QoS Provisioning for Multiple Femtocells via Game Theory

V. Udaya Sankar and Vinod Sharma
Department of Electrical Communication Engineering,
Indian Institute of Science, Bangalore 560012, India
Email: {uday,vinod}@ece.iisc.ernet.in

Abstract—We consider a system with multiple Femtocells operating in a Macrocell. The transmissions in one Femtocell interfere with its neighboring Femtocells. Each Femtocell has multiple users each requiring a minimum transmission rate. There is also peak transmit power constraint in each channel to control interference to the BS and the users in the Macrocell. We formulate the problem of channel allocation and power control at the Femtocells as a noncooperative Game. We develop efficient decentralized algorithms to obtain a Nash point and satisfy the QoS of each user.

Keywords- Femtocell, cellular communication, power control, QoS, Game theory, Decentralized algorithms.

I. INTRODUCTION

The quality of wireless channels indoors suffers due to attenuations from the walls causing significant degradation of voice quality which is 60% of the cellular traffic generated indoors. Thus, to improve the quality, the Base Station (BS) will have to significantly increase the transmit power which may not be allowed. Hence, femtocells [3] are being considered as an option.

Femtocell access points (FAP) are small, inexpensive, low powered base stations designed for indoor deployment. When mobile Stations (MSs) are indoor and encounter outage from the (Macro) Base Station outside these can be handed over to the FAP in their building. A FAP is connected to the wireline infrastructure. Thus, it can receive data from a MS in its building and transfer to the backhaul network through the wireline connection. This improves the quality of service (QoS) to the indoor applications and also offloads a significant fraction of cellular traffic to wireline network.

Although, deployment of FAPs in a macrocell (MC) environment improves the performance of the users inside the FC, it causes interference to the MSs in the MC and adjacent FCs. Therefore, to reap the benefits of a FC one needs careful interference management ([18], [20]). An upper limit must also be imposed on the transmit power of the MSs within a FC. Reducing the transit power in a FC will also help reduce the carbon footprint of the cellular systems ([14]). We address this problem in this paper.

In the following we provide related literature survey. A good overview on the topic is provided in [3], [6]. Distributed game theoretic resource allocation is studied in [1], [2] and [16]. A distributed signal to interference noise ratio (SINR) adaptation at FCs is proposed in [4]. In [25], we proposed subchannel allocation and power control algorithms for uplink and downlink users within a FC.

Joint subchannel allocation and power control in a multichannel environment is addressed in [7] and [30]. In [7] the problem of providing minimum rate to each FC is considered via a distributed Auction. In [30] QoS for delay sensitive users (to provide minimum rate guarantee) in each FC is provided. A distributed convex optimization problem is formulated via dual decomposition to maximize the throughput of delay tolerant users. In [19] game theory is used to provide a minimum average rate for each FC via power control.

In this paper we consider the problem of providing satisfactory QoS to the uplink and downlink MSs in a multi FC dense environment via non-cooperative game theory. We consider subchannel and power allocation jointly. Our model includes the subchannel power constraints and considers the QoS to individual users within each FC. We develop low complexity distributed algorithms to compute Nash Equilibria (NE) which are energy efficient and provide QoS to all users in each FC. When it is not possible to satisfy the QoS of all users in each FC, we obtain efficient distributed NEs which are fair. Our problem has some similarity to that in [30], but as against [30], we use game theory and consider inter-FC interference. Also in [30] the case when the minimum rate guarantees are not satisfied, is not considered.

This paper is organized as follows. Section II describes the system model and formulates the problem. Section III, IV provides a game theoretic framework and also provides efficient algorithms. Section V shows the efficacy of the algorithms via a few examples. Section VI concludes the paper.

