This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing

Offloading in HetNet: A Coordination of Interference Mitigation, User Association and Resource Allocation

Thant Zin Oo, Nguyen H. Tran, Walid Saad, Dusit Niyato, Zhu Han, and Choong Seon Hong

***Abstract*—The use of heterogeneous small cell-based networks to offload the traffic of existing cellular systems has recently attracted significant attention. One main challenge is solving the joint problems of interference mitigation, user association and resource allocation. These problems are formulated as an optimization which is then analyzed using two different approaches: Markov approximation and log-linear learning. However, finding the optimal solutions of both approaches requires complete information of the whole network which is not scalable with the network size. Thus, an approach based on a Markov approximation with a novel Markov chain design and transition probabilities is proposed. This approach enables the Markov chain to converge to the bounded near optimal distribution without complete information. In the game-theoretic approach, the payoff-based log-linear learning is used, and it converges in probability to a mixed-strategy ϵ–Nash equilibrium. Based on the principles of these two approaches, a highly randomized self-organizing algorithm is proposed to reduce the gap between optimal and converged distributions. Simulation results show that all the proposed algorithms effectively offload more than 90% of the traffic from the macrocell base station to small cell base stations. Moreover, the results also show the the algorithms converge quickly irrespective of the number of possible configurations.**

***Index Terms*—Heterogeneous Cellular Networks, HetNets, Interference Mitigation, User Association, Resource Allocation.**

I. Introduction

The demand for wireless data traffic has increased considerably in the past decade and is expected to continue to grow in the near future. However, mobile operator revenues are flattening due to saturated markets, flat-rate tariffs and competitive and regulatory pressure [1]. This decoupling of network traffic and operator revenue has led the mobile operators to increase the network efficiency in order to maximize their revenue. One viable solution is the deployment of multi-tier dense small cell base stations (SBSs) overlaid on the existing macrocells. Economically, deploying and operating SBSs cost only a small fraction of the macrocell base stations (MBSs) in terms of both CAPEX and OPEX.

The major challenges for small cell based heterogeneous networks (HetNets) are interference mitigation, user association and resource allocation problems [2]–[7]. Unlike classical wireless networks, in HetNets, the number of choices or configurations increases exponentially with the number of deployed SBSs. Thus, existing centralized resource management algorithms such as in [8] and [9] can no longer cope with the massive overhead in computation and signaling required by the HetNet small cells. This challenge is further exacerbated by the fact that these problems are coupled and must be solved simultaneously. This problem becomes non-trivial when coordination and tradeoff are necessary between the competing interests of users and base stations (BSs). To address this problem, one must design self-organizing algorithms that can enable a small cell network to operate in a distributed manner and with small overhead [10]. Using self-organization, small cells can learn from their environment and autonomously adjust their configuration strategies towards achieving the optimal performance. More importantly, self-organizing mechanisms can be implemented distributedly without complete information, and thus, are scalable with the network size [2].

A. Related Work

Many works have recently developed self-organization algorithms for HetNets based on game-theoretic approaches to analyze user association, resource allocation, power control, interference mitigation, spectrum reuse, network selection and/or admission control [2]–[4], [11]–[15]. However, most of the existing literature focused on one such problem in isolation without considering them jointly [2], [4], [12], [14], [15]. Moreover, none of these approaches considered the balance between the *exploration* and *exploitation* of the learning approaches used in self-organization. In [2], the authors used the potential game approach to reach a self-organizing solution for power control. In [11], the authors employed a semi-Markov decision process to study the admission control problem and designed a power control game to reduce energy consumption. The authors in [12], [14] developed a coalition game approach to tackle interference mitigation and reached self-organizing solutions that can achieve stable network partitions. In [13], the authors studied the dynamic matching game and proposed distributed algorithms for joint

T. Z. Oo, N. H. Tran and C. S. Hong are with the Department of Computer Science and Engineering, Kyung Hee University, Korea (email: {tzoo, nguyenth, cshong}@khu.ac.kr).

W. Saad is with Bradley Department of Electrical and Computer Engineering, Virginia Tech. He was also an International Scholar at the Department of Computer Science and Engineering, Kyung Hee University, Korea (email: walids@vt.edu).

D. Niyato is with School of Computer Science and Engineering, Nanyang Technological University (email:dniyato@ntu.edu.sg).

Z. Han is with the Electrical and Computer Engineering Department, University of Houston, Houston, Texas, USA (email: zhan2@uh.edu).

Dr. C. S. Hong is the corresponding author.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing

user association and resource allocation for femto access points. Similarly, in [15], the authors formulated the resource allocation problem as a cooperative game which jointly performs user association and resource allocation such that the total satisfaction of the users is maximized.

Beyond game theory, the authors in [5] studied the joint resource allocation and power control problem. They proposed an optimal exhaustive algorithm and its corresponding sub-optimal distributed low-complexity algorithm. In [6], the authors focused on resource allocation and inter-cell interference mitigation and formulated the optimization problem as a low-complexity linear programming. In [7], the authors considered a success probability as a QoS constraint and formulated a throughput maximization problem to find the optimal spectrum allocation. The authors in [16] explored up-link scheduling and power allocation problem and used an approximation method to arrive at a sub-optimal solution. The work in [17] studied the spectrum and energy efficiency in HetNets in which the authors quantified the tradeoff between spectrum and energy efficiency as a Lebesgue measure. In [18], the authors studied the backhaul as a bottleneck in HetNets and characterized the behavior of delay and deployment cost. The work in [19] used learning for turning BSs on and off, but it does not look at resource allocation. Although the works in [3], [5]–[7], [11], [13], [16]–[18] considered joint problems, they did not study the self-organization and learning aspects which are critical for future deployment of dense small cell networks. In [20], the authors modeled the energy efficiency in traffic offloading of HetNets as a discrete-time Markov decision process and used Q-learning with compact state representation algorithm to achieve self-organization. The work in [20] considered both the joint problems and learning aspects of HetNets, but did not analyze the learning efficiency and the mixing characteristics of the underlying Markov chain.

Recently, the use of Markov approximation [21] was proposed to solve a number of combinatorial optimization problem. In [21], the authors presented three use cases, (i) utility maximization in CSMA networks, (ii) path selection in wire-line networks and (iii) channel assignment in wireless LANs. In [22], the authors applied Markov approximation to search for the optimal peer-to-peer (P2P) network configuration distributively for video streaming applications. Furthermore, the Markov approximation was also employed in [23] for the joint virtual machine placement and routing in data-centers. However, these existing works do not take into account the fact that the UEs are only interested in maximizing their individual utilities whereas the BSs are concerned with minimizing their total costs which, in a distributed system, can lead to the detrimental performance if not properly modeled.

B. Contributions

We formulate traffic offloading as a joint optimization problem (JOP) of interference mitigation, user association and resource allocation under QoS guarantee which is combinatorial and has a very large solution space. The main goal of this paper is to address JOP by combining different perspectives from Markov approximation and non-cooperative game theory. The Markov approximation has its roots in convex optimization [24]. The drawback of Markov approximation is that it ignores the strategic behaviors of UEs and BSs. On the other hand, log-linear learning is used to find the equilibrium of a noncooperative game [25]. The advantage of a noncooperative game is that we can model the competitions and strategic interactions between all the UEs and BSs. The drawback of the noncooperative game is that it can reach inefficient equilibrium solutions. However, both methods share the same approach of using an underlying Markov chain to yield probabilistic solutions. Based on these two methods, our contributions are summarized as follows:

- To solve JOP, we first analyze it using Markov approximation. We then design an algorithm, called Markov chain Directed Algorithm (MIDA), which allows UEs and BSs to self-configure the network according to the transition probabilities of the underlying Markov chain. The self-configuration involves two steps, *exploration* in one time slot followed by *consolidation* in subsequent time slot. The *exploration–consolidation* steps are repeated until MIDA converges to a bounded near-optimal solution. However, the performance gap of MIDA is still quite large due to the lack of randomness in *exploration–consolidation* steps.
- To overcome some of the drawbacks of MIDA, we propose a log-linear algorithm to solve JOP. We formulate the problem as a noncooperative game where UEs and BSs are joint players who self-configure the network. To solve the formulated game, we design a distributed algorithm, called Payoff-based log-linear Learning Algorithm (POLA), which takes into account the strategic interactions between players and introduces randomness into the algorithmic structure with a Bernoulli process.
- We decrease the optimality gap between converged and optimal distributions in POLA via the design of a highly randomized algorithm called Randomized Self-organized Algorithm (ROSE). Due to the highly randomized algorithmic structure, ROSE outperforms both MIDA and POLA in terms of utility.
- Simulation results show that all of our proposed algorithms can effectively offload the traffic from MBS to SBSs. The results also show that MIDA and POLA converge quickly irrespective of the size of feasible solution space of JOP. Furthermore, ROSE yields the best performance in terms of utility, and at the same time, have a bounded expected stopping time irrespective of the number of BSs and UEs in the network.

The rest of the paper is organized as follows: In Section II, we present our system model. We formulate the problem in Section III. We present MIDA in Section IV. We formulate the noncooperative game in Section V and present POLA and ROSE in Section V-B and Section V-C, respectively. We present our simulation results in Section VI, and we finally conclude this paper in Section VII.

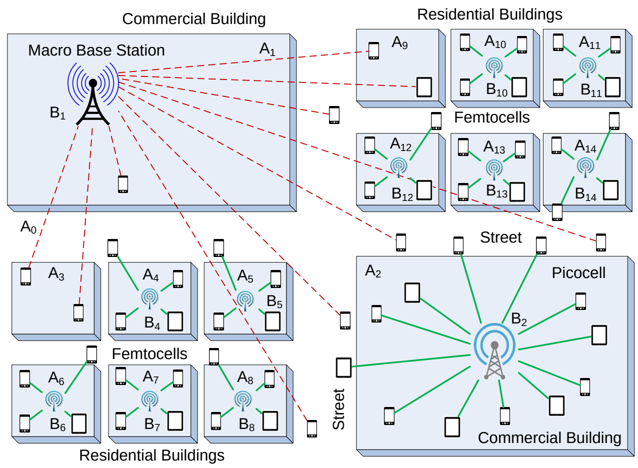


Fig. 1: An example of a three-tier HetNet including one macrocell BS: $\{B_1\}$, one picocell BS: $\{B_2\}$, and ten femtocell BSs: $\{B_4, \ldots, B_8, B_{10}, \ldots, B_{14}\}$. The picocell and femtocell BSs are usually located at buildings that constitute hotspots for wireless traffic. Some buildings have no SBSs and the UEs located within these buildings will be served by the MBS. For example, buildings A_6 and A_9 are served by the MBS. Furthermore, UEs in an outdoor region, such as A_0, are either served by MBS or SBSs.

II. System Model

Consider the downlink of a HetNet consisting of fixed BSs and randomly located UEs as illustrated in Fig. 1. The area shown in Fig.1 is covered by three-tiers of BSs: macrocell base station (MBS) tier, picocell base station (PBS) tier, and femtocell base station (FBS) tier. The coverage area of those BSs will be overlapping, and each UE is in the range of at least one BS. The set of BSs is denoted by $\mathcal{B}$. Let $\mathcal{S}$ be the set of available sub-channels that each BS $m \in \mathcal{B}$ can use. These sub-channels will be further divided and allocated to the UEs associated to each BS m. We assume that each BS $m \in \mathcal{B}$ transmits with a constant per sub-channel transmit power P_{mi}^k on sub-channel k, and total transmit power of BS m is $\widehat{P}_m = \sum_{k\in\mathcal{S}} P_{mi}^k$. All BSs are connected to a high speed backhaul with negligible delay (such as a high speed fiber). Let $\mathcal{U}$ be the set of UEs located inside region $\mathcal{A}$ and $\psi_i \in \Psi$ be the requested downlink rate (bits per second) of UE i, where Ψ is the discrete set of service classes[1].

A. Data Rate and QoS

In this network, we consider a log-distance path loss model, and the positive channel power gain between UE i and BS m can be calculated as: $h_{mi} = 10^{-\mu/10}$, where μ is the total path loss between BS m and UE i in decibels (dB). We assume that each UE i is capable of measuring h_{mi} for all BSs $m \in \mathcal{B}$. Let $\mathcal{B}^k \subseteq \mathcal{B}$ denote the set of BSs that use sub-channel k. Then, the interference to BS m on sub-channel k is $\sum_{n\in\mathcal{B}^k\setminus\{m\}} h_{ni} P_{ni}^k$. Hence, the instantaneous signal-to-interference-plus-noise-ratio (SINR) received at UE i from BS m on sub-channel k is:

$$\Gamma_{mi}^k = \frac{h_{mi}\, P_{mi}^k}{\sum_{n\in\mathcal{B}^k\setminus\{m\}} h_{ni}\, P_{ni}^k + W\, N_0}, \tag{1}$$

where W is the bandwidth of the sub-channel and N_0 is the thermal noise spectral power.

