratio and the CRE bias by $\gamma \in \Gamma$ and $\phi \in \Phi$, respectively, where Γ and Φ are the finite sets comprising all available configurations for these parameters. Time is divided into stages denoted by $k \in \{0, 1, \ldots\}$.

1) States: Let e_k^j be the state of the pico eNB $j \in \mathcal{P}^m$ at stage k, where $e_k^j = 1$ when j is switched on and $e_k^j = 0$ otherwise. Let $p_k^m = (e_k^1, \ldots, e_k^{|\mathcal{P}^m|})$ be the vector indicating the on/off state of all the pico eNBs in $\mathcal{P}^m$. The joint state of all picos in the network at stage k is represented as $p_k = (p_k^1, \ldots, p_k^M)$, and the set of all possible values of p_k is denoted by $\mathcal{E}$. Let $\lambda_k \in \Lambda$ be the aggregate traffic load in the network at stage k, where Λ is a set containing all possible values of traffic intensity. We define the network state at stage k as $s_k = (\lambda_k, p_{k-1}) \in \mathcal{S}$, where $\mathcal{S} = \Lambda \times \mathcal{E}$ is the state space.

2) Controls: The network control is given by $a_k = (p_k, \gamma_k, \phi_k) \in \mathcal{A}$, where $\mathcal{A} = \mathcal{E} \times \Gamma \times \Phi$ is the control space. Let $a_k^m = (p_k^m, \gamma_k, \phi_k)$ be the local control for the sector m. Given s_k, the decision maker selects a control a_k based on its previous knowledge. The upcoming network state, $s_{k+1} = (\lambda_{k+1}, p_k)$, depends on the current control and on the traffic at the next stage, which is unknown in advance.

3) Feedback functions: We define two feedback functions: $C : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ providing the aggregated power consumption of macro and pico eNBs in the network and $Q : \mathcal{S} \times \mathcal{A} \to [0, 1]$ which gives the ratio of UEs in the network fulfilling a minimum value of a performance metric selected by the operator. Let Q_{min} be the minimum value of Q allowed in the network, i.e., the minimum ratio of UEs in the network meeting the performance metric value selected by the operator. Note that the values obtained from C and Q are random variables due to the randomness of UE locations and traffic demands.

D. Constrained Contextual Bandit Formulation

We define a policy as a function $\pi : \mathcal{S} \to \mathcal{A}$ which maps network states into controls. A learning agent following a policy π operates as follows: (i) the learning agent obtains the network state s_k at stage k and selects the control a_k that policy π prescribes for s_k; (ii) the network operates according to the control a_k during stage k, gathering performance measures from each eNB in order to obtain the feedback values (C_k, Q_k) that are sent back to the learning agent at the end of stage k; (iii) the learning agent receives the feedback and updates the policy π accordingly.

Our goal is to learn, stage by stage, a policy π minimizing the power consumption, while attaining a minimum desired QoS threshold Q_{min}. We define the set of policies satisfying this QoS requirement as follows

$$\Pi^{\text{QoS}} = \{\pi \in \Pi : Q(s, \pi(s)) \geq Q_{\text{min}} \ \ \forall s \in S\} \tag{4}$$

where Π is the set of all possible policies. The optimal policy π^* is the one that minimizes the average consumption per stage in the long term, i.e.,

$$\pi^* = \arg\min_{\pi \in \Pi^{\text{QoS}}} \lim_{N\to\infty} \frac{1}{N} E\left[\sum_{k=1}^{N} C(s_k, \pi(s_k))\right] \tag{5}$$

where the expectation is taken with respect to the traffic intensity (λ_k in s_k).

Note that Π^{QoS} is initially unknown and must also be discovered by the learning algorithm. This implies that the algorithm needs to select, during the learning process, controls a_k that violate the QoS constraint. Therefore, in order to evaluate the efficiency of the learning process, we need to define a per-stage cost function including a penalty term for violations of the QoS constraint. For this purpose we define

$$\rho(s, a) = C(s, a) + C^{\text{QoS}}(s, a). \tag{6}$$

where the penalty function C^{QoS} satisfies the following conditions: if $Q(s, a) < Q_{\min}$, then $C(s, a) + C^{\text{QoS}}(s, a) > C(s, a')$, where a' is any control satisfying the QoS requirement; and if $Q(s, a) \geq Q_{\min}$, then $C^{\text{QoS}}(s, a) = 0$. Given this definition of $\rho(s, a)$, the optimal policy π^* can be alternatively defined as follows

$$\pi^* = \arg\min_{\pi \in \Pi} \lim_{N\to\infty} \frac{1}{N} E\left[\sum_{k=1}^{N} \rho(s_k, \pi(s_k))\right]. \tag{7}$$

The pseudo-regret (referred to as regret henceforth) of a policy π over N stages is given by

$$R_\pi(N) = \sum_{k=0}^{N} \Big(E\left[\rho(s_k, \pi(s_k))\right] - E\left[\rho(s_k, \pi^*(s_k))\right] \Big). \tag{8}$$

This metric accumulates the loss incurred when selecting a suboptimal control at each stage and can then be used to assess the performance of a policy. That is, the lower the regret of a policy, the closer the policy to the optimal one. In addition, it also characterizes the convergence rate towards the optimal policy. A sub-linear regret implies that the performance of the algorithm converges towards the optimum.

In order to find policies minimizing (8), it is necessary to deal with the curse of dimensionality since the dimensions of the state and control spaces ($\mathcal{S}$ and $\mathcal{A}$) make the problem intractable as the network size grows. In particular, the size of the set $\mathcal{E}$ grows exponentially with the number of pico eNBs per sector and with the number of sectors ($|\mathcal{E}| = 2^{\sum_{m=1}^{M} |\mathcal{P}^m|}$). Therefore, a solution algorithm should incorporate a dimension reduction strategy, as the one explained in next section.

IV. Proposed Scheme

In this section we first describe the general framework in terms of black box functionality and data flow. Then, we detail the proposed algorithm for finding an efficient control policy. Table II summarizes the most relevant parameters of the proposal.

TABLE II
NOTATION TABLE

Notation	Definition
M, $\mathcal{M}$	Number of macro eNBs, set of macro eNBs
$\mathcal{P}^m$	Set of pico eNBs in sector m
$\gamma \in \Gamma, \phi \in \Phi$	ABS ratio, CRE bias
p^m	Activation state of pico eNBs in sector m
p	Activation state of pico eNBs in the network
$\lambda \in \Lambda$	Traffic intensity
$s = (\lambda, p) \in \mathcal{S}$	Network state, network state space
$a = (p, \gamma, \phi) \in \mathcal{A}$	Control, control space
$a^m = (p^m, \gamma, \phi)$	Local control for sector m
C	Consumption feedback function
Q	QoS feedback function
$r \in \mathcal{R}$	Ratio of active pico eNBs in the sector
$\tilde{s} = (\lambda, r) \in \tilde{\mathcal{S}}$	Global state, global state space
$u = (r, \lambda, \phi) \in \mathcal{U}$	Global control, global control space

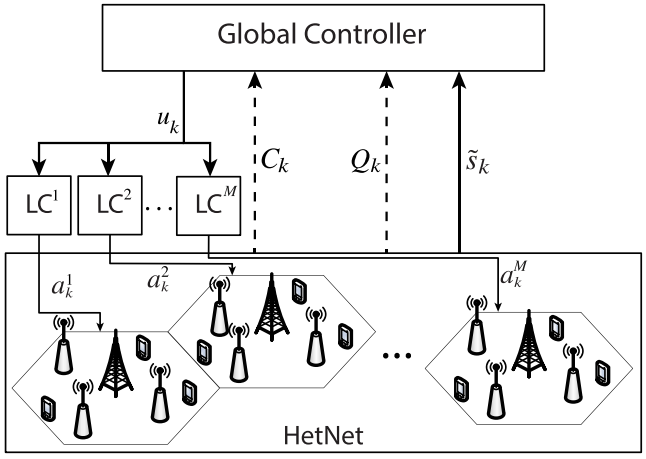

Fig. 2. ES and IC framework comprising a HetNet with M sectors, the global controller that learns efficient global controls from the network state and performance metrics, and M local controllers (LC^m) which compute local controls for each sector.

A. Two-Level Framework

Previous works [7], [12] have discussed the performance benefits of applying the same eICIC parameter configuration in contiguous sectors (synchronized ABS). Following this approach, we consider that the access network is divided into groups of contiguous macro cells, referred to as clusters, with similar traffic conditions, such that all the sectors of the cluster share the same eICIC configuration values, γ_k and ϕ_k.

The functional elements of the proposed scheme are depicted in Fig. 2 and consist of the controlled cluster comprising M sectors, a central entity (global controller), and M local controllers (LCs), one per sector. This scheme decomposes the decision problem into two levels. At the higher level, the global controller operates with state and action spaces with reduced dimensions, while at the lower level, the local controllers translate the *global controls* sent by the global controller into M *local controls* a_k^m, with full dimension, that together constitute the complete network control a_k.

This dimension reduction is done by replacing the activation state vector p_k by the ratio of active pico eNBs per sector in the HetNet, $r_k \in \mathcal{R}$, where $\mathcal{R}$ is a finite set containing the possible values of r_k. The *global state* is then defined as $\tilde{s}_k = (\lambda_k, r_k)$. Therefore, the *global control* is defined as

the triplet $u_k = (r_k, \gamma_k, \phi_k)$. Note that the dimension of the state and control spaces, denoted by $\tilde{\mathcal{S}}$ and $\mathcal{U}$, respectively, are linear with $|\mathcal{R}|$ and independent of M. Therefore, the idea is that similar traffic conditions allowing the M sectors to benefit from using the same eICIC configuration γ_k and ϕ_k, also allow them to share the same ratio r_k, provided that each local controller LC^m applies r_k efficiently to generate its local activation vector p_k^m.