II. SYSTEM MODEL AND PROBLEM FORMULATION

We consider a two tier cellular system in which within a MC there may be many FCs. The cellular system has multiple subchannels, perhaps using OFDMA (e.g., LTE or WiMAX) and the subchannels are shared by the FCs and the outdoor users in the MC. The transmission between FAP and MSs may be in the uplink or downlink or in both directions. The allocation of channels and powers to different users will be done by the FAP using the same algorithms (although the peak power constraint in the two directions can be different for the same channel). Thus, for simplicity we will consider down link only.
The MSs using a FC are also the MSs for the Macro BS. Thus, these MSs can get the information from the Macro BS about which subchannels are being used by the users in the MC (e.g., from the DL/UL map in Wimax, PDCCH/PUCH in LTE sent by the MBS in each frame). These channels may not be used by the FC at that time. The indoor MSs can also sense the SINR in each subchannel and can further be directed by the MBS on the maximum power they can use in transmitting in different available subchannels. This information can be sent to the FBS by the MSs within its domain or by the MBS directly (see e.g., [31]). FBS uses this information to decide on subchannel allocation and power control within its FC to provide QoS to its users while using minimum power within the limits prescribed by the MBSs. Minimizing power reduces interference to other FCs and to MSs outside. We assume a control channel via which the different FCs can coordinate with each other and self configure.

Motivated by the above setup, in the following we formulate our problem. Let there be $K$ FCs deployed in a neighborhood sharing $N$ channels, FC $k$ has $M_k$ users. We consider allocation of channels to different users in a slot although our algorithms can be used for multiple slots also. Let $P^k_i$ be the maximum power that can be used in channel $i$ in FC $k$. Let $\bar{R}^k_i$ be the rate requirement of user $j$ in FC $k$. Let $C^k_{i,j}$ be the channel gain of channel $i$ for user $j$ in FC $k$. This includes the effect of fading and shadowing. We assume that it will stay same during the slot. Since FCs cater to indoor users, this is a reasonable assumption. We make the same assumption for other channel gains we consider in the paper. We denote by $G^k_{i,j}$ the channel gain for user $j$ in channel $i$ in FC $k$ from FC $l$. Also, $I^k_{i,j}$ denotes the interference (caused by outdoor users and MBS) experienced by user $j$ in channel $i$ in FC $k$. Let $\sigma^2$ denotes the receiver noise at a receiver, assumed same for simplicity, for all receivers. The FAP $k$ is supposed to have all this information. Such assumptions are also made in [17]. Based on this information each FAP $k$ has to decide the subchannel allocation to different users and the power $P^k_i$ to be used in subchannel $i$ such that the rate $\bar{R}^k_i$ is received by each user $j$ if possible. Also we desire an energy efficient solution so that the FC uses minimum power so as to reduce interference to users outside the FC and also to conserve energy. The decisions of one FAP will affect the decisions of others due to the interference caused to MSs in their FCs.

In the following we formulate this problem mathematically. Superscript $k$ will refer to FC $k$. Let

$$A^k_{i,j} = \begin{cases} 1 & \text{if subchannel } i \text{ is assigned to user } j, \\ 0 & \text{otherwise.} \end{cases}$$

Also let $C^k_{i,j} = \log_2 \left( 1 + \frac{P^k_i G^k_{i,j}}{\Gamma \sigma^2 + I^k_{i,j} + \sum_{j', j 
eq j'} G^k_{i,j'} P^k_{i'}} \right)$, where $\Gamma$ is the SNR gap included for practical rates achievable depending on the modulation and coding scheme(5)).

We consider the following problem: Find $P^k_i, i = 1, 2, ..., N$ and $A^k_{i,j}, i = 1, 2, ..., N, j = 1, 2, ..., M_k, k = 1, 2, ..., K$ to

$$\min \sum_{i=1}^{N} P^k_i \quad (1)$$

such that

$$\sum_{i=1}^{N} C^k_{i,j} A^k_{i,j} \geq \bar{R}^k_j, \quad \forall j = 1, 2, ..., M_k, \quad (2)$$