Accordingly, the achievable per sub-channel downlink rate from BS m to UE i is

$$R_{mi}^k = W\, \log_2(1+\Gamma_{mi}^k). \tag{2}$$

Then, the downlink rate achieved by UE i will be given by:

$$\widehat{R}_i = \textstyle\sum_{m\in\mathcal{B}} x_i^m \sum_{k\in\mathcal{S}} y_{mi}^k\, R_{mi}^k, \tag{3}$$

where $x_i^m \in \{0,1\}$ and $y_{mi}^k \in \{0,1\}$ are the binary decision variables used for user association and sub-channel (resource) allocation, respectively. In other words, $x_i^m = 1$ if UE i is associated with BS m and $y_{mi}^k = 1$ if sub-channel k is allocated to the downlink from BS m to UE i, and vice versa. Note that UE i can only associate with at most one BS at any time instance, i.e.,

$$\textstyle\sum_{m\in\mathcal{B}} x_i^m \leq 1, \quad \forall i \in \mathcal{U}. \tag{4}$$

A BS must serve its associated UEs with a minimum QoS requirement, i.e., $\widehat{R}_i \geq \psi_i, \ \forall i \in \mathcal{U}$. Hence, the QoS constraint will be given by:

$$\textstyle\sum_{m\in\mathcal{B}} x_i^m \sum_{k\in\mathcal{S}} y_{mi}^k\, R_{mi}^k \geq \psi_i, \quad \forall i \in \mathcal{U}. \tag{5}$$

To satisfy (5), BS m must allocate minimum number of sub-channels to UE i. Assuming that R_{mi}^k is the same across all sub-channels, we have $\sum_{k\in\mathcal{S}} y_{mi}^k = \lceil \psi_i / R_{mi}^k \rceil$, where $\sum_{k\in\mathcal{S}} y_{mi}^k$ represents the total number of allocated sub-channels, and $\lceil\cdot\rceil$ denotes the ceiling function.

TABLE I: Table of Notations

Notation	Description
$\mathcal{U}$, $\lvert\mathcal{U}\rvert$, i, j	set, cardinality and indexes of UEs
$\mathcal{B}$, $\lvert\mathcal{B}\rvert$, m, n	set, cardinality and indexes of BSs
$\mathcal{S}$, $\lvert\mathcal{S}\rvert$, k	set, cardinality and index of sub-channels (sCHs)
$\mathcal{B}_{\text{conflict}}$, $\mathcal{B}_{\text{reuse}}$	sets of conflict and reuse BS pairs
μ, h_{mi}	path loss and CH gain between BS m and UE i
P_{mi}^k, $\widehat{P}_m$	per sCH and total Tx power of BS m
W, N_0	sCH bandwidth and thermal noise spectral power
Γ_{mi}^k, R_{mi}^k	SINR and data rate of downlink mi on sCH k
ψ_i, $\widehat{R}_i$	requested and achieved downlink rates of UE i
$\widehat{C}_i$, $U_i(\boldsymbol{x}, \boldsymbol{y})$	incurred cost and individual utility of UE i
x_i^m, $\boldsymbol{x}$	user association decision variable and vector
y_{mi}^k, $\boldsymbol{y}$	resource allocation decision variable and vector
$U(\boldsymbol{x}, \boldsymbol{y})$, U_f	utility function (or) sum rate with pricing
$f = \{\boldsymbol{x}, \boldsymbol{y}\}$	configuration of the network
$p_f(U_f)$, p_f	probability of configuration in f
$q_{(f\to f')}$	transition probability from configuration f to f'

[1]Note that hereinafter we use service classes or QoS levels interchangeably.

B. Interference Mitigation via Dynamic Spectrum Partitioning

Due to the disparity in the transmit powers,[2] the macrocell base station (MBS) can cause significant interference to small cell base station (SBS) links. Inter-tier interference mitigation is the key challenge in HetNets since it involves mutual interference between an MBS and an SBS. The MBS link usually overwhelms the SBS link with its interference, decreasing the SBS link's SINR. Thus, an SBS link might need a large number of sub-channels which can be highly resource consuming. In [26]–[29], the authors proposed various methods for interference mitigation and spectrum sharing and reuse. The proposed approaches are diverse in terms of performance metrics and mechanisms. However, there is a basic common idea underlying these existing works which is to isolate (allocate orthogonal resources) high interfering links. Hence, we isolate the high-interference links by using orthogonal allocation [26]–[29].

We assume that the locations of BSs are fixed, and we propose a dynamic spectrum partitioning scheme based on the notions of a conflict graph and a reuse graph [30]. In practice, interference measurement and estimation for the downlink must be made at the UE side. The intuition is to isolate any two BSs with high-interference links from each other by spectrum partitioning [26]. Thus, depending on the interference levels between BS m and BS n, we define the sets of conflict and reuse BS for sub-channel k as: $\forall k \in \mathcal{S}$, $\forall m, n \in \mathcal{B}^k \subseteq \mathcal{B}$, $\forall i, j \in \mathcal{U}$,

$$\mathcal{B}_{\text{conflict}} = \left\{ (m,n) \mid \min\{\Gamma^k_{mi}, \Gamma^k_{nj}\} \leq \widetilde{\Gamma} \right\}, \tag{6}$$

$$\mathcal{B}_{\text{reuse}} = \left\{ (m,n) \mid \min\{\Gamma^k_{mi}, \Gamma^k_{nj}\} > \widetilde{\Gamma} \right\}, \tag{7}$$

where $\widetilde{\Gamma}$ is the SINR threshold and $\min\{\cdot\}$ is the commonly used minimum operator such that $\min\{a, b\} = a$, if $a < b$ and $\min\{a, b\} = b$, if $a > b$. $\mathcal{B}_{\text{conflict}}$ represents the set of BS pairs that conflict with each other due to their high-interference links. $\mathcal{B}_{\text{reuse}}$ represents the set of BS pairs that do not conflict with each other because of their low-interference links, and thus, they can reuse the same sub-channels. Note that $\min\{\cdot\}$ is used since the interfering links may not be symmetric due to disparity in transmit power.

Then, the interference constraints and the resource allocation constraint will be:

$$y^k_{mi} + y^k_{nj} \leq 1,\ \forall (m,n) \in \mathcal{B}_{\text{conflict}},\ \forall k \in \mathcal{S}, \tag{8}$$

$$y^k_{mi} + y^k_{nj} \leq 2,\ \forall (m,n) \in \mathcal{B}_{\text{reuse}},\ \forall k \in \mathcal{S}, \tag{9}$$

$$\textstyle\sum_{i \in \mathcal{U}} \sum_{k \in \mathcal{S}} x^m_i\, y^k_{mi} \leq |\mathcal{S}|,\ \forall m \in \mathcal{B}. \tag{10}$$

Here, the constraints in (8) and (9) are used to determine whether any pair of BSs (m, n) can reuse the same sub-channel k or not. If $(m, n) \in \mathcal{B}_{\text{reuse}}$, the sub-channel k can be reused. On the other hand, if $(m, n) \in \mathcal{B}_{\text{conflict}}$, the spectrum reuse will not be possible. In other words, we identify the highest interfering BSs to BS m (i.e. its neighboring BSs) and isolate them by spectrum partitioning. Furthermore, the constraint in (10) ensures that the number of sub-channels allocated to UEs by BS m does not exceed the total number of available sub-channels.

Initially, using pilot signals, each BS identifies from which BSs it receives high or low power interference. (6) identifies BSs which conflict with one another, and hence, these BSs cannot be allocated the same resources. (7) identifies BSs which do not conflict with one another, and hence, these BSs can be allocated the same resources. Depending on the UE traffic demand, the resource partition sizes are changing dynamically with respect to the control variable y^k_{mi}, which will be discussed further in the next section. Note that the spectrum partitioning resembles a graph multi-coloring problem [30]–[32] which jointly covers interference mitigation and resource allocation, where $|\mathcal{S}|$-colors are assigned to $|\mathcal{B}|$ vertices.

III. Problem Formulation

Our goal is to design a mechanism that can offload traffic from the MBS to SBSs under given QoS constraints. Such an offload must be done in a self-organized fashion. Next, we consider possible objective functions and formulate the traffic offloading as an optimization problem.

A. Objective Function

In our system, there are two types of entities with different perspectives and objectives, namely, UEs and BSs. On the one hand, each UE wants to achieve maximum data rate given its QoS requirement. On the other hand, the BSs want to minimize their operational cost while meeting the QoS requirement of UEs. Hence, we define the objective function as the sum rate minus the total operating cost.

Since a major portion of the operational cost is the electricity bill, we define the operation cost in the downlink as: $C_{mi} = \lambda_m \sum_{k \in \mathcal{S}} y^k_{mi} P^k_{mi}$, where P^k_{mi} is the per sub-channel transmit power of BS m and λ_m is the unit transmit power price of BS m, expressed in bits/s/Hz/W. Therefore, each UE incurs a cost of $\widehat{C}_i = \sum_{m \in \mathcal{B}} x^m_i\, C_{mi}$. We consider a fixed per sub-channel transmission power, P^k_{mi}. However, due to the coupling between the decision variables x^m_i and y^k_{mi} in (11), the operation cost varies proportionally with the number of allocated sub-channels to a downlink between BS m and UE i. Hence, the BSs have an economic incentive to reduce the number of sub-channels (resources) occupied which, in turn, will reduce the operating cost. The individual and network utilities are given by:

$$U_i(\boldsymbol{x}, \boldsymbol{y}) = \sum_{m \in \mathcal{B}} x^m_i \sum_{k \in \mathcal{S}} y^k_{mi} \left(R^k_{mi} - \lambda_m P^k_{mi} \right), \tag{11}$$

$$U(\boldsymbol{x}, \boldsymbol{y}) = \sum_{i \in \mathcal{U}} U_i(\boldsymbol{x}, \boldsymbol{y}). \tag{12}$$

Note that a UE can be served by either the MBS or an SBS. On the one hand, the MBS can offer a high SINR link requiring only a few sub-channels to meet UE's QoS requirement. On the other hand, the SBS provides a low SINR link requiring many sub-channels. Furthermore, low SINR SBS links can reuse the spectrum efficiently whereas

[2] In practice, the typical transmit power of an MBS is around 43 dBm, and that of SBSs is $20 \sim 30$ dBm lower than that of the MBS [26].

high SINR links that use the MBS can impede spectrum reuse. Thus, there exists a delicate balance between the three sub-problems: interference mitigation, user association and resource allocation. Moreover, the tradeoff between the sum rate $\widehat{R}_i$ and the operating cost $\widehat{C}_i$ reflects the balance among the three sub-problems. The utilities defined in (11) and (12) incentivize UEs to choose low power SBSs instead of the MBS by pricing for higher powers. We remark that the proposed pricing differs from cell biasing [33], [34] in that λ_m is directly coupled with the control variables, and thus, all the three sub-problems. In contrast, cell biasing considers a bias factor to only influence user association.

B. Optimization Problem

We define the Joint Optimization Problem (JOP) for offloading traffic from MBS to SBSs under QoS provisioning as follows:

$$\begin{aligned} \text{JOP : } & \underset{\boldsymbol{x},\boldsymbol{y}}{\text{maximize:}} && U(\boldsymbol{x},\boldsymbol{y}) \\ & \text{subject to:} && (4),(5),(8),(9),(10). \end{aligned} \tag{13}$$

The individual elements of $\boldsymbol{x}$ and $\boldsymbol{y}$ can have binary values ('0' or '1'), i.e., $x_i^m \in \{0,1\}$ and $y_{mi}^k \in \{0,1\}$. Due to the unique association constraint given in (4), the number of possible downlinks (UE associations) is reduced from $2^{|\mathcal{U}|\cdot|\mathcal{B}|}$ to $|\mathcal{B}|^{|\mathcal{U}|}$. However, the number of possible resource allocations for each downlink is combinatorial, i.e., $2^{|\mathcal{S}|}$. Hence, the solution space of JOP is $|\mathcal{B}|^{|\mathcal{U}|}\cdot 2^{|\mathcal{S}|}$, and no computationally efficient solution for JOP exists. Moreover, JOP belongs to a class of assignment problems which are proven to be combinatorial and NP-hard [35], [36].

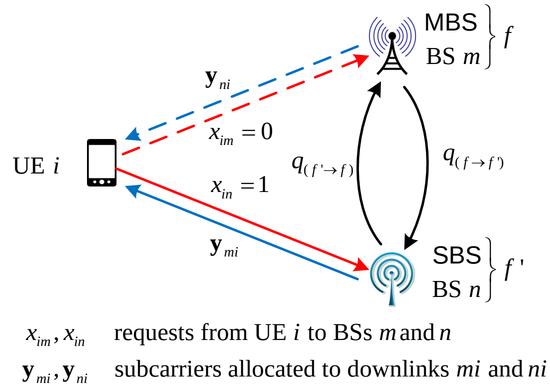
C. Offloading Procedure

Our objective is to design a UE initiated self-organizing algorithm for data offload in HetNets. Hence, we let the UEs decide on the association through control variable x_i^m where UE i randomly chooses BS m in its range according to a discrete uniform distribution. On the BS side, given the association, we minimize their operating cost through control variable y_{mi}^k. Hence, the computationally complex aspect of the user association is divided and distributed to all UEs. Thus, the problem becomes more tractable for the BSs.

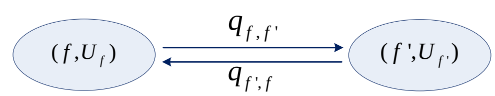
First, every UE i measures the received power from each BS m within its transmission range by using a pilot signal. From the measured received signal strength, UE i computes the minimum number of required sub-channels for its traffic demand so as to satisfy the QoS constraint in (5).

Then, UE i sends its request to BS m that the UE has chosen to associate with. The action of each UE i, $\boldsymbol{x}_i$, is $|\mathcal{B}|\times 1$ vector. As shown in Fig. 2a, when a given UE i sends its request to BS n, its corresponding decision variable is $x_i^n = 1$. The request contains the required data rate and the number of sub-channels needed to satisfy its QoS constraint defined in (5). Note that UE i can send only one request at any time according to the association constraint in (4).