The parameters determined by the general controller are passed to the different LCs, which in turn must determine the actual eNB activation patterns p_k^m. To this end, the LCs consider the *cell adjacency* parameter, which measures how much a pico eNB is isolated, i.e., far from other base stations. The idea is that isolated pico eNBs should not be switched off, to avoid that all their UEs get connected to the macro eNB, increasing the energy consumption. Formally, the cell adjacency of the pico eNB j is defined as

$$d^j = w \cdot d_m^j + (1 - w) \cdot d_p^j \tag{9}$$

where d_{m}^j is the distance to macro eNB, d_{p}^j is the average distance to the rest of pico eNBs in the sector and $w \in [0, 1]$ is a weighting factor. Then, the $\lceil r_k \cdot |\mathcal{P}^m| \rceil$ pico eNBs with the largest cell adjacency will be activated, while the others will be switched off. Formally, the elements in $p_k^m = (e_k^1, \ldots, e_k^{|\mathcal{P}^m|})$ are obtained as follows

$$e_k^j = \begin{cases} 1 & \text{if } D^m(j) \leq \lceil r_k \cdot |\mathcal{P}^m| \rceil \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

where $D^m(j)$ is the ranking of pico eNB j when the pico eNBs are listed in decreasing order of cell adjacency.

To summarize, the proposed framework operates according to the following data flow:

- At each stage k, all macro eNBs receive from their corresponding pico eNBs the information about traffic intensity in their cells. The macro eNB aggregates this information and sends it to the global controller. The global controller computes the overall traffic $\tilde{s}_k$ from the information received from the macro eNBs and determines the general state (λ_k, r_k).
- The global controller selects a global control u_k and broadcasts it to the *LCs*.
- Each local controller (LC^m for $m \in \mathcal{M}$) receives $u_k = (r_k, \gamma_k, \phi_k)$ and transforms r_k into a local vector p_k^m in order to obtain a local control $a_k^m = (p_k^m, \gamma_k, \phi_k)$.
- Each sector $m \in \mathcal{M}$ operates with its corresponding local control a_k^m during stage k.
- At the end of the stage, each macro eNB receives data about the power consumption and the QoS from its corresponding pico eNBs. The macro eNBs aggregates this information to be sent to the global controller. The global controller computes the feedback Q_k and C_k from the data received from all the macro eNBs.
- Using this feedback, the global controller updates its knowledge according to the learning process described in the following subsection.

Note that the duration of a stage is determined by the time between consecutive performance observations. Therefore, increasing the signaling frequency implies reducing the duration of a stage, resulting in a tradeoff between signaling overhead and convergence time.

B. Global Controller Algorithms

In this section we present the *Bayesian Response Estimation and Threshold Search* (BRETS) algorithm that operates in the global controller selecting the global control u_k at each stage according to past observations. For each control $u \in \mathcal{U}$, BRETS handles two learning processes in parallel, *Threshold Search* and *Response Estimation*, associated to the energy consumption and the QoS fulfillment, respectively. We describe first the principles and strategies used by these processes, and then we will present the complete BRETS algorithm in which both schemes are coordinated.

1) Threshold Search (TS): TS is based on the following premise: each control u determines the number of pico eNBs in active mode, as well as the interference level and available frame resources for pico and macro UEs (eICIC parameters). Thus, u is indeed determining the capacity of the controlled network. This implies that, for each u, there exists a user traffic intensity λ_{th}^u above which the QoS requirement cannot be met. We refer to λ_{th}^u as the traffic threshold for control u. Only if the traffic intensity is below λ_{th}^u, is the network capacity associated to u sufficient to attain the required QoS objective.

The value λ_{th}^u for each control $u \in \mathcal{U}$ is unknown a priori. The objective of TS is to progressively discover the traffic threshold λ_{th}^u for each control $u \in \mathcal{U}$. Let l_k^u denote the highest λ_k under which control u has been used so far, such that the QoS requirement has been satisfied ($Q_k \geq Q_{\mathrm{min}}$), and let h_k^u denote the lowest λ_k for which u is known not to satisfy the QoS requirement ($Q_k < Q_{\mathrm{min}}$). We define the *uncertainty region* of a control u at stage k as the values of λ between the bounds l_k^u and h_k^u. The threshold λ_{th}^u is thus contained in this region, i.e., $l_k^u < \lambda_{\mathrm{th}}^u < h_k^u$. The narrower the uncertainty region, the more accurate the knowledge about λ_{th}^u.

According to the system data flow (Fig. 2), at the end of each stage k the algorithm receives the feedback measurements C_k and Q_k associated with the selected control u_k and the global state $\tilde{s}_k$. The QoS feedback is used to gradually narrow the *uncertainty region* of the selected control. Thus, for a selected control u, when $Q_k \geq Q_{\mathrm{min}}$ (and thus $\lambda_k \leq \lambda_{\mathrm{th}}^u$), its lower bound is updated as follows $l_k^u = \max\{\lambda_k, l_{k-1}^u\}$. Otherwise ($\lambda_k > \lambda_{\mathrm{th}}^u$), the upper bound is updated according to $h_k^u = \min\{\lambda_k, h_{k-1}^u\}$.

The QoS fulfilling function Q, could comprise multiple QoS constraints, according to the preferences and objectives of the operator. For example, the operator could be interested in guaranteeing that the distribution of the user throughput satisfies a minimum required profile by setting a throughput objective to be attained by at least 5% of its users and another throughput objective for 50% of its users. The traffic threshold for each control will be defined as the maximum network traffic beyond which these two objectives cannot be met using this control. As a consequence, if for a given traffic λ_k, a control u does not satisfy both QoS constraints, then TS updates the upper bound h_k^u to λ_k.

2) Response Estimation: For each control u, the function that maps the traffic intensity λ to the network energy consumption is referred to as the *network response* for control u. These functions are initially unknown, and must be learned as well, which in our proposal is done by associating each control u to a Gaussian Process (GP) [40].

Let $\mathcal{L}^u$ be a finite set in which we store tuples (λ_k, C_k) associated to the performance of the control u. The maximum length of $\mathcal{L}^u$ is set to N. In Bayesian estimation a posterior distribution is computed from a prior distribution and a set of observations. In this work, the GP obtains the posterior distribution of the power consumption (the mean $\mu^{u,\lambda}$ and the standard deviation $\sigma^{u,\lambda}$) for a control u and a traffic λ from a prior distribution and the set of observations $\mathcal{L}^u$.

Let $C_k^{u,\lambda} = \mu^{u,\lambda} - 2\sigma^{u,\lambda}$ denote the lower bound of the expected consumption for the control u under the traffic λ at stage k. The estimation of $C_k^{u,\lambda}$ will allow BRETS to find low power consumption controls.

The reasons for using GPs are basically two. First, they require relatively few samples to approximate the functions, because they assume smoothness in the response. In other words, the use of GPs implicitly assumes that a given control obtains similar energy savings for similar traffic intensities, which is reasonable in our setting. Second, GPs provide an estimation of the uncertainty of the estimated response for any traffic intensity. This is especially useful in an online learning framework since the algorithm can exploit the principle of *optimism in the face of uncertainty*. In our setting, using this principle implies selecting the controls whose estimated energy cost is lowest, considering the lower bound on this estimation $C_k^{u,\lambda}$. By doing this, the algorithm tends to explore the areas of the functions with higher uncertainty $\sigma^{u,\lambda}$, but also to exploit the areas with lower empirical average $\mu^{u,\lambda}$. This extends the upper confidence bound (UCB) strategy to the case of learning a *function* per control action, instead of a scalar.

3) Algorithm Operation: At each stage, the algorithm can be in three different modes of operation: initialization, exploration, and exploitation. The algorithm starts in the initialization mode, which is visited only once. Then, at each stage, the algorithm can enter either the exploration or the exploitation mode. Let us detail each operation mode of the algorithm.

1) **Initialization.** The algorithm initializes the values of the bounds $l_k^u = 0$ and $h_k^u = \lambda^{\max}$ for all $u \in \mathcal{U}$. Then, each $u \in \mathcal{U}$ is selected once. That is, during the first $|\mathcal{U}|$ stages the algorithm explores all controls. After this initial exploration, each control $u \in \mathcal{U}$ has an initial estimation of its average consumption and an initialized lower or upper bound.
2) **Exploration.** In this operation mode the algorithm aims at reducing the uncertainty regions. To this end, the controller select actions for which the sustainability of the current traffic load λ_k is unknown, i.e., actions belonging to the set

$$\mathcal{H}_k = \{u \in \mathcal{U} : l_k^u < \lambda_k < h_k^u\}. \tag{11}$$

An example is sketched in Fig. 3, where we can see that the first control satisfies the condition of being in $\mathcal{H}_k$.

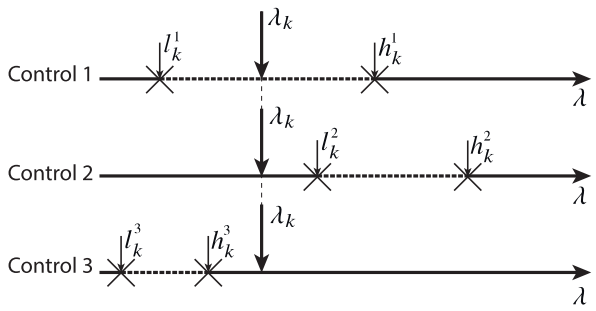

Fig. 3. Example of the values of the bounds l_k^u and h_k^u for three controls: $\mathcal{U} = \{1, 2, 3\}$. The *uncertainty region* of each control is marked with a dotted line. Given the traffic intensity λ_k shown in the example, control 1 can be selected in the exploration mode; control 2 can be selected in the exploitation mode; control 3 will not be selected since it is expected to fail the QoS requirement ($\lambda_k > h_k^3$).

Let $\mathcal{H}'_k$ denote the set of controls $u \in \mathcal{H}_k$ for which either $l_k^u = 0$ or $h_k^u = \lambda^{\max}$. If $\mathcal{H}'_k \neq \emptyset$ then $\mathcal{H}_k$ is replaced by $\mathcal{H}'_k$ ($\mathcal{H}_k \leftarrow \mathcal{H}'_k$). This allows the algorithm to continue the initialization of the uncertainty regions in the exploration mode.