$$P^k_i \leq \bar{R}^k_i, \quad \forall i = 1, 2, ..., N, \quad (3)$$

$$\sum_{j=1}^{M_k} A^k_{i,j} \leq 1, \quad \forall i = 1, 2, ..., N. \quad (4)$$

Equation (2) specifies QoS requirements and (3) specifies power constraints. The constraint (4) ensures that any subchannel is allocated to only one user within a FC. If one central controller has all the global information of the system then this problem could be solved by it. This is a mixed constrained integer, nonlinear programming problem and is NP hard. However, presently we are assuming that each FAP $k$ knows only its own $\bar{R}^k_i, G^k_{i,j}, G^k_{i,j}, I^k_{i,j}$ and not of other FCs. Also each FAP does scheduling of channels and power control of its own users only. We provide a decentralized solution for this problem using non-cooperative game theory assuming that each FC behaves selfishly to satisfy the QoS of its own users. The algorithms provided have low computational complexity also.

### III. Game Theoretic Solution

#### A. Game formulation and solution

We formulate the game for our system as $\mathcal{G} = (\mathcal{I}, \mathcal{X}, (\Phi_k(x))_{x \in \mathcal{X}})$, where $\mathcal{I} = \{1, 2, ..., K\}$, the set of FCs is the set of players, $\mathcal{X}$ is the overall strategy space and $\Phi_k$ is the utility of player $k$. Let $P^k_i = \{P^k_i, i = 1, 2, ..., N\}$. We define strategy set $\mathcal{X} = \{P^k_i \in \mathcal{X}^k, A^k_{i,j} \in \{0, 1\}, k = 1, 2, ..., K\}$: $\sum_{i=1}^{N} C^k_{i,j} A^k_{i,j} \geq \bar{R}^k_j, j = 1, 2, ..., M_k, P^k_i \leq \bar{R}^k_i, \sum_{j=1}^{M_k} A^k_{i,j} \leq 1, i = 1, 2, ..., N$. We are interested in finding a decentralized energy efficient Nash Equilibrium ([26]) which provides QoS to each user in the system (if at all possible). If it is not possible we provide a fair NE (to be defined later).

Define the utility function for FC $k$ by

$$\Phi_k(x) = - \sum_{i=1}^{N} P^k_i, \forall x \in \mathcal{X}. \quad (5)$$

Maximizing $\Phi_k(x)$ will minimize the total average power used by FC $k$. It is easy to verify that the game $\mathcal{G}$ defined above is an exact Potential game ([22]) with Potential function

$$f(x) = - \sum_{k=1}^{K} \sum_{i=1}^{N} P^k_i. \quad (6)$$

We observe that $\mathcal{X}$ is a compact set. Also the potential function $f$ is continuous. Therefore, it has a global maximizer and hence has a NE ([19], [26]). In the following we provide distributed algorithms to compute NE.
Let

$$\mathcal{D}_k(x_{-k}) = \{x_k \in \mathcal{D}_k(x_{-k}) : x_k = \arg \max_{x_k} \Phi_k(x_k, x_{-k}) \}$$

(7)

where \(\mathcal{D}_k(x_{-k})\) is the set of strategies of \(k\) which are possible when the strategies of all other users are \(x_{-k}\). The set \(\mathcal{D}_k(x_{-k})\) provides strategies for player \(k\) which maximize its utility for a fixed strategy \(x_{-k}\) by all other users. For a given \(x_{-k}\), best response provides an element of \(\mathcal{D}_k(x_{-k})\).

Let for an \(\epsilon > 0\) and \(x_{-k}\),

$$\mathcal{D}_k(x_{-k}) = \{x_k \in \mathcal{D}_k(x_{-k}) : \Phi_k(x_k, x_{-k}) \geq \Phi_k(x, x_{-k}) + \epsilon, \quad \forall x_k \}$$

(8)

Then \(\epsilon\)-Better Response provides a point in \(\mathcal{D}_k(x_{-k})\).

In the following we develop distributed best response and \(\epsilon\)-better response algorithms for our game. The best response algorithm has high computational complexity while the \(\epsilon\)-better response is much less complex. At each FC \(k\) we can use these algorithms iteratively to reach a NE.