Subsequently, BS n either accepts or rejects the traffic request from UE i depending on its available resources, i.e. (8)–(9). The acceptance of a request is indicated by the allocation of sub-channels in the form of a reply with sub-channel frequencies to UE i, i.e. $\boldsymbol{y}_{ni}$ (size $|\mathcal{S}|\times 1$) as shown in Fig. 2a. BS n will simply not reply in the case of a rejection, i.e. $\boldsymbol{y}_{ni}$ be a vector of zeros.



(a) Change in network configuration.



(b) Corresponding Markov chain.

Fig. 2: Transition rates from configuration f to f'.

IV. Traffic Offload via Markov Approximation

The optimization problem JOP is combinatorial and NP-hard as shown in [35], [36]. Thus, we can adopt the Markov approximation approach to solve JOP because of its ability to solve multiple sub-problems simultaneously without disjoint step-by-step solutions [21], [22]. Furthermore, additional control variables can be introduced in the optimization problem without altering the framework to handle the increased complexity [23]. Markov approximation was first proposed in [21]. A Markov approximation framework consists of two steps: log-sum-exp approximation and the development of problem-specific Markov chains that allow distributed implementation.

Let $f = \{\boldsymbol{x},\boldsymbol{y}\}$ be a network configuration with Ω being the set of all possible configurations, and $\mathcal{F} \subset \Omega$ be the set of all feasible configurations that satisfies the constraints in (4), (5), (8), (9), and (10). For ease of presentation, the utility in (12) is shortened to $U_f = U(\boldsymbol{x},\boldsymbol{y})$ and that in (13) is likewise shortened to $\max_{f\in\mathcal{F}} U_f$. Hence, the equivalent maximum weight independent set (MWIS) problem of JOP is [21]:

$$\max_{f\in\mathcal{F}} U_f \iff \begin{aligned} & \max_{\boldsymbol{p}\geq 0} \; \textstyle\sum_{f\in\mathcal{F}} p_f U_f \\ & \text{s.t.} \; \textstyle\sum_{f\in\mathcal{F}} p_f = 1 \end{aligned} \tag{14}$$

where p_f is the probability of choosing configuration f, i.e., its weight, and $\boldsymbol{p}$ denotes the vector of weights p_f. We can view p_f as the fraction of the time that configuration f is activated.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing

A. Log-sum-exp Approximation

The log-sum-exp function, $g_\beta(\boldsymbol{U})$, is convex and the closed function [21], [24, p. 93]. Thus, the conjugate of its conjugate $g_\beta^*(\boldsymbol{p})$ is itself, i.e. $g_\beta(\boldsymbol{U}) = g_\beta^{**}(\boldsymbol{U})$ [21], [24, p. 93]. Following the Markov approximation framework, the log-sum-exp approximation of $\max_{f\in\mathcal{F}} U_f$ yields

$$U_{\max} \approx \frac{1}{\beta}\log\left[\sum_{f\in\mathcal{F}}\exp(\beta U_f)\right] \triangleq g_\beta(\boldsymbol{U}), \tag{15}$$

where β is a positive constant, $\boldsymbol{U} \triangleq [U_f, f\in\mathcal{F}]$ and $U_{\max} = \max_{f\in\mathcal{F}} U_f$. Let $|\mathcal{F}|$ be the cardinality of set $\mathcal{F}$, then the approximation accuracy is given by [21], [24, p. 72]:

$$0 \le |U_{\max} - g_\beta(\boldsymbol{U})| \le \frac{1}{\beta}\log|\mathcal{F}|. \tag{16}$$

As $\beta \to \infty$, the approximation gap, $\frac{1}{\beta}\log|\mathcal{F}| \to 0$, and thus, the approximation becomes exact.

The log-sum-exp approximation in (15) is equivalent to solving the following optimization problem [21] [24, p. 93],

$$\begin{aligned}\max_{\boldsymbol{p}\ge 0}\quad & \underbrace{\textstyle\sum_{f\in\mathcal{F}} p_f U_f}_{\text{MWIS objective}} - \underbrace{\textstyle\frac{1}{\beta}\sum_{f\in\mathcal{F}} p_f \log p_f}_{\text{entropy term}} \\ \text{s.t.}\quad & \textstyle\sum_{f\in\mathcal{F}} p_f = 1.\end{aligned} \tag{17}$$

By finding the Karush-Kuhn-Tucker (KKT) conditions [24, p. 243] of the optimization problem given in (17), we obtain the optimal probability distribution, $\boldsymbol{p}^*$, which is given by

$$p_f^*(U_f) = \frac{\exp(\beta U_f)}{\sum_{f'\in\mathcal{F}}\exp(\beta U_{f'})}, \quad \forall f\in\mathcal{F}. \tag{18}$$

However, (18) requires *completeness*, i.e. complete information on $\mathcal{F}$ which can be difficult to find in a practical small cell network due to the large solution space. Thus, to obtain $\mathcal{F}$, we must solve the feasibility problem on Ω which is computationally exhaustive. Moreover, (18) is equivalent to $p_f^*(U_f) = \left(\sum_{f'\in\mathcal{F}}\exp[\beta(U_{f'} - U_f)]\right)^{-1}$, which considers the difference in utilities.

B. Markov Chain and Transition Rate

The next step is to design a problem specific Markov chain. Each state f represents a configuration with its corresponding stationary distribution $p_f^*(U_f)$ given by (18) and the set of states $\mathcal{F}$ contains all feasible configurations. As the probability distribution of the Markov chain converges, the configurations will be time-shared according to p_f^*. Hence, according to (18), the configurations with high utilities will have high probability, and thus, the network will operate in those configurations most of the time. It was proven in [21] that for any probability distribution of the product form $p_f^*(U_f)$ given in (18), there exists at least one time-reversible ergodic Markov chain whose stationary distribution is $p_f^*(U_f)$.

Let configurations $f, f' \in \mathcal{F}$ be the states of a time-reversible ergodic Markov chain with stationary distributions $p_f^*(U_f)$, $(f\in\mathcal{F})$ in (18). Let $q_{(f\to f')}$ and $q_{(f'\to f)}$ be the non-negative transition rates from $f\to f'$ and $f'\to f$, respectively. Then, the two following conditions are sufficient to allow a large degree of freedom in the algorithm design [21]:

- Any two states are reachable from each other,
- All $f, f'\in\mathcal{F}$ satisfy the balanced equation, (19).

$$\begin{aligned} p_f^*(U_f)\, q_{(f\to f')} &= p_{f'}^*(U_{f'})\, q_{(f'\to f)}, \\ \exp(\beta U_f)\, q_{(f\to f')} &= \exp(\beta U_{f'})\, q_{(f'\to f)}. \end{aligned} \tag{19}$$

The balance equation in (19) is significant because complete information on all possible configurations, $\mathcal{F}$, is no longer necessary. Moreover, as long as (19) is satisfied, any $q_{(f\to f')}$ and $q_{(f'\to f)}$ values can be used to design an algorithm. In essence, (19) shifts the problem of finding optimal $\boldsymbol{p}^*$ into designing a transition rate that will enable the Markov chain to converge to $\boldsymbol{p}^*$. Furthermore, the Markov chain is time-reversible, and hence, it will converge to $\boldsymbol{p}^*$ with probability 1. For our design, we consider the following condition:

$$q_{(f\to f')} + q_{(f'\to f)} = \exp(-\tau), \tag{20}$$

where τ is a positive constant. From (19) and (20), we have

$$q_{(f\to f')} = \exp(-\tau)\cdot\left(1+\exp[\beta\,(U_f - U_{f'})]\right)^{-1}, \tag{21}$$

$$q_{(f'\to f)} = \exp(-\tau)\cdot\left(1+\exp[\beta\,(U_{f'} - U_f)]\right)^{-1}, \tag{22}$$

which are logistic functions of utility differences.

Distributed Implementation: Note that (21)–(22) depend on U_f, which is the global utility given in (12). However, due to the distributed nature of the network, a UE can know only its own individual local utility given in (11) without additional signaling. For notational convenience, let $U_{f_i} = U_i(\boldsymbol{x}_i, \boldsymbol{y}_i)$, where $\boldsymbol{x}_i$ and $\boldsymbol{y}_i$ denote the individual user association and resource allocation actions of UE i, respectively. Then, we substitute the individual utilities in (21)–(22) to obtain

$$q_{(f_i\to f_i')} = \exp(-\tau)\cdot\left(1+\exp[\beta_i\,(U_{f_i} - U_{f_i'})]\right)^{-1}, \tag{23}$$

$$q_{(f_i'\to f_i)} = \exp(-\tau)\cdot\left(1+\exp[\beta_i\,(U_{f_i'} - U_{f_i})]\right)^{-1}. \tag{24}$$

Since we are using local utilities instead of global utilities for the transition probabilities, the Markov chain converges to a distribution $\widetilde{p}_f(U_f)$ instead of $p_f^*(U_f)$ given in (18). The authors in [22] have proven that the gap between $p_f^*(U_f)$ and $\widetilde{p}_f(U_f)$ is bounded. The total variation distance $d_{TV}(\boldsymbol{p}^*, \widetilde{\boldsymbol{p}})$ [37] between $p_f^*(U_f)$ and $\widetilde{p}_f(U_f)$ is given as:

$$0 \le d_{TV}(\boldsymbol{p}^*, \widetilde{\boldsymbol{p}}) \le 1 - \exp(-2\,\beta\,\delta_{\max}), \tag{25}$$

where $d_{TV}(\boldsymbol{p}^*, \widetilde{\boldsymbol{p}}) \triangleq \frac{1}{2}\sum_{f\in\mathcal{F}}|p_f^* - \widetilde{p}_f|$, $\delta_{\max} = \max_{f\in\mathcal{F}}\delta_f$, and $U_{\max} = \max_{f\in\mathcal{F}} U_f$. Moreover, the optimal gap between the utilities is bounded as follows:

$$0 \le \left|\boldsymbol{p}^*\boldsymbol{U}^T - \widetilde{\boldsymbol{p}}\boldsymbol{U}^T\right| \le 2\,U_{\max}(1-\exp(-2\,\beta\,\delta_{\max})). \tag{26}$$

The detailed analysis is presented in Appendix A.

C. MIDA: Markov Chain Directed Algorithm

The next challenge is to design an effective and economical algorithm using (23)–(24). We present the Markov chain Directed Algorithm (MIDA) in Alg. 1, which is computationally efficient and can be implemented distributively MIDA solves (17) in two steps, *exploration* and *consolidation*. During the *exploration*, a UE randomly chooses a configuration $f_i \in \mathcal{F}_i$ with probability $1/|\mathcal{F}_i|$, since it does not know how much utility it will receive in advance. In the subsequent time slot, the UE performs *consolidation* in which it compares the utilities it achieved in two previous time slots. The utilities are calculated using (11) and the transition probabilities are calculated using (23)–(24). If $U_{f_i'} > U_{f_i}$, then $q_{(f_i \to f_i')} > q_{(f_i' \to f_i)}$, and the configuration f_i' has a higher probability to be chosen at next time slot than f_i. In this manner, MIDA repeatedly solves (17) and the solution will eventually converge to (18) due to the property of the underlying Markov chain.

In actual implementation, MIDA is separated into three phases as shown in Alg. 1; initialization (Lines 1–4), user association (Lines 6–14) and resource allocation (Lines 15–21). In the initialization phase (Lines 1–4), $\mathcal{U}_P$ denotes a set of participating UEs, $\boldsymbol{X} := [\boldsymbol{x}_1, ..., \boldsymbol{x}_{|\mathcal{U}|}]$ (size $|\mathcal{B}| \times |\mathcal{U}|$) denotes the matrix of variable x_i^m, $\boldsymbol{Y} := [\boldsymbol{y}_1, ..., \boldsymbol{y}_{|\mathcal{B}|}]$ (size $|\mathcal{S}| \times |\mathcal{B}|$) denotes the matrix of variable y_{mi}^k, $\boldsymbol{U} := [U_1, ..., U_{|\mathcal{U}|}]^T$ (size $1 \times |\mathcal{U}|$) denotes the vector of utilities achieved by UEs, $\boldsymbol{\Upsilon} := [\Upsilon_1, ..., \Upsilon_{|\mathcal{U}|}]$ (size $N \times |\mathcal{U}|$) denotes the matrix for convergence analysis, and $\chi_i(t) \in \{0, 1\}$ denotes the binary variable indicating whether UE i explores in time slot t. BSs construct the conflict and reuse graphs in advance for interference mitigation (Line 4).

In the user association phase, if UE i did not explore in time slot t, it will *explore* in time slot $t+1$. Otherwise, it will *consolidate* knowledge learned in the previous time slots. In the *exploration* (Lines 6–7), UE i randomly chooses a BS to associate with in time slot $t+1$ and sends an association request as shown in Fig. 2. In the *consolidation* (Lines 8–13), UE i probabilistically compares the utilities achieved in time slot t and $t-1$ using (23)–(24). In the resource allocation phase (Lines 15–20), BS m processes the received requests from UEs in a FIFO manner. BS m allocates resources (sub-channels) to UE i if the constraints in (8), (9), and (10) are satisfied. In our model, all the BSs will guarantee that the constraints in (8)–(10) are not violated by storing and updating the binary resource allocation matrix $\boldsymbol{Y}$ (size $|\mathcal{S}| \times |\mathcal{B}|$) where resource vector $\boldsymbol{y}_m$ is the m-th column of $\boldsymbol{Y}$. Each BS m finds free resources in its own resource vector and resources vectors of its reuse BSs, i.e., $\boldsymbol{y}_m$ and $\boldsymbol{y}_n$, $\forall(m,n) \in \mathcal{B}_{\text{reuse}}$.

Convergence of MIDA*:* The Markov chain design leads to the concept of *convergence in probability*. To explain the concept, we give the following definitions.