To balance exploration and exploitation, we define $B \in \mathbb{N}$ as the exploration budget of the algorithm, which decreases by one unit every time the algorithm enters the exploration mode. Let $\varepsilon_0 \in \mathbb{R}^+$ be the exploration constant. A control from $\mathcal{H}_k$ is selected if one of the following events takes place: (i) $B > 0$ (ii) $\chi < \varepsilon_0/k$, where $\chi \in [0, 1]$ is a uniformly distributed random number. Then, the algorithm picks a control from $\mathcal{H}_k$ for which the potential reduction in its uncertainty is the largest:

$$u_k = \underset{u \in \mathcal{H}_k}{\operatorname{argmax}} \left[\min\left[(\lambda_k - l_k^u), (h_k^u - \lambda_k)\right]\right]. \tag{12}$$

If no exploration control is selected (i.e., $\mathcal{H}_k = \emptyset$ or $B = 0$ and $\chi > \varepsilon_0/k$), the algorithm switches to the exploitation mode.

3) **Exploitation.** Let $\mathcal{T}_k$ be the set containing all exploitation controls (those that are known to satisfy the QoS constraints):

$$\mathcal{T}_k = \{u \in \mathcal{U} : \lambda_k \leq l_k^u\} \tag{13}$$

For example, the second control in Fig. 3 belongs to $\mathcal{T}_k$ and can potentially be selected in the exploitation phase. More specifically, the algorithm selects a control from the set $\mathcal{T}_k$ such that

$$u_k = \underset{u \in \mathcal{T}_k}{\operatorname{argmin}} C_k^{u,\lambda_k}. \tag{14}$$

If $\mathcal{T}_k = \emptyset$, the control closest to satisfying the QoS is selected, that is,

$$u_k = \underset{u \in \mathcal{U}}{\operatorname{argmin}} \left[\lambda_k - l_k^u\right]. \tag{15}$$

The operation of BRETS is summarized in Algorithm 1.

V. Algorithm Analysis

This section analyzes the exploration strategy (12) of the BRETS algorithm. First we provide a bound on the expected

Algorithm 1 BRETS Algorithm

Input parameters: ε_0, B
Initialization: $l_k^u = 0$, $h_k^u = \lambda^{\max}$
Pick every control $u \in \mathcal{U}$ once
for each stage k **do**
  $\mathcal{H}_k = \{u \in \mathcal{U} : l_k^u < \lambda_k < h_k^u\}$
  $\mathcal{H}'_k = \{u \in \mathcal{H}_k : l_k^u = 0 \text{ or } h_k^u = \lambda^{\max}\}$
  if $\mathcal{H}'_k \neq \emptyset$ **then**
    $\mathcal{H}_k \leftarrow \mathcal{H}'_k$
  end if
  if $\mathcal{H}_k \neq \emptyset$ **and** $(B > 0$ **or** $\chi < \varepsilon_0/k$ **or** $\mathcal{H}'_k \neq \emptyset)$ **then**
    $B = B - 1$
    $u_k = \underset{u \in \mathcal{H}_k}{\text{argmax}} \left[\min\left[(\lambda_k - l_k^u), (h_k^u - \lambda_k)\right]\right]$
  else
    $\mathcal{T}_k = \{u \in \mathcal{U} : \lambda_k < l_k^u\}$
    if $\mathcal{T}_k \neq \emptyset$ **then**
      $u_k = \underset{u \in \mathcal{T}_k}{\text{argmin}}\, C_k^{u,\lambda_k}$
    else
      $u_k = \underset{u \in \mathcal{U}}{\text{argmin}} \left[\lambda_k - l_k^u\right]$
    end if
  end if
  Send u_k to *LCs*
  Receive the feedback C_k and Q_k from the network
  Update $\mathcal{L}^{u_k}$ with C_k and Q_k
  Update the GP associated with u_k using $\mathcal{L}^{u_k}$
  if $Q_k > Q_{\min}$ **then**
    $l_k^{u_k} = \max\left\{\lambda_k, l_{k-1}^{u_k}\right\}$
  else
    $h_k^{u_k} = \min\left\{\lambda_k, h_{k-1}^{u_k}\right\}$
  end if
  $n^{u_k} = n^{u_k} + 1$
end for

convergence time of the exploration process, and then we formulate this process as a stochastic shortest path (SSP) problem. This allows us to show how our BRETS algorithm exploits the structure of the optimal cost-to-go function.

A. Bound on the Expected Convergence Time

For the analysis, we consider that, at each stage k, the traffic intensity λ_k takes a random value from the finite set $\Lambda = \{0, \delta, 2\delta, \ldots, \lambda^{\max}\}$, where δ denotes the granularity of the traffic measurement. We further assume that the random variables λ_k, for $k = 0, 1, \ldots$ are i.i.d. with a probability distribution such that $P(\lambda_k = \lambda) > 0$ for all $\lambda \in \Lambda$. The values of λ_{th}^u for each $u \in \mathcal{U}$ are also randomly distributed over Λ, and are initially unknown. Let $x_k^u = \{\lambda \in \Lambda : l_k^u < \lambda < h_k^u\}$ denote the uncertainty region for control u at stage k, and let $x_k = (x_k^1, \ldots, x_k^{|\mathcal{U}|})$ be a vector containing all the uncertainty regions.

For discrete λ_k values, the exploration ends when $h_k^u = \lambda_{\text{th}}^u$ and $l_k^u = \lambda_{\text{th}}^u - \delta$,[1] which is equivalent to $x_k^u = \emptyset$ or $|x_k^u| = 0$, for all $u \in \mathcal{U}$. With a slight abuse of notation we define $|x_k| = (|x_k^1|, \ldots, |x_k^{|\mathcal{U}|}|)$, so that the exploration ending condition can be expressed as $|x_k| = (0, \ldots, 0) = \mathbf{0}$. Given a sequence $x_0, x_1, \ldots$ the convergence time is defined as $T = \min\{k : |x_k| = \mathbf{0}\}$.

Definition 1: Let us define as an *appropriate* exploration strategy any strategy selecting $u \in \mathcal{H}_k$ whenever $\mathcal{H}_k \neq \emptyset$ (as our BRETS strategy does).

The following result provides an upper bound on the expected convergence time of any appropriate exploration strategy.

Lemma 1: Consider that, at every k, λ_k takes a random value from a finite set Λ, with a stationary distribution such that $P(\lambda_k = \lambda) > 0$ for all $\lambda \in \Lambda$. Then, the expected convergence time $E[T]$ for any appropriate exploration strategy is bounded as follows

$$E[T] \leq \frac{3}{2} \frac{|\mathcal{U}|}{\min_{\lambda \in \Lambda} P(\lambda_k = \lambda)} \tag{16}$$

Proof: See appendix A.

B. Stochastic Shortest Path Model

The exploration process can be modeled as an SSP over an infinite time horizon [41], comprising the following elements:

1) The state of the process at stage k is given by the vector of uncertainty regions x_k and the traffic intensity λ_k.
2) At each k, the exploration strategy μ (*policy* in the SSP terminology) determines which $u \in \mathcal{U}$ should be selected according to the observed state (x_k, λ_k). We restrict our attention to stationary deterministic policies, i.e., $u_k = \mu(x_k, \lambda_k)$ for all $k = 0, 1, \ldots$
3) When a control $u \in \mathcal{U}$ is selected, $x_{k+1}^{u'} = x_k^{u'}$ for all $u' \neq u$, and x_{k+1}^u can take two values: $\bar{x}_{k+1}^u$, and $\underline{x}_{k+1}^u$. The first one corresponds to the update $l_{k+1}^u = l_k^u$, $h_{k+1}^u = \min[\lambda_k, h_k^u]$ which occurs with probability $P(\lambda_k \geq \lambda_{\text{th}}^u)$; the second one $\underline{x}_{k+1}^u$ corresponds to the update $l_{k+1}^u = \max[\lambda_k, l_k^u]$, $h_{k+1}^u = h_k^u$, with probability $P(\lambda_k < \lambda_{\text{th}}^u)$.
4) The transition probabilities between consecutive states are determined by the policy μ, and by the probability distributions of λ_k and λ_{th}^u.
5) The termination state corresponds to $|x_k| = \mathbf{0}$.
6) The per-stage cost is 1 when $|x_k| \neq \mathbf{0}$ and 0 otherwise.

With the above elements we can define the following cost-to-go function for policy μ at a given state (x_k, λ_k), as follows

$$J_\mu(x_k, \lambda_k) = \lim_{T \to \infty} E\left[\sum_{t=k}^{T} \mathbb{I}_{\{|x_t| \neq \mathbf{0}\}} \middle| x_k, \lambda_k\right], \tag{17}$$

where $\mathbb{I}$ is the indicator function, and the conditional expectation is obtained with respect to the transition probabilities induced by μ and the distributions of λ_k and λ_{th}^u. The objective is to find, for an initial state (x_0, λ_0), the optimal policy $\mu^* = \arg\min_\mu J_\mu(x_0, \lambda_0)$. At every state (x_k, λ_k), the optimal cost-to-go function J^* satisfies Bellman's equation

$$J^*(x_k, \lambda_k) = 1 + \min_{u \in \mathcal{U}} E\left[J^*(x_{k+1}, \lambda) \middle| x_k, \lambda_k, u_k = u\right]. \tag{18}$$

[1] This ending condition implicitly assumes that $\lambda_{\text{th}}^u > 0$, which is equivalent to assuming that every u fulfills the QoS requirement in the absence of user traffic. Otherwise this u should not even be considered for exploration.