From [21] the limit points of the best response algorithm are NE. Furthermore, now since the potential function is bounded and in the \(\epsilon\)-better response algorithm, the potential function increases by at least \(\epsilon\), it converges in a finite number of steps to an \(\epsilon\)-Nash Point (23)).

The Best Response algorithm obtains, at iteration \(n + 1\), at each FC \(k\), \(k = 1, 2, ..., K\),

$$x_k^{n+1} \in \mathcal{D}_k(x_1^{n+1}, x_2^{n+1}, ..., x_{k-1}^{n+1}, x_{k+1}^{n}, ..., x_K^{n}),$$

(9)

and then passes on the best response to FC \(k + 1\). It can find \(x_k^{n+1}\) via solving the optimization problem (1)-(4) to obtain \(A_{i,j}^{k,n}\) and \(P_{k,i}^{n}\). This can be done by using mixed integer programming. However, this algorithm has a very high complexity because of a large number of variables.

For computationally simple algorithms, we consider random-better response (or random \(\epsilon\)-better response) [21] which converges for the above potential game [23]. These algorithms have lower complexity per iteration. But for a continuous strategy space one may need a large number of iterations. Thus, we develop a novel variation of these algorithms which converge much faster.

We take a random sample over \(A_{i,j}^k\). Once a random sample of \(A_{i,j}^k\) is taken, we pick \(P_{k,i}^n\) that optimizes (1)-(4). The KKT solution (28) for this is

$$P_{i}^k = \left(\frac{\lambda_k A_{i,j}^k}{\lambda_{k,j} A_{i,j}^k + \mu_{k,i} - \frac{1}{d_{i,j}^k}}\right)^+, \quad i = 1, 2, ..., N,$$

(10)

where \(d_{i,j}^k = \frac{G_{i,j}^k}{\lambda_{k,j} + \mu_{k,i}}, x^+ = \max\{x, 0\}\) and \(\lambda_k, \mu_{k,i}, i = 1, 2, ..., N, j = 1, 2, ..., M_k, k = 1, 2, ..., K\) are Lagrange multipliers such that (2) and (3) are satisfied with equality. This way we eliminate a large number of dominated strategies. As further improvement, even for \(A_{i,j}^k\) we do not sample from the whole sample space, but only a neighborhood of \(A_{i,j}^k\) (by changing only one row of \(A^k\)) which has a higher probability of providing an improved strategy. Next, instead of random better response, we opt for \(\epsilon\)-better response which converges in a finite number of iterations.

Furthermore, to get \(\epsilon\)-improvement we use a (partial) random algorithm: We do the random search in an iteration for \(N_1\) steps; if we still do not get it we keep the previous strategy itself. This provides the \(\epsilon\)-improvement with a low computational complexity with a high probability. If in \(N_1\) random samplings we do not get \(\epsilon\) improvement then either the algorithm has converged to an \(\epsilon\)-Nash point or it failed to find \(\epsilon\)-improvement even though we have not reached the \(\epsilon\)-Nash point. By increasing \(N_1\) we can reduce the probability of failure to an arbitrarily small value. This overall algorithm is summarized in Algorithms 1 and 2 below. This algorithm converges much faster than the original best response algorithm mentioned above.

For these algorithms the FCs do not really need to exchange their policies; each FC can measure its interference plus noise after implementing its policy and update the new policy.

Algorithm 1 Randomized Algorithm (RA)

Requirement: For each time instant generate Random Subchannel Allocation Matrix (RSAM) from Algorithm 2 and compute power allocation for this matrix using eq (10).