Definition 1 (Normalized performance gap)**.** *Let $U(t)$ denote the utility achieved at time slot t and ε_0 denote the accuracy level. Then, we define the normalized performance gap between $U(t)$ and $U_{\max}$ as follows:*

$$\varepsilon(t) = \frac{|U_{\max} - U(t)|}{U_{\max}}. \tag{27}$$

Definition 2 (Convergence in Probability)**.** *$U(t)$ converges in probability to $U_{\max} = \max_{f \in \mathcal{F}} U_f$ as $t \to \infty$, if and only if $\lim_{t \to \infty} \Pr(\,\varepsilon(t) \geq \varepsilon_0\,) = 0$.*

There are two performance gaps associated with MIDA: (i) between $U_{\max}$ and $\boldsymbol{p}^* \boldsymbol{U}^T$ given in (16), and (ii) between $\boldsymbol{p}^* \boldsymbol{U}^T$ and $\widetilde{\boldsymbol{p}} \boldsymbol{U}^T$ given in (26). Both performance gaps are covered by (27). In [38], the authors showed that, by dynamically assigning β, we can arrive at $U_{\max}$. Hence, we use individual dynamic β_i for each UE and β_i can be assigned as follows:

- Linear assignment: $\beta_i(0) = 0$, $\beta_{\text{step}} > 0$, and $\beta_i(n+1) = \beta_i(n) + \beta_{\text{step}}$,
- Geometric assignment: $\beta_i(0) = 1$, $\beta_{\text{step}} > 1$, and $\beta_i(n+1) = \beta_i(n) \cdot \beta_{\text{step}}$.

We choose the geometric assignment for MIDA (Line 11) since it provides better mixing characteristics for the underlying Markov chain. Furthermore, the stopping rules for MIDA are designed as follows. The past configurations of each UE i are stored in Υ (Lines 12–13). If the configuration of each UE i remains unchanged for N time slots, the UE no longer participates in the *exploration* (Lines 22-23). Thus, MIDA stops when $\mathcal{U}_P$ becomes an empty set.

Note that, when only one UE in the network changes its configuration while all other UEs keep their configurations fixed as shown in Fig. 2, there is no difference between (21)–(22) and (23)–(24), i.e., if $\{\boldsymbol{x}, \boldsymbol{y}\} \to \{\boldsymbol{x}', \boldsymbol{y}'\} \equiv \{\boldsymbol{x}_i, \boldsymbol{y}_i\} \to \{\boldsymbol{x}_i', \boldsymbol{y}_i'\}$, then $d_{TV}(\boldsymbol{p}^*, \widetilde{\boldsymbol{p}}) = 0$. We refer to this special case as a *singular* case in which MIDA has only the approximation gap between $U_{\max}$ and $\boldsymbol{p}^* \boldsymbol{U}^T$ given in (16). Note that in MIDA, each *exploration* in time slot t is subsequently followed by the *consolidation* in time slot $t+1$, which will be further discussed in Section V-C.

V. Game Theoretic Perspective

In this section, we analyze JOP using a noncooperative game since it enables decision makers to take individual actions under strategic competition. In particular, the Markov approximation in Section IV does not take into account the competition between UEs whose interest is only in maximizing their own utilities. Further, the BSs are interested in minimizing the total operating cost. JOP presented in Section III can be modeled by a noncooperative, strategic game, defined as follows:

$$\mathcal{G} = (\{\mathcal{U} \times \mathcal{B}\}, \{\mathcal{X}_i, \mathcal{Y}_{mi}\}_{i \in \mathcal{U}, m \in \mathcal{B}}, \{V_{mi}\}_{i \in \mathcal{U}, m \in \mathcal{B}}), \tag{28}$$

where $\mathcal{X}_i = \{\boldsymbol{x}_i^{(1)}, \ldots, \boldsymbol{x}_i^{(|\mathcal{B}|)}\}$, $\mathcal{Y}_{mi} = \{\boldsymbol{y}_{mi}^{(1)}, \ldots, \boldsymbol{y}_{mi}^{(2^{|\mathcal{S}|})}\}$, and sizes of vectors $\boldsymbol{x}_i$ and $\boldsymbol{y}_j$ are $(|\mathcal{B}| \times 1)$ and $(|\mathcal{S}| \times 1)$, respectively. Note that $\boldsymbol{x}_i$ and $\boldsymbol{y}_{mi}$ are vectors of control variables x_i^m and y_{mi}^k, respectively. Furthermore,

- a player is a UE–BS pair $(i, m) \in \{\mathcal{U} \times \mathcal{B}\}$;
- the actions of the joint players are
  - UE i chooses BS m and sends request, $\{\mathcal{X}_i\}_{i \in \mathcal{U}}$;
  - BS m either accepts or rejects the request, $\{\mathcal{Y}_{mi}\}_{m \in \mathcal{B}}$;
- the joint payoff of UE i and BS m is, $\{V_{mi}\}_{i \in \mathcal{U}, j \in \mathcal{B}}$.

Let $l = (m, i)$ denote the downlink between UE i and BS m. We further define that $l(i) = 1$, if UE $i \in l$, and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing

Algorithm 1: MIDA

Result: $\boldsymbol{X}$, $\boldsymbol{Y}$, $\boldsymbol{U}$

1: **initialization:** $\mathcal{U}_P := \mathcal{U}$, $\forall i \in \mathcal{U}_P$
2: $\quad \boldsymbol{X} := \boldsymbol{0}$, $\boldsymbol{Y} := \boldsymbol{0}$, $\boldsymbol{\Upsilon} := \boldsymbol{0}$.
3: $\quad \tau := 0$, $\beta_i(0) := 1$, $U_i(\boldsymbol{x}, \boldsymbol{y}) := 0$, $\chi_i(t) := 0$.
4: $\quad$ Construct (6)–(7) from pilot signals.
5: **while** $t \leq T$, $\mathcal{U}_P \neq \emptyset$, *UE* $i \in \mathcal{U}_p$ **do**
6: $\quad$ **if** $\chi_i(t) = 0$ **then**
7: $\quad\quad \boldsymbol{x}_i(t+1)$ is randomly chosen.
8: $\quad$ **else**
9: $\quad\quad$ Calculate $\nu := q_{(f \to f')}$ using (23)
10: $\quad\quad \boldsymbol{x}_i(t+1) := \begin{cases} \boldsymbol{x}_i(t), & \text{with prob. } \nu. \\ \boldsymbol{x}_i(t-1), & \text{with prob. } 1-\nu. \end{cases}$
11: $\quad\quad \beta_i(n+1) := \beta_i(n) \cdot \beta_{\text{step}}$.
12: $\quad\quad \Upsilon_i(n-1) := \Upsilon_i(n)$, $\forall n \in \{2, ..., N\}$.
13: $\quad\quad \Upsilon_i(n) := \boldsymbol{x}_i(t+1)$.
14: $\quad \chi_i(t+1) := 1 - \chi_i(t)$.
15: $\quad$ Find $\boldsymbol{y}_{mi}$ in FIFO manner that satisfies (5),(8),(9).
16: $\quad$ **if** *BS* m *satisfies* (10) **then**
17: $\quad\quad$ Send $\boldsymbol{y}_{mi} \geq \boldsymbol{0}$ to UE i.
18: $\quad$ **else**
19: $\quad\quad$ Send $\boldsymbol{y}_{mi} := \boldsymbol{0}$ to UE i.
20: $\quad$ Calculate $U_i(\boldsymbol{x}, \boldsymbol{y})$ using (11).
21: $\quad$ Update $\boldsymbol{X}$, $\boldsymbol{Y}$, $\boldsymbol{U}$.
22: $\quad$ **if** Υ_i *remains the same for* N *times* **then**
23: $\quad\quad \mathcal{U}_P := \mathcal{U}_P \setminus \{i\}$.

$l(m) = 1$, if BS $m \in l$. Moreover, let $f_l = (\boldsymbol{x}_i, \boldsymbol{y}_{mi})$ be the joint configuration or action of UE i's request and BS m's reply. Let $\boldsymbol{f}_{-l}$ denote the actions of other UEs and their associated BSs. Thus, for each individual UE–BS downlink l, the joint payoff function is:

$$V_l(f_l, \boldsymbol{f}_{-l}) = U_i(\boldsymbol{x}, \boldsymbol{y}). \tag{29}$$

On the one hand, each UE i wants to maximize its data rate. On the other hand, each BS m wants to minimize its operating cost. Note that $f_l \in \mathcal{F}_l$, where $\mathcal{F}_l$ is the set of feasible configurations that downlink l can take, which is defined by the constraints in (4), (5), (8), (9), and (10).

A. Mixed-strategy and Solution Concept

Mixed-strategy is a solution concept in game theory where each pure strategy or action is taken probabilistically. For our case, the network configurations are the actions of the players which are randomly selected according to the assigned probability distribution. The mixed-strategy solution can be used to time-share the network configurations to achieve the optimal payoff over a long time. Furthermore, the mixed-strategy corresponds to the probabilistic MWIS solution in (18).

First, the joint action set of each UE–BS pair l is discrete with ($|\mathcal{B}| \times |\mathcal{S}|$) possible actions, i.e. $f_l \in \mathcal{F}_l$. The probability that action $\hat{f}_l \in \mathcal{F}_l$ is chosen can be given by

$$\pi_l(f_l, \boldsymbol{f}_{-l}) = \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{I}_{\{f_l = \hat{f}_l\}}, \tag{30}$$

where $\mathbb{I}_{\{\cdot\}}$ is an indicator function. For notational convenience, we use $\pi_l = \pi_l(f_l, \boldsymbol{f}_{-l})$, $\pi_l \in \Pi_l$. Then, the expected payoff of the mixed-strategy can be calculated as

$$\bar{V}_l(\pi_l, \boldsymbol{\pi}_{-l}) = \sum_{f_l \in \mathcal{F}_l} \pi_l(f_l, \boldsymbol{f}_{-l}) \cdot V_l(f_l, \boldsymbol{f}_{-l}). \tag{31}$$

The notion of a mixed-strategy Nash equilibrium (NE) is used to solve this game [39].

Definition 3 (Nash Equilibrium)**.** *A strategy profile* $\boldsymbol{\pi}^* \in \boldsymbol{\Pi}$ *is a Nash equilibrium (NE) if no unilateral deviation in strategy by any single player is profitable for that player, i.e.,*

$$\forall l \in \mathcal{U} \times \mathcal{B},\ \pi_l \in \Pi_l :\ \bar{V}_l(\pi_l^*, \boldsymbol{\pi}_{-l}^*) \geq \bar{V}_l(\pi_l, \boldsymbol{\pi}_{-l}^*).$$

For any noncooperative game, at least one mixed-strategy NE always exists [39]. Our goal is to find a Pareto-efficient NE and design a learning algorithm that will converge to the NE.

B. POLA: Payoff-based Log-linear Learning Algorithm

We propose a learning algorithm to find a mixed-strategy NE. There are many learning algorithms that can converge to the NE [40]. Among those, we choose to explore log-linear learning because it is similar to Markov approximation discussed in Section IV.

1) Log-linear learning: Initially, we assume *completeness*, which means that each UE–BS pair l has complete information on other UE–BS pairs, i.e., $\mathcal{F}_l, \forall l \in \mathcal{U} \times \mathcal{B}$. Furthermore, we assume *singularity*, i.e. at each time $t > 0$, only one UE–BS pair $l \in \mathcal{U} \times \mathcal{B}$ is randomly chosen and allowed to change its configuration whereas all other UE–BS pairs must repeat their current configuration, i.e. $\boldsymbol{f}_{-l}(t) = \boldsymbol{f}_{-l}(t-1)$. This reflects the similar assumption in Section IV-C. At time t, downlink l employs the strategy $\boldsymbol{\pi}_l \in \Pi_l$ where

$$\pi_l(f_l, \boldsymbol{f}_{-l}) = \frac{\exp[\beta_l\, V_l(f_l, \boldsymbol{f}_{-l})]}{\sum_{f_l' \in \mathcal{F}_l} \exp[\beta_l\, V_l(f_l', \boldsymbol{f}_{-l})]}, \tag{32}$$

for any configuration $f_l \in \mathcal{F}_l$. As $\beta \to \infty$, (32) assigns more weight to actions with higher payoffs [3], [25]. However, (32) requires complete information on $\mathcal{F}_l$ which we consider as a major drawback. To achieve our goal of a self-organizing distributed algorithm, we must relax *singularity* and *completeness* assumptions.

2) Payoff-based log-linear learning algorithm: POLA is an extension of log-linear learning that relaxes the two major assumptions, i.e. *completeness* and *singularity* [3]. POLA is presented in Alg. 2. POLA is structured similarly to MIDA and solves $\mathcal{G}$ in an iterative manner. We add an *independently and identically distributed* (i.i.d.) Bernoulli process with exploration rate (probability) ω. This simple process adds the learning phenomenon of *exploration* versus *exploitation* into POLA. In the *exploration* phase, each UE i chooses either of the two options for time slot $t+1$ (Line 7). With probability ω, UE i experiments by finding new random feasible downlink configuration $f_l(t) = f_{l'}(t)$, $l(i) = 1$, to execute (i.e., *exploration*). With probability $1-\omega$, UE i repeats its current configuration, $f_l(t) = f_l(t-1)$, $l(i) = 1$, (i.e., *exploitation*).

The *consolidation* phase follows in a subsequent time slot for every experimentation where UE i compares the current utility obtained with the previously achieved utility. UE i probabilistically chooses the configuration which achieves the maximum utility using (33). These steps are repeated until the underlying Markov chain converges to a stationary distribution, i.e. ϵ–Nash equilibrium. Due to the addition of Bernoulli process, POLA has a higher degree of randomness than that of MIDA which will further be discussed in Section V-C.