To simplify the notation, we consider only two controls $\mathcal{U} = \{1, 2\}$, although the following discussion can be generalized to any finite set of controls. For $x_k = (x_k^1, x_k^2)$ we can use the notation $J^*(x_k^1, x_k^2, \lambda_k)$ or $J^*(x_k, \lambda_k)$ conveniently. Let us develop (18) for $\mathcal{U} = \{1, 2\}$:

$$J^*(x_k^1, x_k^2, \lambda_k) = 1 + \min \left[E\left[J^*(x_{k+1}^1, x_k^2, \lambda) \middle| x_k, \lambda_k \right], E\left[J^*(x_k^1, x_{k+1}^2, \lambda) \middle| x_k, \lambda_k \right] \right] \quad (19)$$

where

$$\begin{aligned} &E\left[J^*(x_{k+1}^1, x_k^2, \lambda) \middle| x_k, \lambda_k \right] \\ &\quad = P(\lambda_k \geq \lambda_{\text{th}}^1) E_\lambda \left[J^*(\bar{x}_{k+1}^1, x_k^2, \lambda) \right] \\ &\qquad + P(\lambda_k < \lambda_{\text{th}}^1) E_\lambda \left[J^*(\underline{x}_{k+1}^1, x_k^2, \lambda) \right], \text{ and} \\ &E\left[J^*(x_k^1, x_{k+1}^2, \lambda) \middle| x_k, \lambda_k \right] \\ &\quad = P(\lambda_k \geq \lambda_{\text{th}}^2) E_\lambda \left[J^*(x_k^1, \bar{x}_{k+1}^2, \lambda) \right] \\ &\qquad + P(\lambda_k < \lambda_{\text{th}}^2) E_\lambda \left[J^*(x_k^1, \underline{x}_{k+1}^2, \lambda) \right] \end{aligned} \quad (20)$$

and E_λ denotes the expectation computed with respect to λ. Now we provide a definition followed by a result characterizing the structure of the optimal cost-to-go function J^*.

Definition 2: Given two uncertainty region vectors $x_k = (x_k^1, x_k^2)$ and $z_k = (z_k^1, z_k^2)$, we say that x_k *contains* z_k if $z_k^1 \subseteq x_k^1$ and $z_k^2 \subseteq x_k^2$, and we express it as $z_k \subseteq x_k$.

Lemma 2 (Monotonicity of J^*): The optimal cost-to-go function satisfies the following property:

$$J^*(x_k, \lambda_k) \geq J^*(z_k, \lambda_k), \quad \text{if } z_k \subseteq x_k \quad (21)$$

for all $\lambda_k \in \Lambda$.

Proof: See Appendix B

This monotonicity result can be used to approximately solve the SSP using an index policy, as discussed in the next subsection.

C. Index Policy

There are two main difficulties for computing an optimal policy. First, the dimensionality of the problem. Specifically, the state space comprises $\binom{|\Lambda|}{2}^{|\mathcal{U}|}$ possible states, which renders the SSP intractable for practical values of $|\Lambda|$ and $|\mathcal{U}|$ (note that $\mathcal{U}$ comprises all possible combinations of the ratio of active stations r, the ABS ratio γ, and the CRE value ϕ). Second, the distribution of λ_{th}^u for each $u \in \mathcal{U}$ is, in general, not known *a priori*, which means that the transition probabilities of the SSP cannot be computed accurately.

A feasible approach to overcome the previous limitations is to use an index policy exploiting the structure of the optimal cost-to-go function J^*. According to Lemma 2, for a given λ and u, the reduction in the cost-to-go function, $J^*(x_k, \lambda) - J^*(x_{k+1}, \lambda)$, is larger for larger reductions of the uncertainty region $|x_k^u| - |x_{k+1}^u|$. Because there are two possible values of x_{k+1}^u ($\bar{x}_{k+1}^u$ and $\underline{x}_{k+1}^u$) for each control u, one reasonable index policy is to select the u with the largest expected reduction:

$$u_k = \underset{u \in \mathcal{H}_k}{\text{argmax}} \left[P(\lambda_k \geq \lambda_{\text{th}}^u)(h_k^u - \lambda_k) + P(\lambda_k < \lambda_{\text{th}}^u)(\lambda_k - l_k^u) \right]. \quad (22)$$

Because the distribution of λ_{th}^u is in general unknown, BRETS implements a worst case approach to (22). Note that (12) is equivalent to

$$u_k = \underset{u \in \mathcal{H}_k}{\text{argmax}} \left[\min \left[|\bar{x}_{k+1}^u|, |\underline{x}_{k+1}^u| \right] \right] \quad (23)$$

and thus it is a maxmin index policy. Our numerical simulations have shown that the index policy (23) converges almost at the same rate of (22). We evaluated index policies determined by the minimum or the expected largest uncertainty region, and both of them obtained slower convergence rates than (23).

VI. Numerical Results

A. Description of the Simulation Framework

The numerical evaluations have been performed by using a custom simulation framework developed in Python, which is based on the 3GPP guidelines for the evaluation of LTE networks [42]. The numerical results shown in this section are obtained using synthetic network data generated by this simulator. The network layout comprises 5 sectorized macro eNBs (120 degrees) and several pico eNBs overlapping each macro coverage area. We simulate the central sector using the remaining ones to emulate the aggregated interference of a larger network. The wireless channel is composed of pathloss and stochastic shadow fading. The aggregated interference at each UE receiver consists of the power received from all interfering eNBs in the sector (picos and macro) plus the interference from the macro eNBs from nearby sectors as detailed in (1).

In our numerical evaluations, we define Q as the ratio of UEs whose throughput is above a minimum value denoted by $T_{\text{min}} = 100$ kbps. Each incoming UE generates one throughput measurement, which is defined according to the 3GPP guidelines [42]. The power consumption model is defined in Section III-B and the values of its parameters are shown in Table III.

The number of pico eNBs per sector is $P = 6$ and the number of ES controls is $|\mathcal{R}| = 7$. The sets of available configurations of eICIC parameters are $\Gamma = \{0, \frac{1}{8}, \frac{2}{8}, \ldots, \frac{7}{8}\}$ and $\Phi = \{0, 6, 9, 12, 18\}$. For each control u one Gaussian process (GP) is maintained by BRETS. For each GP, the maximum length of the set $\mathcal{L}^u$ is fixed to $N = 100$, the mean of the prior distribution is set to zero, and the covariance matrix is specified by the Matern kernel. The hyperparameters of the kernel are optimized using a quasi-Newton algorithm. For the sake of computational efficiency, these hyperparameters are updated (step 20 in Algorithm 1) only 10 times during the training phase of our simulations, whenever the set $\mathcal{L}^u$ has the length 1, 2, 3, 5, 8, 13, 22, 36, 60, or 100. Gaussian processes have been implemented using the Python toolbox Scikit-learn [43]. The remaining simulation parameters are shown in Table III.

B. Comparison of Local Control Strategies

In this subsection we compare our proposed local control strategy (eNBs with lower values of cell adjacency are switched off first) to other alternative strategies, including the optimal one obtained by evaluating all possible orderings.

TABLE III
SIMULATION PARAMETERS

Network layout	5 sectorized macro eNBs, 500 m ISD, P = 6 pico eNBs per sector
System bandwidth	10 MHz
LTE frame duration	Subframe 1 ms, Protected-subframe pattern 8 ms, Frame 10 ms
Transmit power	Macro eNB 46 dBm, pico eNB 30 dBm
Macro sector antenna pattern	$A_H(\phi) = -\min[12(\frac{\phi}{\phi_{3dB}})^2, A_m]$, A_m = 70 degrees A_m = 25 dB
Pico antenna pattern	Omnidirectional
Antenna gains	macro: 14 dBi; pico: 5 dBi; UE: 0 dBi
Macro to UE path loss	128.1 + 37.6 · $\log_{10}(R[\text{Km}])$ where R is the macro eNB to UE distance
Pico to UE path loss	149.7 + 36.7 · log10(R[Km]) where R is the pico eNB to UE distance
Shadow fading	Lognormal distribution with 10 dB standard deviation
Thermal noise density	-176 dBm/Hz
Scheduling algorithm	Proportional Fair (PF)
Traffic model	File Transfer Protocol (FTP)
File size	0.5 Mbytes
Minimum distances	Macro - pico: 70 m; Macro - UE: 35 m; Pico - pico: 40 m; Pico - UE: 10 m
Macro power consumption parameters	N_{TRX} = 6, P_0^{m} = 130 W, $P_{\text{max}}^{\text{m}}$ = 20 W
Pico power consumption parameters	N_{TRX} = 2, P_0 = 56 W, P_{max} = 6.3 W, P_{sleep} = 39 W

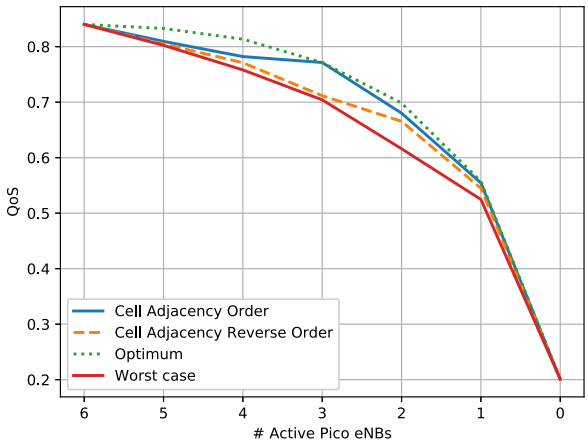

Fig. 4. Performance of the switching on-off policies in terms of QoS for different numbers of active pico eNBs.

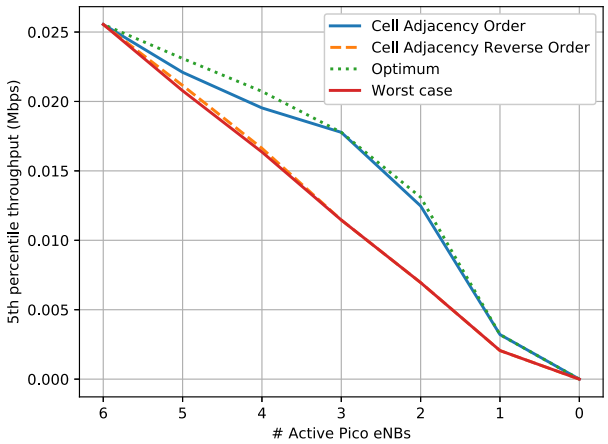

Fig. 5. Performance of the switching on-off policies in terms of 5$^{\text{th}}$ percentile throughput for different numbers of active pico eNBs.