For notational convenience, let $\Theta_0 = V_l(f_l(t-1))$ and $\Theta_1 = V_{l'}(f_{l'}(t))$ where $l(i) = 1$ and $l'(i) = 1$. Then, we design the consolidation rate of UE i as:

$$\nu = \frac{\exp[\beta_l\,\Theta_0]}{\exp[\beta_l\,\Theta_1] + \exp[\beta_l\,\Theta_0]}, \tag{33}$$

$$= \left(1 + \exp[\beta_l\,(\Theta_1 - \Theta_0)]\right)^{-1}, \tag{34}$$

where $f_l(t-1)$, $l(i) = 1$, and $f_{l'}(t)$, $l'(i) = 1$, represent the configurations of UE i at time $(t-1)$ and t, respectively. Note that (33) is a log-linear function of utilities, and thus, named log-linear learning. However, as $\beta_l \to \infty$, (33) becomes undefined. Hence, as done in (23)–(24) for MIDA, we use the logistic function of utility difference (34) in POLA.

3) Convergence of POLA*:* We consider a generic noncooperative game in which POLA is guaranteed to converge to an ϵ–Nash equilibrium of game $\mathcal{G}$ as the underlying Markov chain converges to its stationary distribution. ϵ–Nash equilibrium is defined as follows:

Definition 4 (ϵ–Nash equilibrium)**.** *A strategy $\tilde{\boldsymbol{\pi}} \in \boldsymbol{\Pi}$ is an ϵ–Nash equilibrium for $\mathcal{G}$ if*

$$\forall l \in \mathcal{U} \times \mathcal{B},\ \pi_l \in \Pi_l:\ \bar{V}_l(\tilde{\pi}, \tilde{\boldsymbol{\pi}}_{-l}) \geq \bar{V}_l(\pi_l, \tilde{\boldsymbol{\pi}}_{-l}) - \epsilon.$$

At a given ϵ–NE, no UE can increase its own average utility by more than ϵ by unilaterally deviating from its current strategy. As $\epsilon \to 0$, the ϵ–NE can approach the NE. The bound for utility improvement $\forall l \in \mathcal{U} \times \mathcal{B}$ obtained by the unilateral deviation from a given ϵ–NE is

$$\forall \pi'_l \in \Pi_l:\ \bar{V}_l(\pi'_l, \tilde{\boldsymbol{\pi}}_{-l}) - \bar{V}_l(\tilde{\pi}, \tilde{\boldsymbol{\pi}}_{-l}) \leq \frac{1}{\beta_l} \log|\mathcal{F}_l|. \tag{35}$$

Hence, $\tilde{\boldsymbol{\pi}}$ is an ϵ–NE with $\epsilon = \max\limits_{l \in \mathcal{U}\times\mathcal{B}} \frac{1}{\beta_l} \log|\mathcal{F}_l|$. As $\beta_l \to \infty$, $\epsilon \to 0$, and hence, the ϵ–NE can be made sufficiently close to the NE by choosing the parameter β_l. This corresponds to the optimality gap of the Markov approximation given in (16).

Since the game $\mathcal{G}$ is finite, the existence of at least one ϵ–NE follows from [41, Theorem 1]. However, this result does not guarantee the uniqueness of the ϵ–NE, which strongly depends on the parameter β_l. For instance, when $\beta_l \to 0$, there exists a unique ϵ–NE since it corresponds to $\tilde{\pi}_l = 1/|\mathcal{F}_l|,\ \forall l \in \mathcal{U} \times \mathcal{B}$ by (32). Clearly, this ϵ–NE is unique and independent of the number of NEs that game $\mathcal{G}$ might have. On the other hand, when $\beta_l \to \infty$, $\epsilon \to 0$. Hence, the set of ϵ–NEs becomes identical to the set of NEs by the definition of ϵ–NE.

Algorithm 2: POLA

Result: $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{U}$
initialization: $\mathcal{U}_P := \mathcal{U},\ \forall i \in \mathcal{U}_P$
  $\boldsymbol{X} := \boldsymbol{0},\ \boldsymbol{Y} := \boldsymbol{0},\ \boldsymbol{\Upsilon} := \boldsymbol{0}$.
  $\tau := 0,\ \beta_i(0) := 1,\ U_i(\boldsymbol{x}, \boldsymbol{y}) := 0,\ \chi_i(t) := 0$.
  Construct (6)–(7) from pilot signals.
while $t \leq T,\ \mathcal{U}_P \neq \emptyset$, *UE* $i \in \mathcal{U}_p$ **do**
  if $\chi_i(t) = 0$ **then**
    $\boldsymbol{x}_i(t+1) := \begin{cases} \mathrm{rand}(\mathcal{X}_i) & \text{with prob. } \omega. \\ \boldsymbol{x}_i(t), & \text{with prob. } 1-\omega. \end{cases}$
  else
    Calculate ν using (34).
    $\boldsymbol{x}_i(t+1) := \begin{cases} \boldsymbol{x}_i(t), & \text{with prob. } \nu. \\ \boldsymbol{x}_i(t-1), & \text{with prob. } 1-\nu. \end{cases}$
    $\beta_i(n+1) := \beta_i(n) \cdot \beta_{\text{step}}$.
    $\Upsilon_i(n-1) := \Upsilon_i(n),\ \forall n \in \{2, ..., N\}$.
    $\Upsilon_i(n) := \boldsymbol{x}_i(t+1)$.
  $\chi_i(t+1) := 1 - \chi_i(t)$.
  Find $\boldsymbol{y}_{mi}$ in FIFO manner that satisfies (5),(8),(9).
  if *BS m satisfies* (10) **then**
    Send $\boldsymbol{y}_{mi} \geq \boldsymbol{0}$ to UE i.
  else
    Send $\boldsymbol{y}_{mi} := \boldsymbol{0}$ to UE i.
  Calculate $U_i(\boldsymbol{x}, \boldsymbol{y})$ using (11).
  Update $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{U}$.
  if Υ_i *remains the same for N times* **then**
    $\mathcal{U}_P := \mathcal{U}_P \setminus \{i\}$.

C. ROSE*: Randomized Self-organizing Algorithm*

1) Observations: We analyze MIDA and POLA (Section IV–V) and make the following observations.

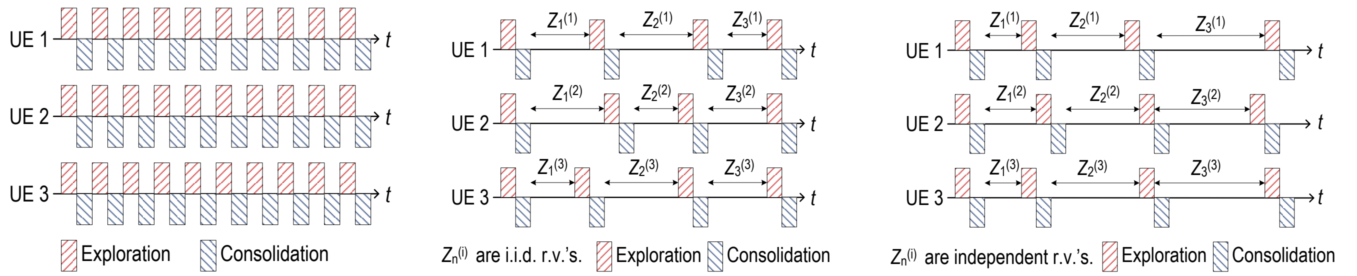
Assumptions and relaxations: Both MIDA and POLA have two key initial assumptions;

- *Completeness*: UEs and BSs have complete knowledge on the sets of feasible configurations.
- *Singularity*: at any given time t, only a single downlink configuration can be changed.

These assumptions yield the softmax functions given in (18) and (32). Furthermore, relaxation of these assumptions leads to the transition probabilities given in (23)–(24) and (34). MIDA and POLA have the same underlying Markov chain, and hence, convergence is assured with probability one. The performance bounds are given in (16), (26) and (35), respectively.

Algorithmic structure: Although MIDA and POLA appear to share some similar algorithmic structure as shown in Alg. 1 and Alg. 2, there exists a profound non-intuitive difference. Fig. 3 shows the renewal cycles [42] of *exploration–consolidation* phases of MIDA and POLA. As shown in Fig. 3a, MIDA has a fixed structure for *exploration–consolidation* which does not *exploit* knowledge learned in the *exploration*. At any time slot t, all UEs participate in the perfectly synchronized *exploration–consolidation* cycles. Thus, MIDA is a deterministic algorithm with no randomness in its algorithmic structure. Due to the synchronized cycle, all UEs are actively competing against each other

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing



(a) Renewal cycle of MIDA. (b) Renewal cycle of POLA. (c) Renewal cycle of ROSE.

Fig. 3: Renewal cycles of MIDA, POLA and ROSE for 3 UEs showing their randomness. *Exploitation* intervals, Z_n^i, are geometric r.v.'s with *exploration* rate, ω_i, as their parameter.

without *exploiting* any opportunities or knowledge learned.[3]

As shown in Fig. 3b, POLA *exploits* the knowledge learned during *exploration* with rate $1-\omega$. As described in Alg. 2, at any time slot t, all UEs will either explore with probability ω or exploit the learned knowledge with probability $1-\omega$. As shown in Fig. 3b, the introduction of ω removes the synchronization between UEs, and thus, some UEs explore while some other UEs repeat their configurations. From the UE population perspective, $\omega\,|\mathcal{U}|$ UEs are actively competing while the other $(1-\omega)\,|\mathcal{U}|$ UEs are *exploiting* and waiting for opportunities. Hence, POLA is not deadlocked into a constantly active competition among UEs as in MIDA. In other words, the introduction of randomness through ω improves the mixing characteristics of the underlying Markov chain, and thus, improves the performance. However, the rate of *exploration* versus *exploitation* is fixed by the exploration rate ω.

Convergence: In real-life implementation, $U_{\max}$ is not known in advance. Thus, the stopping times of MIDA and POLA are hard to control due to the underlying Markov chain. Furthermore, the performance bound $\varepsilon(t)$ is *submodular* with respect to running time t. In other words, the running time t increases linearly whereas the performance bound $\varepsilon(t)$ increases logarithmically, i.e., diminishing returns.

2) ROSE: Randomized Self-organizing Algorithm: From the observations made in Section V-C1, we refer to followings as the key determining factors for the learning algorithm's performance;

- the randomness of the algorithm, and
- the balance between *exploration* and *exploitation*.

Furthermore, these two factors are closely related where the exploration rate ω determines the balance between *exploration* and *exploitation*. As shown in Fig. 3b, the *exploitation* intervals of POLA are *independent and identically distributed* (i.i.d.) geometric random variables (r.v.'s) with ω as their parameter. Fixed ω leads to a static fixed balance point which does not react well in a dynamically changing environment. We will take humans as an example. As small children, we explore all day trying and learning new things. As we grow older, we explore less because we have more knowledge to exploit on. This shows that the learning rate of humans is dynamically changing (decreasing) over time.

Algorithm 3: ROSE

Result: $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{U}$
initialization: $\mathcal{U}_P := \mathcal{U},\ \forall i \in \mathcal{U}_P$
  $\boldsymbol{X} := \boldsymbol{0},\ \boldsymbol{Y} := \boldsymbol{0},\ \boldsymbol{\Upsilon} := \boldsymbol{0}$.
  $\tau := 0,\ \beta_i(0) := 1,\ U_i(\boldsymbol{x},\boldsymbol{y}) := 0,\ \chi_i(t) := 0$.
  Construct (6)–(7) from pilot signals.
while $t \le T,\ \mathcal{U}_P \neq \emptyset$, *UE* $i \in \mathcal{U}_p$ **do**
  if $\chi_i(t) = 0$ **then**
    $\boldsymbol{x}_i(t+1) := \begin{cases} \text{rand}(\mathcal{X}_i) & \text{with prob. } \omega_i. \\ \boldsymbol{x}_i(t), & \text{with prob. } 1-\omega_i. \end{cases}$
  else
    Calculate ν using (34)
    $\boldsymbol{x}_i(t+1) := \begin{cases} \boldsymbol{x}_i(t), & \text{with prob. } \nu. \\ \boldsymbol{x}_i(t-1), & \text{with prob. } 1-\nu. \end{cases}$
    $\beta_i(n+1) := \beta_i(n) \cdot \beta_{\text{step}}$.
    $\Upsilon_i(n-1) := \Upsilon_i(n)$.
    $\Upsilon_i(n) := \boldsymbol{x}_i(t+1)$.
    if $\Upsilon_i(n-1) := \Upsilon_i(n)$ **then**
      $\omega_i := \max\{0,\ \omega_i - \omega_{\text{step}}\}$.
  $\chi_i(t+1) := 1 - \chi_i(t)$.
  Find $\boldsymbol{y}_{mi}$ in FIFO manner that satisfies (5),(8),(9).
  if *BS* m *satisfies* (10) **then**
    Send $\boldsymbol{y}_{mi} \ge \boldsymbol{0}$ to UE i.
  else
    Send $\boldsymbol{y}_{mi} := \boldsymbol{0}$ to UE i.
  Calculate $U_i(\boldsymbol{x},\boldsymbol{y})$ using (11).
  Update $\boldsymbol{X}, \boldsymbol{Y}, \boldsymbol{U}$.
  if $\omega_i = 0$ **then**
    $\mathcal{U}_P := \mathcal{U}_P \setminus \{i\}$.

Thus, we adopt a dynamic exploration probability, ω_i, for each UE i in ROSE. As shown in Fig. 3c, the *identical* assumption over time is relaxed, where $0 \le \omega_{\min} \le \omega_i \le \omega_{\max} \le 1$. Hence, for each renewal process of UE i, the recurrence times $\{Z_1^{(i)}, Z_2^{(i)}, \ldots, Z_n^{(i)}\}$ are *independently distributed* geometric random variables with mean $\frac{1-\omega_i}{\omega_i}$ and variance $\frac{1-\omega_i}{\omega_i^2}$. There are several ways to choose ω_i to make it dynamic:

- *Random*: ω_i is uniformly chosen from $[\omega_{\min}, \omega_{\max}]$.
- *Non-increasing*: $\omega_i(0) = \omega_{\max},\ \forall i \in \mathcal{U}$, and repeat
  - $\omega_i(n+1) = \max\{\omega_{\min}, \omega_i(n) - \omega_{\text{step}}\}$

[3] Note that MIDA is a special case of POLA with the *exploration* rate $\omega = 1$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing

until $\omega_i(n) = \omega_{\min}$.

- *Periodic*: $\omega_i(0) = \omega_{\max}$, $\forall i \in \mathcal{U}$, and repeat
  1) $\omega_i(n+1) = \max\{\omega_{\min}, \omega_i(n) - \omega_{\text{step}}\}$ until $\omega_i(n) = \omega_{\min}$;
  2) $\omega_i(n+1) = \min\{\omega_{\max}, \omega_i(n) + \omega_{\text{step}}\}$ until $\omega_i(n) = \omega_{\max}$.

We choose the non-increasing ω_i for ROSE following the intuition from the human example, which also fits with our traffic offloading scenario (Lines 14–15). In practice, when a UE joins the network initially, the UE has no information, and thus, must explore for possible configurations. After some time has passed, the UE will have learned some knowledge and can exploit it. Thus, the UE can reduce its exploration probability. Furthermore, when we choose $\omega_{\max} = 1$ and $\omega_{\min} = 0$, we have a bounded expected stopping time for ROSE. The details of ROSE are presented in Alg. 3.

Now, we will analyze and calculate the expected stopping of ROSE. Let N_ω be the number of steps in decrements of ω_i. For each UE i, the expected stopping time can be calculated as:

$$\mathrm{E}[T_i] = 2 \cdot (N_\omega + 1) + \sum_{t=1}^{N_\omega} \frac{1 - \omega_i}{\omega_i}, \quad i \in \mathcal{U}, \tag{36}$$

where T_i follows a negative binomial distribution. Since the random variables $\{Z_1^{(i)}, Z_2^{(i)}, ..., Z_n^{(i)}\}$ are independent, the variance of T_i can be calculated as:

$$\eta_i^2 = \sum_{t=1}^{N_\omega} \frac{1 - \omega_i}{\omega_i^2}, \quad i \in \mathcal{U}. \tag{37}$$

Moreover, consider all the individual random processes of UEs with stopping times $\{T_1, T_2, ..., T_{|\mathcal{U}|}\}$ which are i.i.d. random variables. Since ROSE will stop only when the last UE stops the exploration–consolidation sequence, we apply the superposition of renewal processes [42, Chapter 6]. The expected stopping time of ROSE can be approximated as:

$$\mathrm{E}[T] \approx \mathrm{E}[T_i] + \frac{(|\mathcal{U}| - 1)(\mathrm{E}^2[T_i] - \eta_i^2)}{2 \cdot |\mathcal{U}| \cdot \mathrm{E}[T_i]}. \tag{38}$$

In summary, relaxing the identical assumption (i.e. introducing dynamic ω_i) improves the mixing characteristics of the underlying Markov chain. In other words, the underlying Markov chain undergoes more thorough mixing by dynamically balancing *exploration* and *exploitation* in a shorter amount of time.

VI. Simulation Results and Analysis

We perform extensive simulations in MATLAB to evaluate our proposed algorithms. For the benchmark, we use the optimal solution $U_{\max}$ which is computed using the built-in simulated annealing functions in MATLAB. The major simulation parameters are given in Table II.

A. Simulation Settings

First, for all our experiments, we assume the BSs to be deployed at fixed locations. Second, we randomly deploy UEs following a homogeneous PPP for different experiments. Third, we consider discrete user demands (i.e. requested data rate) whose probability mass function (PMF) is a binomial distribution. In this network, we consider a log-distance path loss model given by: $\mu = \mu_0 + 10\,\zeta \log_{10} \frac{d}{d_0} + X_g$, where μ is the total path loss in (dB), μ_0 is the path loss at reference distance d_0 for the BS, d is the length of transmission path, ζ is the path loss exponent, and X_g is the attenuation in dB caused by fading. Moreover, we assume that

- for indoors, $d \leq 20\,\mathrm{m}$, $\zeta = 3$, and X_g is a Gaussian random variable with zero mean and standard deviation σ reflecting attenuation caused by shadow fading, and
- for outdoors, $d > 20\,\mathrm{m}$, $\zeta = 4$, and X_g is a Rayleigh random variable for fast fading.

The reference path loss is calculated using two-ray ground reflection model as:

$$\mu_0 = 40 \log_{10}(d_0) - 10 \log_{10}(G h_t^2 h_r^2), \tag{39}$$

where G is the transmit antenna gain, h_t and h_r are the heights of the antenna of transmitter and receiver, respectively.

TABLE II: Default Simulation Parameters

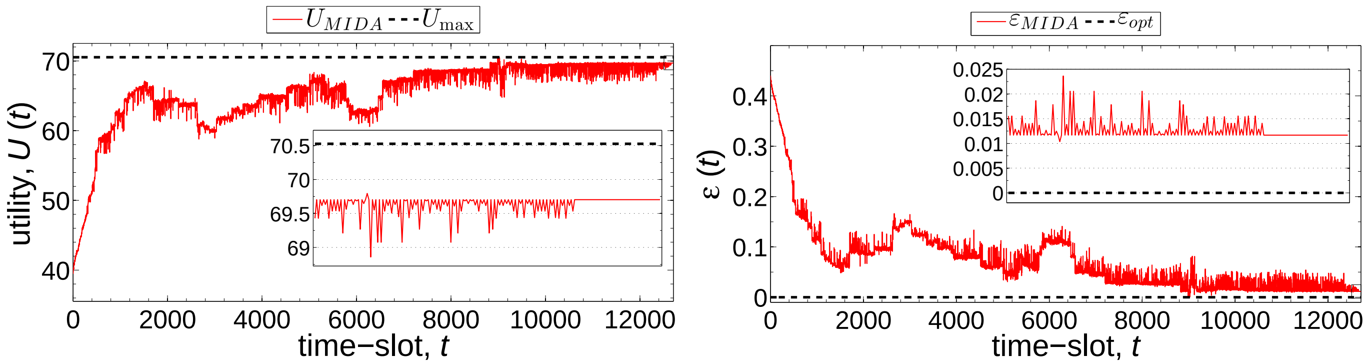
Quantity	Values
Area of region (A)	$200\,\mathrm{m} \times 200\,\mathrm{m}$
Static UE population ($\lvert\mathcal{U}\rvert$)	100
UE traffic demand (ψ_i)	$[0.1, 1]\,\mathrm{Mbps}$
# of BS ($\lvert\mathcal{B}\rvert = \lvert\mathcal{B}_\mathrm{m} \cup \mathcal{B}_\mathrm{p} \cup \mathcal{B}_\mathrm{f}\rvert$)	$20 = 1 + 2 + 17$
Total transmit power of BSs	$\{46, 36, 26\}\,\mathrm{dBm}$
Antenna gain of BSs (G)	$\{12, 9, 6\}\,\mathrm{dBi}$
Reference distance of BSs (d_0)	$\{1000, 100, 20\}\,\mathrm{m}$
Transmit antenna height of BSs (h_t)	$\{30, 10, 3\}\,\mathrm{m}$
# of sub-channels ($\lvert\mathcal{S}\rvert$)	12×100
Bandwidth of each sub-channel (W)	$15\,\mathrm{kHz}$
Thermal noise for $1\,\mathrm{Hz}$ at 20° C	$-174\,\mathrm{dBm}$
Unit price of transmit power (λ)	2×10^6

B. Static Traffic

We first perform experiments for static traffic with a fixed number of users to validate the theoretical background and check the performance of our proposed algorithms. First, we randomly generated $|\mathcal{U}| = 100$ UEs. We then examine various aspects to evaluate: 1) the log-sum-exp approximation gap, 2) the convergence of MIDA, POLA and ROSE, 3) the breakdown of individual BS-tier for ROSE, 4) the effect of λ on JOP, and 5) the effect of ω_{step} on ROSE.

1) Log-sum-exp approximation gap: In this experiment, we validate the log-sum-exp approximation and its performance gap given in (16). The exact solution of the log-sum-exp approximation given in (18) can be obtained from MIDA by allowing only one UE to change its configuration in a time slot while other UEs keep their configurations fixed. The results are shown in Fig. 4 which indicate the real time utility calculated by (11)–(12) and the normalized performance gap computed by (27). The results verify that as $\beta \to \infty$, $\varepsilon(t) \to 0$. However, MIDA requires a relatively long period of time to converge to $\boldsymbol{p}^*$ in (18), since only one

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing



(a) Utility.

(b) Normalized performance gap.

Fig. 4: Log-sum-exp approximation gap in (16) for $|\mathcal{U}| = 100$, $N = 50$ and $\lambda = 2 \times 10^6$.

UE is changing its configuration in a time slot. This corroborates our diminishing returns observation in Section V-C1.

2) Convergence: We perform experiments to test the convergence of the proposed algorithms: MIDA, POLA and ROSE. In the experiments, we randomly deploy $|\mathcal{U}| = 100$ UEs and set the unit power price, $\lambda = 2 \times 10^6$. The results are shown in Figs. 5–6. Figs. 5a–5c show the real-time and average utility values of MIDA, POLA and ROSE, respectively. The real-time utility values are calculated using (12), and the time averages of utility values are taken by means of a sliding window. Figs. 5a–5c show the time average of utility values slowly converging towards $U_{\max}$ while the real-time utility values are fluctuating as the network changes configurations. As shown in Fig. 5a, MIDA has the highest level of fluctuations. Fig. 5b shows that the utility of POLA is increasing steadily with low level fluctuations compared to MIDA. Due to the dynamic learning rate ω_i, ROSE displays characteristics of both MIDA and POLA. As shown in Fig. 5c, ROSE, initially, displays a high level of fluctuations, and then, transforms into a steadily increasing trend with minor fluctuations.

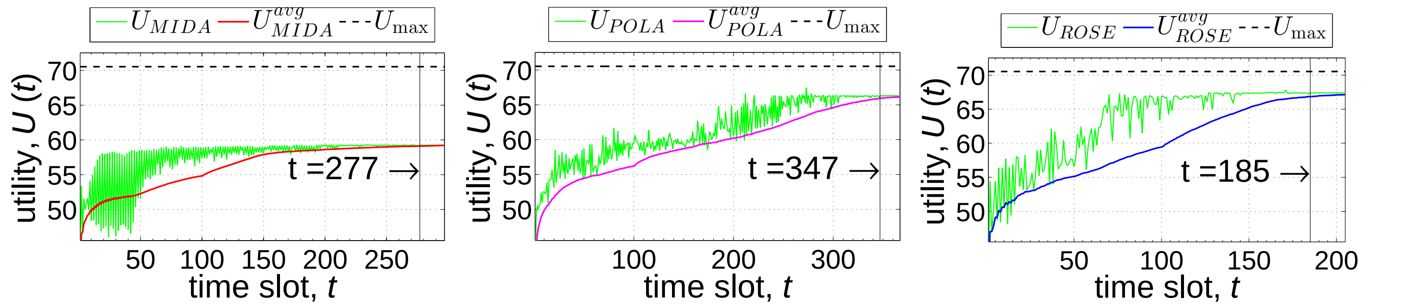
The corresponding normalized performance gaps of MIDA, POLA and ROSE are calculated using (27) and are shown in Figs. 5d–5f, respectively. Figs. 5d–5f show that MIDA, POLA and ROSE converge in probability as defined in Def. 2. The plots show that, after a relatively short time period, the $\varepsilon(t)$ values go below ε_0 fraction of $U_{\max}$ for ROSE, POLA and MIDA. It also supports the diminishing returns observation made in Section V-C1. In terms of convergence in probability, ROSE clearly outperforms both MIDA and POLA, due to its individual dynamic ω_i which continuously updates the balance between exploration and exploitation for each UE.

In Fig. 6, we separate the components of the utility into sum rate and total cost as defined in (11)–(12), i.e., $U(t) = R(t) - C(t)$. Since the sum rate has a higher priority than that of total cost, Figs. 6a and 6b are highly correlated. The priority in the utility is assigned via the unit power price λ. Fig. 6 shows that, in terms of utility and the achieved sum rate, $U_{\text{MIDA}} < U_{\text{POLA}} < U_{\text{ROSE}} < U_{\max}$ and $R_{\text{MIDA}} < R_{\text{POLA}} < R_{\text{ROSE}} < R_{\text{opt}}$. This supports our claim that the higher the degree of randomness, the better the performance of the learning algorithm.