To obtain the numerical results we have set $P = 6$, ABS ratio $= 6/8$ and CRE bias $= 6$ dB. Fig. 4 and 5 show the network performance in terms of QoS (defined in Sec. III-C) and 5$^{\text{th}}$ percentile throughput [42] of the following policies:

- Proposed order: eNBs with lower cell adjacency values are switched off first.
- Reverse order: eNBs with higher values of cell adjacency are switched off first.
- Optimum order: We select the best of the $P!$ possible orderings of P pico eNBs in the sector.
- Worst case order: We select the worst of the $P!$ possible orderings of P pico eNBs in the sector.

Both figures show that the proposed ordering strategy performs similarly to the optimal one. The configuration parameter of the proposed ordering strategy is $w \in [0, 1]$ which has been extensively evaluated by simulation. We found that its best performing value is $w = 0.4$ which has been used in all the numerical experiments of this section.

C. Benchmark Evaluation

In this section, we provide numerical results for our proposal and compare its performance with the following benchmark algorithms operating in the global controller:

- *Oracle* which selects, for each global state $\tilde{s}_k$ at stage k, the optimal control $u_k^* \in \mathcal{U}$, which is found by exhaustive search. Note that the Oracle policy refers to the global controller ($\mathcal{U}$ is the set of global controls) considering near optimal performance of the local controllers. The performance obtained with the oracle is used for measuring the regret of the algorithms evaluated.
- *Default configuration*, which is a fixed control where energy saving and interference mechanisms are deactivated.
- *NeuralBandit* [44] implements a contextual bandit algorithm based on neural networks. [2] It is aimed at learning the cost function $\rho(x, u)$ for each control u (also called *arm* in MAB terminology) given the context x. It comprises, for each arm, a neural network with two hidden layers of 20 units each. ReLU activation function is considered for hidden layers. At each stage, an arm is selected according to an ε-greedy policy with decreasing ε. Then, the selected arm is trained by means of the optimization algorithm Adam [46], using the feedback measures C_k and Q_k. That is, each NN is trained with

[2]Neural networks are implemented using the TensorFlow framework [45].

one sample, by means of backpropagation, every time its associated arm is used.

- *ClassMAB* [11] is our previous proposal for the problem addressed. It comprises a neural network classifier aimed at finding controls satisfying the QoS requirement, and a MAB algorithm in charge of selecting controls with low energy consumption. For the MAB algorithm we proposed a modified version ϵ-greedy in which the set of available controls is variable at each stage.
- *ClassMAB (ES)* refers to the ClassMAB algorithm, but controlling only the energy saving mechanism. This option is included to show the importance of considering interference management when applying energy saving actions.

Other state-of-the-art contextual bandit algorithms, e.g., [10], [14], [47], assume that the expected value of each arm is linear with respect to the context. However, our cost function (6) shows a threshold structure, making these algorithms unsuitable for this application.

Our simulations are aimed at assessing the performance of the algorithms in two phases: a *training* phase composed of 1200 epochs with 200 stages of variable traffic intensities, and a *test* phase where the learning state of the algorithms is frozen setting a greedy policy (i.e., using the control with the lowest expected regret). BRETS, *ClassMAB* and *NeuralBandit* start with an initialization period where each control is selected once and are configured with $\varepsilon_0 = 30$. The exploration budget of BRETS is set to $B = 300$. The QoS threshold is set to $Q_{\text{min}} = 0.6$, and the per-stage cost function is defined as $\rho(s,a) = C(s,a) + \delta \cdot \max(0, Q_{\text{min}} - Q(s,a))$, with $\delta = 10^6$. This definition results convenient because of two reasons: first, the penalty imposed is proportional to the QoS degradation, which is useful in terms of benchmark comparison. And second, according to the simulation results, this ρ complies with the conditions in Section III-D, i.e., under the optimal policy (oracle) the penalty factor is never activated since the QoS constraint is always satisfied. All the results presented in this section are the average of 30 independent simulation runs.

Fig. 6 shows the accumulated regret during the training phase. The slope of the regret curve for BRETS is almost flat, indicating that, in the long term, it selects controls very close to the optimal ones. Besides, its accumulated regret is the lowest, compared with the other benchmarks, which indicates that its learning process is the fastest. The regret slope of *ClassMAB* also approaches zero at the end of the training phase. The accumulated regret of the default configuration is not especially high at the end of the training phase since this policy does not explore, but its regret grows linearly, indicating that this policy does not converge to the optimal one. Figure 7 shows the value of the cost function at each stage. We can observe the fast convergence of BRETS (only a few epochs are needed), and its ability to operate at smaller cost values compared to other benchmarks. Note that, compared to *ClassMAB*, *NeuralBandit* shows a slower learning rate during the first 150 epochs, and a higher long-term cost value. Of these two aspects, the first one is the most relevant in the regret curve for *NeuralBandit* in Fig. 6. The long-term cost

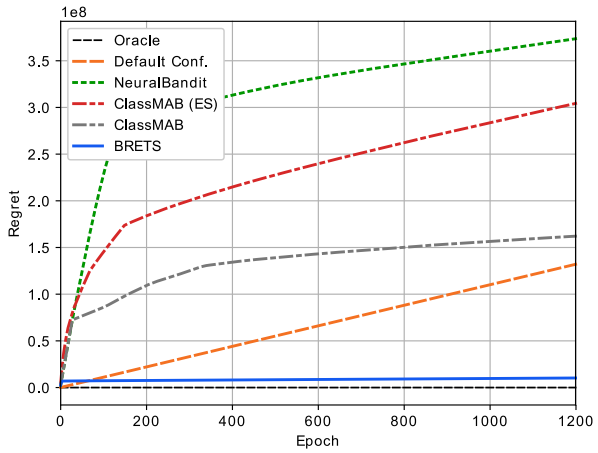

Fig. 6. Regret measured during the training phase. The incurred regret at each epoch is the summation of the regret of each one of its corresponding 200 stages.

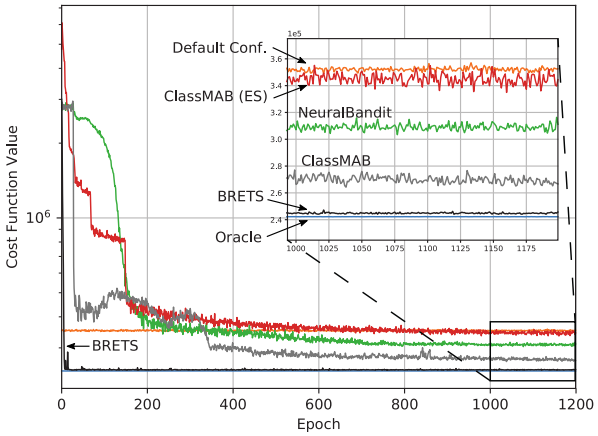

Fig. 7. Evolution of the cost function during the training phase.

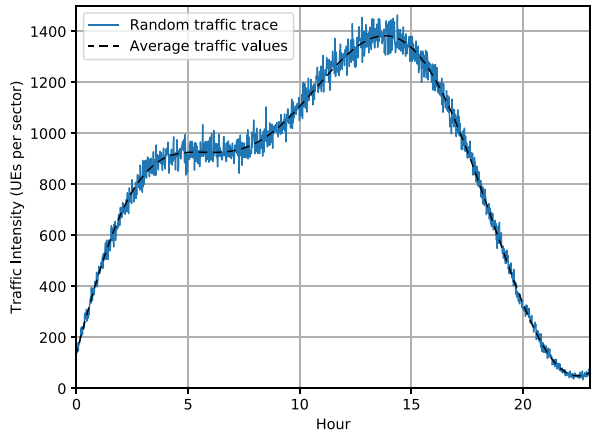

Fig. 8. Traffic pattern of one day considered in the test phase.

value of *ClassMAB (ES)* is higher than BRETS, *ClassMAB* and *NeuralBandit*, which reflects the loss incurred when neglecting the interference coordination mechanism.

We considered a one day period for the test phase, using a stage duration of 10 minutes (i.e., a total of 144 stages). At each stage, a random traffic intensity is generated according to the traffic profile shown in Fig. 8. Note that in the case of special events like sport games or live concerts, the algorithms

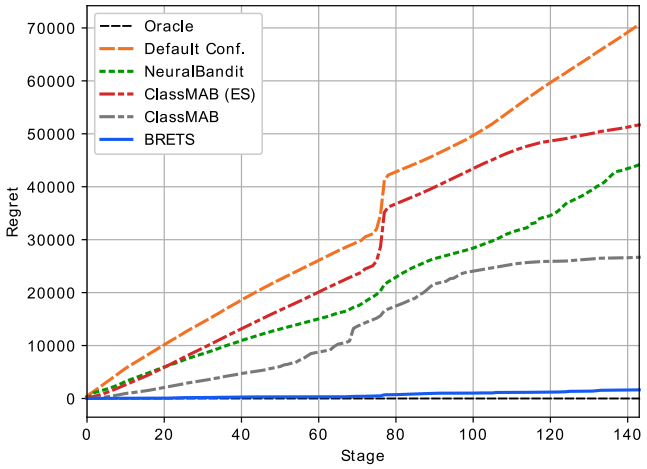

Fig. 9. Evolution of the regret in the test phase.

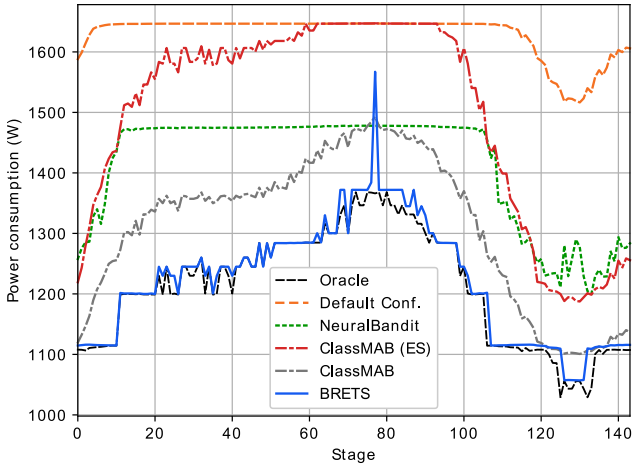

Fig. 10. Power consumption during the test phase.