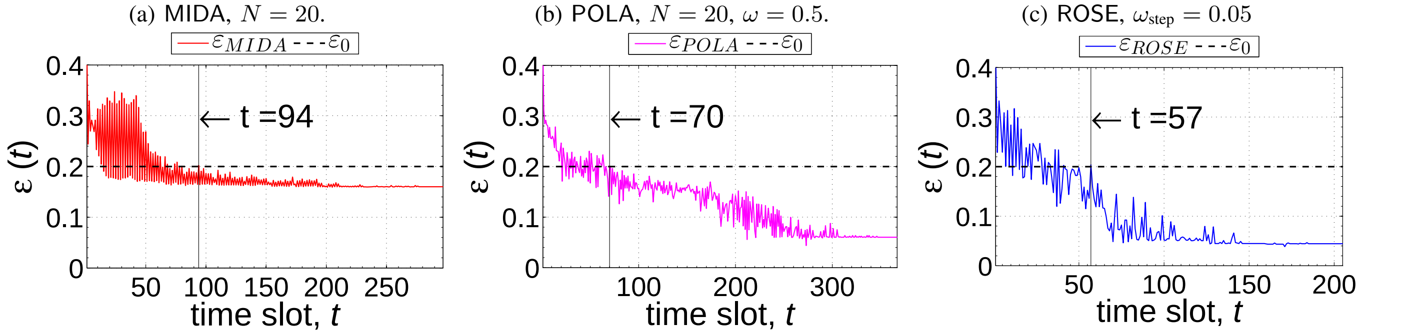
3) Breakdown of individual BS-tiers: In Fig. 7, to observe the performance of individual BS-tier, we breakdown the performance of ROSE. We first divide the network utility into sum rate and total cost, which are our two objectives calculated by (11)–(12). Fig. 7a shows the achieved sum rate of the network broken down for each BS-tier, and Fig. 7b shows the incurred total cost of the network for each BS-tier. Similarly, Fig. 7c shows the number of associated UEs for each BS-tier, and Fig. 7d shows the percentage of sub-channels occupied and reused. From Fig. 7, MBS initially ($t = 0$) has the highest number of users corresponding to the highest achievable sum rate. Moreover, the highest channel occupancy results in the highest incurred total cost. As we run Alg. 3, the UEs and BSs start exploration with probability ω_i and exploitation with $1-\omega_i$. As the time goes on, Fig. 7c shows that more and more UEs are associating with SBSs. Thus, the corresponding sum rate and total cost for the MBS are decreasing in Figs. 7a–7b. As more UEs are offloaded to the SBSs, the SBSs' data rate overtakes that of MBS. Similarly, the channel occupancy is also decreasing in Fig. 7d, since SBSs can reuse more sub-channels. As the time progresses more and more, UEs have explored all feasible configurations and stop exploration, i.e. $\omega_i \to 0$. Thus, the fluctuations decrease in magnitude, and ROSE converges.

4) Effect of unit price of transmit power λ: The unit price of transmit power (λ) is given in (11)–(12). λ can decide how much traffic is offloaded from MBS to SBSs. Since all three algorithms solve JOP, we only perform an experiment to study the effects of λ on ROSE for $|\mathcal{U}| = 100$. We run ROSE 100 times to take the averages of $R(t)$ and $C(t)$ at the stopping time. The average sum rates and total costs are broken down into respective BS-tiers as shown in Fig. 8. The MBS has the highest per sub-channel transmit power (P^k_{mi}), followed by pico-BSs in the second place, and femto-BSs have the lowest power. ROSE is maximizing the sum rate while minimizing the total cost at the same time as defined in (13). λ represents the weight that decides whether maximization or minimization is prioritized here. Hence, depending on the value of λ, ROSE offloads the traffic from

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing

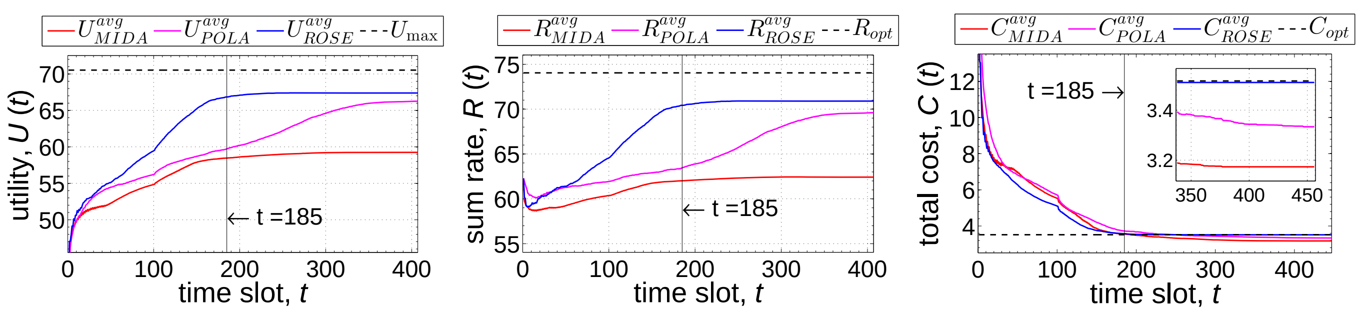


(a) MIDA, $N = 20$. (b) POLA, $N = 20$, $\omega = 0.5$. (c) ROSE, $\omega_{\text{step}} = 0.05$



(d) MIDA, $N = 20$. (e) POLA, $N = 20$, $\omega = 0.5$. (f) ROSE, $\omega_{\text{step}} = 0.05$

Fig. 5: Convergence of MIDA, POLA and ROSE, $|\mathcal{U}| = 100$, $\lambda = 2 \times 10^6$..



(a) Sum rate versus time. (b) Sum rate versus time. (c) Total cost versus time.

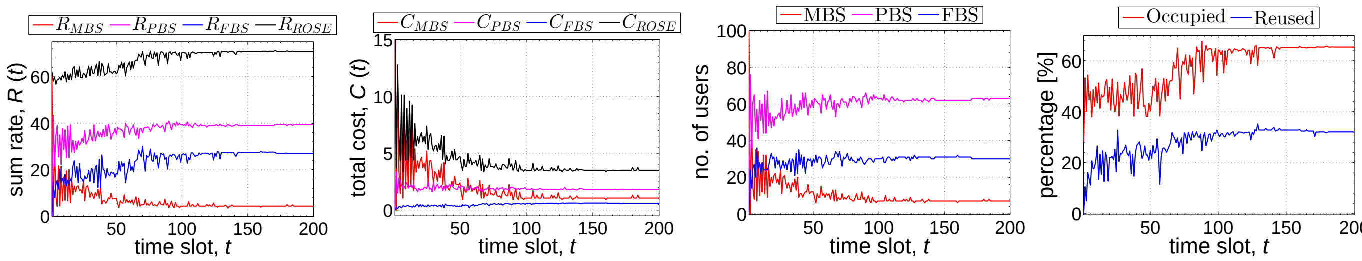
Fig. 6: Comparison between MIDA, POLA and ROSE, $|\mathcal{U}| = 100$, $\lambda = 2 \times 10^6$.

high power MBS to low power pico-BSs and femto-BSs. When $\lambda = 0$, the total incurred cost is not considered in JOP, and MBS tier has the highest sum rate. As we increase λ value, we can see a dramatic decrease in the sum rate of MBS-tier while the sum rates of PBS-tier and FBS-tier are increasing gradually In Fig. 8b, we can see the costs of PBS-tier and FBS-tier climbing steadily as we increase λ. The cost of MBS tier initially climbs with increasing λ, but then, it shows a downward trend as it fluctuates.

5) Impact of ω_{step} on ROSE*:* In this experiment, we vary ω_{step} to test the stopping time of ROSE. Then, we run 100 simulations for each point to find the average number of time slots that ROSE takes to stop and the normalized performance gap $\varepsilon(\omega_{\text{step}})$ at the stopping time. The theoretical results are obtained using (38). The results are shown in Fig. 9a where the analytical results and simulation results well agree, and the stopping times follow a negative binomial distribution. As shown in Fig. 9b, $\varepsilon(\omega_{\text{step}})$ at the stopping time of ROSE increases with the increasing step size, ω_{step}. This indicates that the longer ROSE takes to stop, the better the mixing characteristics of the underlying Markov chain and the better the performance. Thus, a tradeoff exists between the stopping time and the performance of ROSE. We can sacrifice performance to stop the algorithm quickly since there is diminishing returns on performance improvement with respect to the running time of ROSE. Note that all of the values of $\varepsilon(\omega_{\text{step}})$ fall well below the ε_0 of $U_{\max}$ for every value of ω_{step}. Thus, one can see that even when using a large step size, the proposed approach will not yield a performance that is worse than a fraction ε_0 of $U_{\max}$.

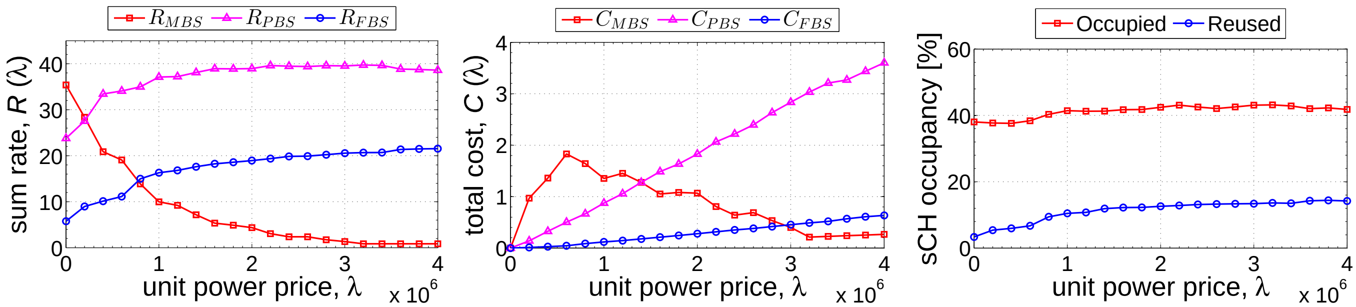
C. Dynamic Traffic

In this section, we randomly generate a network trace in which the arrival of UE requests (demand) follows a Poisson process, and the duration of the request is exponentially distributed. The inter-arrival times between UE demands and the duration of those demands are exponentially distributed



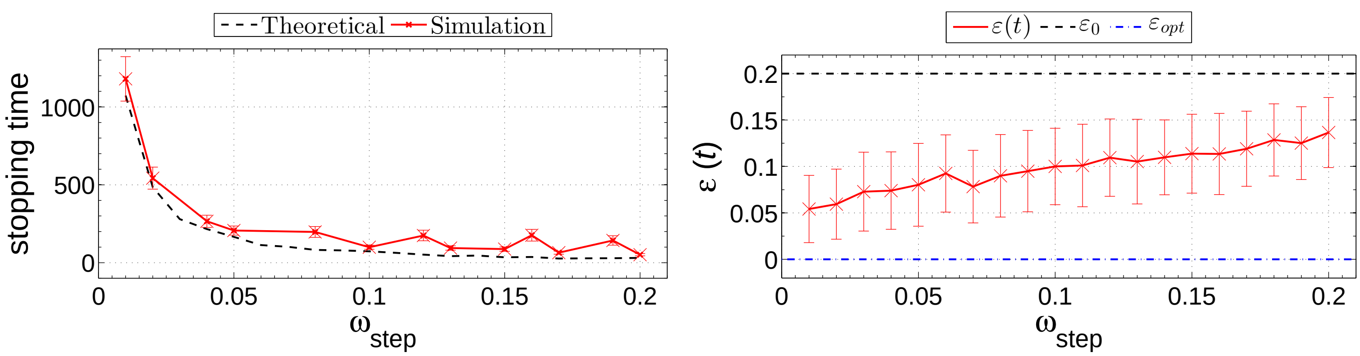
(a) Sum rate vs time slot. (b) Total cost vs time slot (c) No. of UEs vs time slot. (d) sCHs vs time slot.

Fig. 7: Breakdown of individual BS-tier on convergence of ROSE, $|\mathcal{U}| = 100$, $\omega_{\text{step}} = 0.05$.



(a) Sum rate versus λ. (b) Total cost versus λ. (c) Total cost versus λ.

Fig. 8: Effect of λ on ROSE, $|\mathcal{U}| = 100$.



(a) Impact on stopping time of ROSE. (b) Impact on performance of ROSE.

Fig. 9: Impact of ω_{step} on ROSE, $|\mathcal{U}| = 100$ and $\lambda = 2 \times 10^6$.

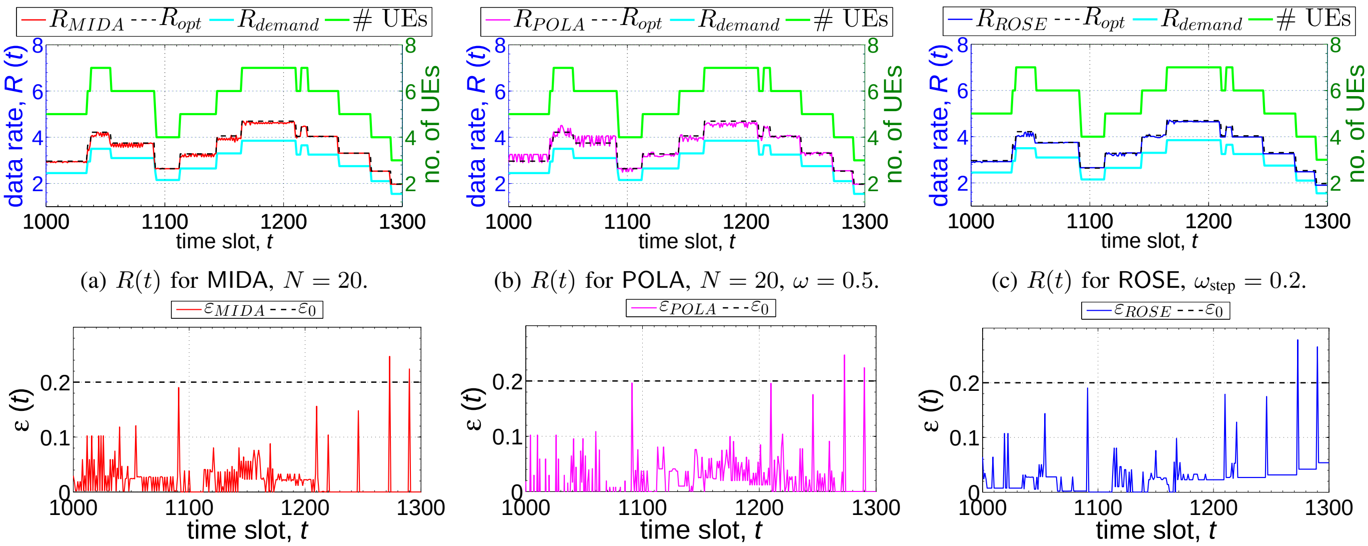
with rates ρ_A and ρ_D, respectively. We assume that each time slot is an LTE frame (10 ms). Hence, the probability of more than one UE demand arriving during a time slot is very low, and can be assumed to be zero. The design of (23)–(24) enables ROSE to be implemented in real-time with a few modifications to the stopping rules. At every demand arrival, ROSE runs until it converges to a local optimal configuration for that UE demand. Once it converges to the local optimal configuration, it will fix the configuration for the UE until its request duration is finished. When another demand request arrives, ROSE will repeat the process.