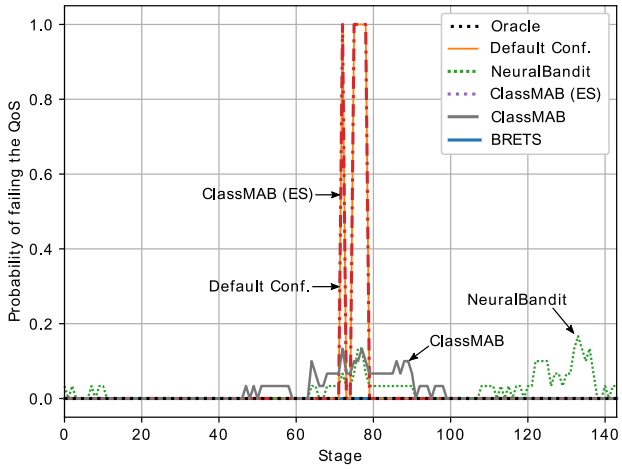

Fig. 11. Probability of failing the QoS requirement at each stage of test phase.

TABLE IV
SUMMARY OF NUMERICAL RESULTS

	Energy saving (%)	Ratio of QoS fulfillment
Oracle	25.701%	1
Default conf.	0%	0.9652
NeuralBandit	13.258%	0.9777
ClassMAB (ES)	8.085%	0.9652
ClassMAB	19.483%	0.9817
BRETS	25.007%	1

could face traffic peak values for which they have not received samples during the training phase. In this situation, BRETS enters the exploration phase (see Algorithm 1) in order to discover which control could better handle this new situation. If the QoS constraints cannot be met, BRETS selects the control closest to this objective. Fig. 9 shows the regret measured during the test phase, during which BRETS obtains the lowest regret, as expected, followed by *ClassMAB* and *NeuralBandit*. Because the regret combines energy consumption and QoS fulfillment, it is interesting to evaluate these two metrics separately for comparing performances. Fig. 10 shows the power consumption at each stage. Note that, in general, the power consumption pattern resembles the traffic profile (Fig. 8). It is clear that the power consumption pattern shown by BRETS is the closest to that of the Oracle. Finally, Fig. 11 shows the estimated probability of failing to satisfy the QoS requirement at each stage of the test phase. Note that during the traffic peak (stages between 70 and 80) there are some stages in which the policies neglecting the IC mechanism are not able to meet the QoS requirement using any control, i.e., they fail with probability 1. This shows again the importance of the joint IC-ES control.

The benchmarks that do not incorporate the interference coordination mechanism, besides consuming more energy, are unable to satisfy the QoS requirement in some stages. This highlights the importance of combining interference coordination and energy saving mechanisms. The numerical results of the test phase are summarized in Table IV where we show the energy savings with respect to the default configuration and the estimated probability of failing the QoS.

VII. CONCLUSION

This paper presented an online learning algorithm for jointly controlling the energy saving and interference coordination mechanisms in HetNets. Our framework considers two levels. At the higher level, a global controller selects global controls applicable to a cluster of contiguous macro cells. At the lower level, local controllers decide how to apply the prescribed global control on each macro cell sector. The proposed learning algorithm, running in the global controller, addresses a constrained contextual bandit problem using a novel approach: for each action, the algorithm handles two simultaneous learning processes, one for the performance objective (network energy consumption) and another for the constraint (QoS fulfillment). Each one of these processes aims at learning a function, instead of a scalar as in conventional bandit algorithms. These functions map the context (network traffic intensity) to the performance metric of interest. For the QoS function, we propose a novel strategy for finding the traffic threshold below which each control fulfills the QoS objective. For estimating the energy consumption functions, the algorithm uses Gaussian processes, which allows the BRETS to balance exploration and exploitation when selecting controls. In our numerical simulations BRETS outperformed other alternatives, attaining an energy saving of 25% and

a QoS fulfillment ratio of 100%, very close to the results obtained by an oracle policy.

Future work includes the following lines. Our proposal could be complemented by an auxiliary learning algorithm for finding clusters of contiguous cells in which joint IC-ES controls can be applied effectively, i.e. for assigning cells to global controllers. BRETS instances should be coordinated among contiguous clusters, in order to avoid the mutual influence caused by the interference at their boundaries. A possible approach would be to alternate the learning periods of adjacent clusters. Finally, it would be interesting to evaluate the potential performance improvement of using one ES control per cell instead of a common ES control for the whole cluster. Because a larger control space is associated to a slower learning rate, this might reveal a tradeoff between long-term performance and convergence time.

APPENDIX A
PROOF OF LEMMA 1

The exploration of a control u ends when this control has been used for the traffic values λ_{th}^u and $\lambda_{\text{th}}^u - \delta$. If only one control u has to be explored ($\mathcal{U} = \{u\}$), we can obtain, by means of an absorbing Markov chain, the following expected convergence time:

$$E[T^u] = \frac{1}{P(\lambda_{\text{th}}^u) + P(\lambda_{\text{th}}^u - \delta)} \left(1 + \frac{P(\lambda_{\text{th}}^u)}{P(\lambda_{\text{th}}^u - \delta)} + \frac{P(\lambda_{\text{th}}^u - \delta)}{P(\lambda_{\text{th}}^u)}\right) \tag{24}$$

where $P(a)$ is short for $P(\lambda_k = a)$. It can be easily checked in (24) that $E[T^u]$ increases when $P(\lambda_{\text{th}}^u) \to 0$, and also when $P(\lambda_{\text{th}}^u - \delta) \to 0$. Since $P_{\min} = \min_{\lambda \in \Lambda} P(\lambda_k = \lambda)$ is the lower bound for both $P(\lambda_{\text{th}}^u)$ and $P(\lambda_{\text{th}}^u - \delta)$ we have

$$\begin{aligned} E[T^u] &\leq \frac{1}{P_{\min} + P_{\min}} \left(1 + \frac{P_{\min}}{P_{\min}} + \frac{P_{\min}}{P_{\min}}\right) \\ &= \frac{3}{2} \frac{1}{P_{\min}}. \end{aligned} \tag{25}$$

Consider a (non-appropriate) exploration strategy that explores the controls in $\mathcal{U}$ sequentially, selecting current u until $|x_k^u| = 0$ before starting to explore the next control. Such strategy has an expected convergence time of $\sum_{u \in \mathcal{U}} E[T^u]$. Now note that $u \in \mathcal{H}_k$, with $\mathcal{H}_k \neq \emptyset$, implies that u has not yet been selected under the current traffic λ_k and therefore $P(\lambda_k = \lambda_{\text{th}}^u) > 0$ and $P(\lambda_k = \lambda_{\text{th}}^u - \delta) > 0$. Indeed, if $u \notin \mathcal{H}_k$, $P(\lambda_k = \lambda_{\text{th}}^u) = P(\lambda_k = \lambda_{\text{th}}^u - \delta) = 0$. Since an appropriate strategy always selects $u \in \mathcal{H}_k$ when $\mathcal{H}_k \neq \emptyset$, the probabilities $P(\lambda_k = \lambda_{\text{th}}^u)$ and $P(\lambda_k = \lambda_{\text{th}}^u - \delta)$ observed at each k by an appropriate strategy are at least as large as those observed by a non-appropriate strategy. As a consequence, the expected convergence time $E[T]$ of an appropriate strategy cannot be larger than $\sum_{u \in \mathcal{U}} E[T^u]$ which, according to (25), is upper bounded by $\frac{3}{2} \frac{|\mathcal{U}|}{P_{\min}}$. □

APPENDIX B
PROOF OF LEMMA 2

This lemma is proved by induction.

Step 1: First, we show that the inequality in (21) holds for $z_k \subseteq x_k$ such that $z_k^1 = \emptyset$ or $z_k^2 = \emptyset$. Let us consider $z_k^1 = \emptyset$ (all the steps are equivalent for $z_k^2 = \emptyset$). First, we need to prove two preliminary inequalities:

1) Inequality 1:
$$E_\lambda \left[J^*(x_k^1, x_k^2, \lambda)\right] > E_\lambda \left[J^*(\emptyset, x_k^2, \lambda)\right] \tag{26}$$
if $x_k^1 \neq \emptyset$. This inequality comes from the fact that reaching a state with $x_{k'}^1 = \emptyset$ from another state with $x_k^1 \neq \emptyset$ takes at least 1 time-slot, therefore $E_\lambda \left[J_\pi(x_k^1, x_k^2, \lambda)\right] \geq 1 + E_\lambda \left[J_\pi(\emptyset, x_k^2, \lambda)\right] > E_\lambda \left[J_\pi(\emptyset, x_k^2, \lambda)\right]$.
2) Inequality 2:
$$E_\lambda \left[J^*(\emptyset, x_k^2, \lambda)\right] > E_\lambda \left[J^*(\emptyset, z_k^2, \lambda)\right] \tag{27}$$
if $z_k^2 \subsetneq x_k^2$. For any x^2 we have that
$$\begin{aligned} &E_\lambda \left[J^*(\emptyset, x^2, \lambda)\right] \\ &= P_1(x^2) \frac{1}{P(\lambda_{\text{th}}^2)} + P_2(x^2) \frac{1}{P(\lambda_{\text{th}}^2 - \delta)} + P_3(x^2) E[T^2] \end{aligned} \tag{28}$$
where
$$\begin{aligned} P_1(x^2) &= P(\lambda_{\text{th}}^2 \in x^2, \lambda_{\text{th}}^2 - \delta \notin x^2) \\ P_2(x^2) &= P(\lambda_{\text{th}}^2 \notin x^2, \lambda_{\text{th}}^2 - \delta \in x^2) \\ P_3(x^2) &= P(\lambda_{\text{th}}^2 \in x^2, \lambda_{\text{th}}^2 - \delta \in x^2) \end{aligned} \tag{29}$$
and $E[T^2]$ is given by (24). If $z^2 \subsetneq x^2$, then we have $\sum_{i=1}^3 P_i(x^2) > \sum_{i=1}^3 P_i(z^2)$, which results in
$$P_3(x^2) > P_3(z^2) + \sum_{i=1}^{2} (P_i(z^2) - P_i(x^2)) \tag{30}$$
We can now obtain (27) for $z^2 \subsetneq x^2$ as follows
$$\begin{aligned} &E_\lambda \left[J^*(\emptyset, x^2, \lambda)\right] \\ &= \frac{P_1(x^2)}{P(\lambda_{\text{th}}^2)} + \frac{P_2(x^2)}{P(\lambda_{\text{th}}^2 - \delta)} + P_3(x^2) E[T^2] \\ &> \frac{P_1(x^2)}{P(\lambda_{\text{th}}^2)} + \frac{P_2(x^2)}{P(\lambda_{\text{th}}^2 - \delta)} + P_3(z^2) E[T^2] \\ &\quad + (P_2(z^2) - P_2(x^2)) E[T^2] \\ &\quad + (P_1(z^2) - P_1(x^2)) E[T^2] \\ &\geq \frac{P_1(x^2)}{P(\lambda_{\text{th}}^2)} + \frac{P_2(x^2)}{P(\lambda_{\text{th}}^2 - \delta)} + P_3(z^2) E[T^2] \\ &\quad + \frac{P_2(z^2) - P_2(x^2)}{P(\lambda_{\text{th}}^2 - \delta)} + \frac{P_1(z^2) - P_1(x^2)}{P(\lambda_{\text{th}}^2)} \\ &= \frac{P_1(z^2)}{P(\lambda_{\text{th}}^2)} + \frac{P_2(z^2)}{P(\lambda_{\text{th}}^2 - \delta)} + P_3(z^2) E[T^2] \\ &= E_\lambda \left[J^*(\emptyset, z^2, \lambda)\right] \end{aligned} \tag{31}$$
where the first inequality comes from (30), and the second inequality comes from the fact that $E[T^2] \geq \frac{1}{P(\lambda_{\text{th}}^2)}$ and $E[T^2] \geq \frac{1}{P(\lambda_{\text{th}}^2 - \delta)}$.

Bellman's equation (19) for z_k is
$$J^*(\emptyset, z_k^2, \lambda) = 1 + E[J^*(\emptyset, z_{k+1}^2, \lambda) | z_k, \lambda_k]. \tag{32}$$

By inequality 1 we have that

$$E[J^*(x_{k+1}^1, x_k^2, \lambda)\big|x_k, \lambda_k] \geq E[J^*(\emptyset, x_k^2, \lambda)\big|x_k, \lambda_k]$$
$$E[J^*(x_k^1, x_{k+1}^2, \lambda)\big|x_k, \lambda_k] > E[J^*(\emptyset, x_{k+1}^2, \lambda)\big|x_k, \lambda_k] \quad (33)$$

and by inequality 2 we have that

$$E[J^*(\emptyset, x_k^2, \lambda)\big|x_k, \lambda_k] > E[J^*(\emptyset, z_{k+1}^2, \lambda)\big|x_k, \lambda_k]$$
$$E[J^*(\emptyset, x_{k+1}^2, \lambda)\big|x_k, \lambda_k] \geq E[J^*(\emptyset, z_{k+1}^2, \lambda)\big|x_k, \lambda_k]. \quad (34)$$

Therefore $J^*(x_k^1, x_k^2, \lambda_k) \geq J^*(\emptyset, z_k^2, \lambda_k)$.

Step 2: The induction step consists of showing that if the inequality (21) holds for uncertainty region vectors contained in x_k, then it holds for x_k, i.e., we assume that given z_k such that $z_k \subseteq x_k$, the inequality $J^*(z_k, \lambda_k) \geq J^*(y_k, \lambda_k)$ holds for any $y_k \subseteq z_k$, and then we show that this implies $J^*(x_k, \lambda_k) \geq J^*(z_k, \lambda_k)$.

Because $z_k \subseteq x_k$, we have $(\bar{z}_{k+1}^1, z_k^2) \subseteq (\bar{x}_{k+1}^1, x_k^2)$, $(\underline{z}_{k+1}^1, z_k^2) \subseteq (\underline{x}_{k+1}^1, x_k^2)$, $(z_k^1, \bar{z}_{k+1}^2) \subseteq (x_k^1, \bar{x}_{k+1}^2)$, and $(z_k^1, \underline{z}_{k+1}^2) \subseteq (x_k^1, \underline{x}_{k+1}^2)$, where the vectors in the right hand side of $\subseteq$ are contained in x_k (thus equivalent to z_k in the induction assumption), and the vectors in the left hand side of $\subseteq$ are contained in z_k (thus equivalent to y_k in the induction assumption). Therefore, by the induction assumption we have

$$E_\lambda[J^*(\bar{x}_{k+1}^1, x_k^2, \lambda)] \geq E_\lambda[J^*(\bar{z}_{k+1}^1, z_k^2, \lambda)]$$
$$E_\lambda[J^*(\underline{x}_{k+1}^1, x_k^2, \lambda)] \geq E_\lambda[J^*(\underline{z}_{k+1}^1, z_k^2, \lambda)]$$
$$E_\lambda[J^*(x_k^1, \bar{x}_{k+1}^2, \lambda)] \geq E_\lambda[J^*(z_k^1, \bar{z}_{k+1}^2, \lambda)]$$
$$E_\lambda[J^*(x_k^1, \underline{x}_{k+1}^2, \lambda)] \geq E_\lambda[J^*(z_k^1, \underline{z}_{k+1}^2, \lambda)] \quad (35)$$

Because $P(\lambda_k \geq \lambda_{\text{th}}^u)$ and $P(\lambda_k < \lambda_{\text{th}}^u)$ are independent of the uncertainty vectors, the above inequalities result in

$$E[J^*(x_{k+1}^1, x_k^2, \lambda)\big|x_k, \lambda_k] \geq E[J^*(z_{k+1}^1, z_k^2, \lambda)\big|x_k, \lambda_k]$$
$$E[J^*(x_k^1, x_{k+1}^2, \lambda)\big|x_k, \lambda_k] \geq E[J^*(z_k^1, z_{k+1}^2, \lambda)\big|x_k, \lambda_k] \quad (36)$$

and therefore $J^*(x_k, \lambda_k) \geq J^*(z_k, \lambda_k)$. □

REFERENCES

[1] N. Bhushan *et al.*, "Network densification: The dominant theme for wireless evolution into 5G," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 82–89, Feb. 2014.

[2] L. Saker, S. E. Elayoubi, and H. O. Scheck, "System selection and sleep mode for energy saving in cooperative 2G/3G networks," in *Proc. IEEE 70th Veh. Technol. Conf. Fall*, Anchorage, AK, USA, Sep. 2009, pp. 1–5.

[3] *Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Overall Description (Release 10)*, document Tech. Spec. 36.300 v10.5.0, 3rd Generation Partnership Project (3GPP), 2011.

[4] J. A. Ayala-Romero, J. J. Alcaraz, and J. Vales-Alonso, "Energy saving and interference coordination in HetNets using dynamic programming and CEC," *IEEE Access*, vol. 6, pp. 71110–71121, 2018.

[5] A. Zakrzewska, L. Ho, H. Gacanin, and H. Claussen, "Coordination of SON functions in multi-vendor femtocell networks," *IEEE Commun. Mag.*, vol. 55, no. 7, pp. 165–171, Jul. 2017.

[6] O.-C. Iacoboaiea, B. Sayrac, S. B. Jemaa, and P. Bianchi, "SON coordination in heterogeneous networks: A reinforcement learning framework.," *IEEE Trans. Wireless Commun.*, vol. 15, no. 9, pp. 5835–5847, Sep. 2016.

[7] J. A. Ayala-Romero, J. J. Alcaraz, J. Vales-Alonso, and E. Egea-López, "Online optimization of interference coordination parameters in small cell networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 10, pp. 6635–6647, Oct. 2017.

[8] J. A. Ayala-Romero, J. J. Alcaraz, and J. Vales-Alonso, "Data-driven configuration of interference coordination parameters in HetNets," *IEEE Trans. Veh. Technol.*, vol. 67, no. 6, pp. 5174–5187, Jun. 2018.

[9] S. Maghsudi and E. Hossain, "Multi-armed bandits with application to 5G small cells," *IEEE Wireless Commun.*, vol. 23, no. 3, pp. 64–73, Jun. 2016.

[10] L. Li, W. Chu, J. Langford, and R. E. Schapire, "A contextual-bandit approach to personalized news article recommendation," in *Proc. 19th Int. Conf. World Wide Web*, Apr. 2010, pp. 661–670.

[11] J. A. Ayala-Romero, J. J. Alcaraz, A. Zanella, and M. Zorzi, "Contextual bandit approach for energy saving and interference coordination in HetNets," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Kansas City, MO, USA, May 2018, pp. 1–6.

[12] *System Performance of Heterogeneous Networks With Range Expansion*, document 3GPP R1-100142, 3rd Generation Partnership Project (3GPP), 2010.

[13] P. Auer, N. Cesa-Bianchi, and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Mach. Learn.*, vol. 47, no. 2, pp. 235–256, 2002.

[14] P. Auer, "Using confidence bounds for exploitation-exploration tradeoffs," *J. Mach. Learn. Res.*, vol. 3, pp. 397–422, Nov. 2003.

[15] S. Cai, Y. Che, L. Duan, J. Wang, S. Zhou, and R. Zhang, "Green 5G heterogeneous networks through dynamic small-cell operation," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1103–1115, May 2016.