We then run the simulation for 20,000 time slots (equivalent to 200 seconds) to evaluate the performance of MIDA, POLA and ROSE. We plot the results in Fig. 10 where Figs. 10a–10c show the real-time trace of the sum rates achieved, and their corresponding normalized performance gap, $\varepsilon(t)$, values are shown in Figs. 10d–10f. Interestingly, we can see the sharp spikes of $\varepsilon(t)$ values in Figs. 10d–10f corresponding to departure of a UE in Figs. 10a–10c. However, we do not see similar spikes for UE arrivals. This shows that MIDA, POLA and ROSE configure each UE as it arrives and do not reconfigure the network when a UE departs. Hence, MIDA, POLA and ROSE can achieve at most the local optimal configuration for each UE. Fig. 11 shows the cumulative distribution function (CDF) of $\varepsilon(t)$ values for MIDA, POLA and ROSE. Fig. 11 shows that, in the presence of dynamic traffic, $\Pr(\varepsilon > \varepsilon_0)$ is within an acceptable margin. In particular, for $\varepsilon_0 = 0.2$, the confidence level is 95 percent, i.e., $\Pr(\varepsilon > \varepsilon_0) < 0.05$.

VII. Conclusions

In this paper, we have analyzed the traffic offload problem from macrocell base stations to small cell base stations

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing



(a) $R(t)$ for MIDA, $N = 20$.

(b) $R(t)$ for POLA, $N = 20$, $\omega = 0.5$.

(c) $R(t)$ for ROSE, $\omega_{\text{step}} = 0.2$.

(d) $\varepsilon(t)$ for MIDA, $N = 20$.

(e) $\varepsilon(t)$ for POLA, $N = 20$, $\omega = 0.5$.

(f) $\varepsilon(t)$ for ROSE, $\omega_{\text{step}} = 0.2$.

Fig. 10: Performance of MIDA, POLA and ROSE for dynamic traffic, $\lambda = 2 \times 10^6$.

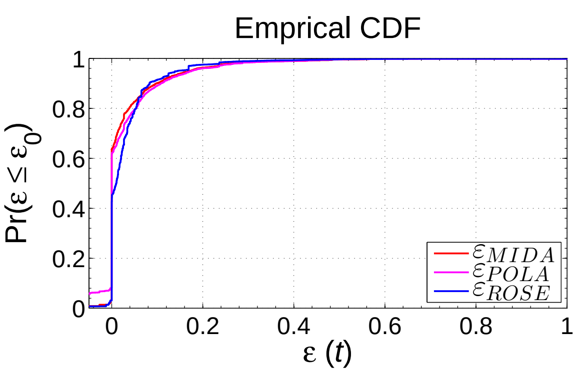


Fig. 11: Empirical CDF of $\varepsilon(t)$.

using the Markov approximation and game-theoretic approaches. We have formulated three joint sub-problems for user association, resource allocation and interference mitigation as the maximization of sum rate with pricing. We have designed a problem specific Markov chain and have introduced appropriate transition probabilities that ensure convergence, in probability, to a close-to-optimal solution. After relaxing the assumptions made in the Markov approximation framework, we have designed a Markov chain guided algorithm (MIDA) using which the network can self-organize to offload traffic from MBS to SBSs. The designed MIDA has been shown to converge to a bounded close-to-optimal solution. Furthermore, we have formulated the problem as a noncooperative game and have designed a payoff-based log-linear learning algorithm (POLA). The designed POLA has been shown to converge to an ϵ–Nash equilibrium. After analyzing the designs of MIDA and POLA, we have discovered that the randomness can improve the mixing characteristics of the underlying Markov chain. We have then proposed a highly randomized self-organizing algorithm (ROSE) which can converge to a pure-strategy mixed strategy. Simulation results verify that MIDA and POLA converge in probability, ROSE converges in real-time, and the traffic is offloaded from MBS to SBSs. Simulation results also prove that more randomized algorithms perform better than deterministic algorithms. Since ROSE has the highest degree of randomness, it has the best performance which concurs with our simulation results. Furthermore, our proposed approaches have been also shown to further decrease the total operating costs of the network.

References

[1] B. Molleryd, J. Markendahl, J. Werding, and O. Makitalo, "Decoupling of revenues and traffic - is there a revenue gap for mobile broadband?" in *9th CTTE*, Ghent, Belgium, June 2010.

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

[3] J. R. Marden and J. S. Shamma, "Revisiting log-linear learning: Asynchrony, completeness and payoff-based implementation," *Games and Economic Behavior*, vol. 75, no. 2, pp. 788 – 808, 2012.

[4] D. Niyato and E. Hossain, "Dynamics of network selection in heterogeneous wireless networks: An evolutionary game approach," *IEEE Trans. Veh. Technol.*, vol. 58, no. 4, pp. 2008–2017, May 2009.

[5] V. N. Ha and L. B. Le, "Fair resource allocation for OFDMA femtocell networks with macrocell protection," *IEEE Trans. Veh. Technol.*, vol. 63, no. 3, pp. 1388–1401, March 2014.

[6] A. Elsherif, W.-P. Chen, A. Ito, and Z. Ding, "Resource allocation and inter-cell interference management for dual-access small cells," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 6, pp. 1082–1096, June 2015.

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

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

[9] T. Wang and L. Vandendorpe, "Iterative resource allocation for maximizing weighted sum min-rate in downlink cellular OFDMA systems," *IEEE Trans. Signal Process.*, vol. 59, no. 1, pp. 223–234, January 2011.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TMC.2016.2613864, IEEE Transactions on Mobile Computing

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

[11] L. B. Le, D. Niyato, E. Hossain, D. I. Kim, and D. T. Hoang, "Qos-aware and energy-efficient resource management in ofdma femtocells," *IEEE Trans. Wireless Commun.*, vol. 12, no. 1, pp. 180–194, January 2013.

[12] F. Pantisano, M. Bennis, W. Saad, M. Debbah, and M. Latva-Aho, "Interference alignment for cooperative femtocell networks: A game-theoretic approach," *IEEE Trans. Mobile Comput.*, vol. 12, no. 11, pp. 2233–2246, November 2013.

[13] S. Bayat, R. Louie, Z. Han, B. Vucetic, and Y. Li, "Distributed user association and femtocell allocation in heterogeneous wireless networks," *IEEE Trans. Commun.*, vol. 62, no. 8, pp. 3027–3043, August 2014.

[14] Z. Zhang, L. Song, Z. Han, and W. Saad, "Coalitional games with overlapping coalitions for interference management in small cell networks," *IEEE Trans. Wireless Commun.*, vol. 13, no. 5, pp. 2659–2669, May 2014.

[15] B. Ma, M. H. Cheung, V. Wong, and J. Huang, "Hybrid overlay/underlay cognitive femtocell networks: A game theoretic approach," *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 3259–3270, June 2015.

[16] E. Biton, A. Cohen, G. Reina, and O. Gurewitz, "Distributed inter-cell interference mitigation via joint scheduling and power control under noise rise constraints," *IEEE Trans. Wireless Commun.*, vol. 13, no. 6, pp. 3464–3477, June 2014.

[17] J. B. Rao and A. O. Fapojuwo, "An analytical framework for evaluating spectrum/energy efficiency of heterogeneous cellular networks," *IEEE Trans. Veh. Technol.*, vol. 65, no. 5, pp. 3568–3584, May 2016.

[18] D. C. Chen, T. Q. S. Quek, and M. Kountouris, "Backhauling in heterogeneous cellular networks: Modeling and tradeoffs," *IEEE Trans. Wireless Commun.*, vol. 14, no. 6, pp. 3194–3206, June 2015.

[19] S. Samarakoon, M. Bennis, W. Saad, and M. Latva-aho, "Dynamic clustering and on/off strategies for wireless small cell networks," *IEEE Trans. Wireless Commun.*, vol. 15, no. 3, pp. 2164–2178, March 2016.

[20] 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, April 2015.

[21] M. Chen, S. C. Liew, Z. Shao, and C. Kai, "Markov approximation for combinatorial network optimization," *IEEE Trans. Inf. Theory*, vol. 59, no. 10, pp. 6301–6327, October 2013.

[22] S. Zhang, Z. Shao, M. Chen, and L. Jiang, "Optimal distributed P2P streaming under node degree bounds," *IEEE/ACM Trans. Netw.*, vol. 22, no. 3, pp. 717–730, June 2014.

[23] J. Jiang, T. Lan, S. Ha, M. Chen, and M. Chiang, "Joint VM placement and routing for data center traffic engineering," in *IEEE INFOCOM*, Orlando, FL, March 2012, pp. 2876–2880.

[24] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY: Cambridge University Press, 2004.

[25] L. E. Blume, "The statistical mechanics of best-response strategy revisions," *Games and Economic Behavior*, vol. 11, no. 2, pp. 111–145, November 1995.

[26] K. Son, S. Lee, Y. Yi, and S. Chong, "REFIM: A practical interference management in heterogeneous wireless access networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 6, pp. 1260–1272, June 2011.

[27] R. Giuliano, C. Monti, and P. Loreti, "WiMAX fractional frequency reuse for rural environments," *IEEE Wireless Commun.*, vol. 15, no. 3, pp. 60–65, June 2008.

[28] K. Son, S. Chong, and G. D. Veciana, "Dynamic association for load balancing and interference avoidance in multi-cell networks," *IEEE Trans. Wireless Commun.*, vol. 8, no. 7, pp. 3566–3576, July 2009.

[29] K. Son, Y. Yi, and S. Chong, "Utility-optimal multi-pattern reuse in multi-cell networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 1, pp. 142–153, January 2011.

[30] W. Ur Rehman, T. Salam, and X. Tao, "Vertex multi-coloring scheduling algorithm for concurrent transmission in 60-ghz networks," in *IEEE GLOBECOM*, Austin, TX, December 2014, pp. 4751–4757.

[31] E. L. Narayanan, *Channel assignment and graph multicoloring*. John Wiley & Sons, 2002.

[32] Y.-J. Chang, Z. Tao, J. Zhang, and C.-C. Kuo, "A graph-based approach to multi-cell ofdma downlink resource allocation," in *IEEE GLOBECOM*, New Orleans, LA, November 2008.

[33] H.-S. Jo, Y. J. Sang, P. Xia, and J. Andrews, "Heterogeneous cellular networks with flexible cell association: A comprehensive downlink sinr analysis," *IEEE Trans. Wireless Commun.*, vol. 11, no. 10, pp. 3484–3495, October 2012.

[34] Y. Wang, S. Chen, H. Ji, and H. Zhang, "Load-aware dynamic biasing cell association in small cell networks," in *IEEE International Conference on Communications (ICC)*, Sydney, Australia, June 2014, pp. 2684–2689.

[35] D. W. Pentico, "Assignment problems: A golden anniversary survey," *European Journal of Operational Research*, vol. 176, no. 2, pp. 774 – 793, 2007.

[36] H. Boostanimehr and V. Bhargava, "Unified and distributed QoS-driven cell association algorithms in heterogeneous networks," *IEEE Trans. Wireless Commun.*, vol. 14, no. 3, pp. 1650–1662, March 2015.

[37] P. Diaconis and D. Stroock, "Geometric bounds for eigenvalues of markov chains," *The Annals of Applied Probability*, vol. 1, no. 1, pp. 36–61, February 1991.

[38] S. Kirkpatrick, "Optimization by simulated annealing: Quantitative studies," *Journal of statistical physics*, vol. 34, no. 5-6, pp. 975–986, 1984.

[39] J. F. Nash, Jr., "Equilibrium points in n-person games." *Proceedings of the National Academy of Sciences of the United States of America*, vol. 36, no. 1, pp. 48 – 49., January 1950.

[40] L. Rose, S. Lasaulce, S. Perlaza, and M. Debbah, "Learning equilibria with partial information in decentralized wireless networks," *IEEE Commun. Mag.*, vol. 49, no. 8, pp. 136–142, August 2011.

[41] R. Mckelvey and T. Palfrey, "Quantal response equilibria for extensive form games," *Experimental Economics*, vol. 1, no. 1, pp. 9–41, 1998.

[42] D. R. Cox, *Renewal theory*. Methuen, 1967.

[43] Y. P. D. A. Levin and E. L. Wilmer, *Markov Chains and Mixing Times*. American Mathematical Society, 2009.