[16] E. Oh, K. Son, and B. Krishnamachari, "Dynamic base station switching-on/off strategies for green cellular networks," *IEEE Trans. Wireless Commun.*, vol. 12, no. 5, pp. 2126–2136, May 2013.

[17] M. Feng, S. Mao, and T. Jiang, "BOOST: Base station on-off switching strategy for green massive MIMO HetNets," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7319–7332, Nov. 2017.

[18] J. He *et al.*, "Energy efficient BSs switching in heterogeneous networks: An operator's perspective," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Doha, Qatar, Apr. 2016, pp. 1–6.

[19] J. Kim, W. S. Jeon, and D. G. Jeong, "Base-station sleep management in open-access femtocell networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 5, pp. 3786–3791, May 2016.

[20] L. Tang, W. Wang, Y. Wang, and Q. Chen, "An energy-saving algorithm with joint user association, clustering, and on/off strategies in dense heterogeneous networks," *IEEE Access*, vol. 5, pp. 12988–13000, 2017.

[21] T. Zhou, N. Jiang, Z. Liu, and C. Li, "Joint cell activation and selection for green communications in ultra-dense heterogeneous networks," *IEEE Access*, vol. 6, pp. 1894–1904, 2018.

[22] B. Zhuang, D. Guo, and M. L. Honig, "Energy-efficient cell activation, user association, and spectrum allocation in heterogeneous networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 4, pp. 823–831, Apr. 2016.

[23] Q. Kuang and W. Utschick, "Energy management in heterogeneous networks with cell activation, user association, and interference coordination," *IEEE Trans. Wireless Commun.*, vol. 15, no. 6, pp. 3868–3879, Jun. 2016.

[24] C. Liu, B. Natarajan, and H. Xia, "Small cell base station sleep strategies for energy efficiency," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1652–1661, Mar. 2016.

[25] A. Virdis, G. Stea, D. Sabella, and M. Caretti, "A distributed power-saving framework for LTE HetNets exploiting Almost Blank Subframes," *IEEE Trans. Green Commun. Netw.*, vol. 1, no. 3, pp. 235–252, Sep. 2017.

[26] X. Xu, C. Yuan, W. Chen, X. Tao, and Y. Sun, "Adaptive cell zooming and sleeping for green heterogeneous ultra-dense networks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 2, pp. 1612–1621, Feb. 2017.

[27] S. Zhang, N. Zhang, S. Zhou, J. Gong, Z. Niu, and X. Shen, "Energy-aware traffic offloading for green heterogeneous networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1116–1129, May 2016.

[28] X. Chen, J. Wu, Y. Cai, H. Zhang, and T. Chen, "Energy-efficiency oriented traffic offloading in wireless networks: A brief survey and a learning approach for heterogeneous cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 4, pp. 627–640, Apr. 2015.

[29] L. Saker, S. E. Elayoubi, R. Combes, and T. Chahed, "Optimal control of wake up mechanisms of femtocells in heterogeneous networks," *IEEE J. Sel. Areas Commun.*, vol. 30, no. 3, pp. 664–672, Apr. 2012.

[30] U. Siddique, H. Tabassum, E. Hossain, and D. I. Kim, "Channel-access-aware user association with interference coordination in two-tier downlink cellular networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 7, pp. 5579–5594, Jul. 2016.

[31] J. Zheng, L. Gao, H. Wang, J. Niu, X. Li, and J. Ren, "EE-eICIC: Energy-efficient optimization of joint user association and ABS for eICIC in heterogeneous cellular networks," *Wireless Commun. Mobile Comput.*, vol. 2017, Sep. 2017, Art. no. 6768415.

[32] I.-S. Comsa, A. De-Domenico, and D. Ktenas, "QoS-driven scheduling in 5G radio access networks - a reinforcement learning approach," in *Proc. GLOBECOM IEEE Global Commun. Conf.*, Dec. 2017, pp. 1–7.
[33] I.-S. Comşa *et al.*, "Towards 5G: A reinforcement learning-based scheduling solution for data traffic management," *IEEE Trans. Netw. Service Manage.*, vol. 15, no. 4, pp. 1661–1675, Dec. 2018.
[34] M. Simsek, M. Bennis, and I. Güvenç, "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.
[35] Y. S. Soh, T. Q. S. Quek, M. Kountouris, and H. Shin, "Energy efficient heterogeneous cellular networks," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 5, pp. 840–850, May 2013.
[36] A. R. Khamesi and M. Zorzi, "Energy harvesting and cell zooming in K- tier heterogeneous random cellular networks," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 1, pp. 63–73, Mar. 2018.
[37] S. Müller, O. Atan, M. van der Schaar, and A. Klein, "Smart caching in wireless small cell networks via contextual multi-armed bandits," in *Proc. IEEE Int. Conf. Commun. (ICC)*, May 2016, pp. 1–7.
[38] M. Hashemi, A. Sabharwal, C. E. Koksal, and N. B. Shroff, "Efficient beam alignment in millimeter wave systems using contextual bandits," in *Proc. INFOCOM IEEE Conf. Comput. Communi.*, Apr. 2018, pp. 2393–2401.
[39] *Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Study on Energy Saving Enhancement for E-UTRAN (Release 12)*, document 3GPP TR 36.887, 3rd Generation Partnership Project (3GPP), 2014.
[40] C. E. Rasmussen and C. K. Williams, *Gaussian Processes for Machine Learning*. Cambridge, MA, USA: MIT Press, 2006.
[41] D. P. Bertsekas, *Dynamic Programming and Optimal Control*, vol. 2. Belmont, MA, USA: Athenea Scientific, 2012.
[42] *Evolved Universal Terrestrial Radio Access (EUTRA); Further Advancements for E-UTRA Physical Layer Aspects*, document 3GPP TR 36.814, 3rd Generation Partnership Project (3GPP), 2010.
[43] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.
[44] R. Allesiardo, R. Féraud, and D. Bouneffouf, "A neural networks committee for the contextual bandit problem," in *Proc. Int. Conf. Neural Inf. Process.*, 2014, pp. 374–381.
[45] M. Abadi *et al.*, "TensorFlow: Large-scale machine learning on heterogeneous distributed systems," *CoRR*, vol. abs/1603.04467, 2016. [Online]. Available: https://dblp.org/rec/bib/journals/corr/Abadi ABBCCCDDDG16
[46] D. P. Kingma and J. Ba. (2014). "Adam: A method for stochastic optimization." [Online]. Available: https://arxiv.org/abs/1412.6980
[47] O. Chapelle and L. Li, "An empirical evaluation of Thompson sampling," in *Proc. Adv. Neural Inf. Process. Syst.*, 2011, pp. 2249–2257.

Jose A. Ayala-Romero received the B.Sc. degree in telematics engineering and the M.Sc. degree in telecommunication engineering from the Technical University of Cartagena (UPCT), Spain, in 2012 and 2014, respectively, where he is currently pursuing the Ph.D. degree with the Department of Information and Communications Technologies. He was a Visiting Ph.D. Student with the Department of Information Engineering, University of Padova, in 2017, and NEC Laboratories Europe, Germany, in 2018. His current research focuses on machine learning algorithms for wireless networks management.

Juan J. Alcaraz received the M.Sc. degree in telecommunications engineering from the Polytechnic University of Valencia, Spain, in 1999, and the Ph.D. degree in telecommunications engineering from the Technical University of Cartagena (UPCT), Spain, in 2007. From 1999 to 2004, he was with several technology companies until joining UPCT in 2004, where he is currently an Associate Professor with the Department of Information and Communication Technologies. He was a Fulbright Visiting Scholar with the Electrical Engineering Department, University of California at Los Angeles (UCLA), in 2013, and a Visiting Professor with the Department of Information Engineering, University of Padova, in 2017. His current research focuses on learning algorithms for wireless networks.

Andrea Zanella (S'98–M'01–SM'13) received the Laurea degree in computer engineering and the Ph.D. degree in electronic and telecommunications engineering from the University of Padova, Italy, in 1998 and 2001, respectively. In 2000, he was a Visiting Scholar with the Department of Computer Science, University of California at Los Angeles (UCLA). He is one of the coordinators of the SIGnals and NETworking (SIGNET) Research Lab. He is currently an Associate Professor with the Department of Information Engineering (DEI), University of Padova. His long-established research activities are in the fields of protocol design, optimization, and performance evaluation of wired and wireless networks. He is also a Technical Area Editor of the IEEE INTERNET OF THINGS JOURNAL and an Associate Editor of the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING, IEEE COMMUNICATIONS SURVEYS AND TUTORIALS, and *Digital Communications and Networks* (DCN).

Michele Zorzi (F'07) received the Laurea and Ph.D. degrees in electrical engineering from the University of Padova in 1990 and 1994, respectively. From 1992 to 1993, he was on leave at the University of California at San Diego (UCSD). In 1993, he joined the Faculty of the Dipartimento di Elettronica e Informazione, Politecnico di Milano, Italy. After spending three years with the Center for Wireless Communications, UCSD, he joined the School of Engineering, University of Ferrara, Italy, in 1998, where he became a Professor in 2000. Since 2003, he has been with the Faculty of the Information Engineering Department, University of Padova. His current research interests include performance evaluation in mobile communications systems, WSN and Internet of Things, cognitive communications and networking, vehicular networks, 5G mm-wave cellular systems, and underwater communications and networks. He was a recipient of several awards from the IEEE Communications Society, including the Best Tutorial Paper Award in 2008, the Education Award in 2016, and the Stephen O. Rice Best Paper Award in 2018. He was the Editor-In-Chief of the IEEE WIRELESS COMMUNICATIONS from 2003 to 2005, the IEEE TRANSACTIONS ON COMMUNICATIONS from 2008 to 2011, and the IEEE TRANSACTIONS ON COGNITIVE COMMUNICATIONS AND NETWORKING from 2014 to 2018. He has served as a Member-at-Large for the Board of Governors of the IEEE Communications Society from 2009 to 2011 and the Director of Education from 2014 to 2015.