1. \(n = 0, x_k(0) = (P_{k,i}^0, A_{i,j}^0), \Phi_k = \Phi_k(x_k(0))\)
2. \(\forall n > 0\)
3. if (Rate requirements of all users satisfied simultaneously)
4. then
5. \(\Phi_n \geq \Phi_{n-1} + \epsilon\)
6. Pick \(x_k(n)\)
7. else
8. Pick randomly \(A^k\) via Algorithm 2 and test for above condition for a specified number of times (\(\cong N_1\)) till satisfies else take \(x_k(n) = x_k(n-1)\)
9. end if
10. else
11. Stop and go for fair allocation algorithm in section IV.
12. end if
13. (3) Do until convergence

Algorithm 2 Generation of RSAM

Requirement: Randomly pick \((i,j)^{th}\) element of subchannel allocation matrix \(A_{i,j}^{n-1}\)

if \((A_{i,j}^{n-1}(i,j) = 0)\) then
1. \(A_{i,j}^{n-1}(i,j) = 1\)
2. make another random element of \(i^{th}\) row which is 1, as 0 (if there is no other element of \(i^{th}\) row 1 then no change is required).
else
3. \(A_{i,j}^{n-1}(i,j) = 0\)
4. make another random element of \(i^{th}\) row as 1.
5. end if

Remark 1. The fact that \(G\) is a potential game, implies that the NE we obtain by our algorithms provided above when each
user is trying to maximize (improve) its utility, should provide an NE which performs well from the global perspective ([15], [23]).

IV. FAIR ALLOCATION

If the QoS of all the users in all the FCs cannot be satisfied then we obtain “fair” NE from the algorithms developed in this section. Of course, we need to first know if QoS of all the users can be satisfied in each FC or not. We can check this fairly quickly by using maximum possible power in each channel in each FC and getting good channel allocations for each FC separately via Algorithm 2.

Now we can allocate channels to each FC independently of others. For each FC, we randomly allocate channels via Algorithm 2 till QoS of all users are satisfied. If it does not happen for some time for a FC, then we will use the algorithms for the present section. This will generally provide a good test. If all QoS can be satisfied then we get the NE from Section III. If not, we proceed as follows. If by mistake we decided by above test that the QoS of all the users in all the FCs cannot be met, our Algorithm 3 in this section will provide the right NE.

If the QoS of all the users in the system cannot be satisfied simultaneously, then we try to satisfy the largest fraction of QoS of all the users in each FC. In particular, for each FC \( k \) we obtain power and subchannel allocation \( P^k \) and \( A^k \) that

\[
\max \quad \alpha_k
\]

such that

\[
\frac{1}{P^k_{ij}} \sum_{i=1}^{N} C^k_{i,j} A^k_{i,j} \geq \alpha_k, \quad \forall j = 1, 2, \ldots, M_k, \quad (11)
\]

and

\[
\sum_{j=1}^{M_k} A^k_{i,j} \leq 1, \quad \forall i = 1, 2, \ldots, N, \quad (12)
\]

where \( A^k_{i,j} \in \{0, 1\} \) and

\[
C^k_{i,j} = \log_2 \left( 1 + \frac{P^k_{i,j} G^k_{i,j}}{\sigma^2 + P^k_{i,j} \sum_{j \neq i} G^k_{i,j} P^k_{i,j}} \right).
\]

We use \( \alpha_k \) as the utility function for FC \( k \). This max-min fairness criterion within a FC has been used before ([10], [27]). We define strategy space for FC \( k \) as \( \mathcal{X}_k = \{ (P^k \in R^N_k, A^k_{i,j} \in \{0, 1\} : 0 \leq P^k_i \leq P^k_i \sum_{j=1}^{M_k} A^k_{i,j}, \leq 1, \quad i = 1, 2, \ldots, N \} \).

It is observed that \( \mathcal{X}_k \) is compact. Also the utility function \( \Phi_k(x_k, x_{-k}) \) for FC \( k \) is continuous. Thus the game has a (mixed) NE[8]. However since it is not a potential game, there is no known algorithm to compute a NE in this general setup. In the following we provide an algorithm to compute (approximately) the NE.

This algorithm actually computes an approximate Correlated Equilibrium (CE) [13], defined below, which also exists and can indeed provide a better equilibrium point.

Definition 1. A probability distribution \( p = (p(x))_{x \in \mathcal{X}} \) is a CE of game \( \mathcal{G} \) if \( \forall k \in \mathcal{I}, x_k \in \mathcal{X}_k, \ x_{-k} \in \mathcal{X}_{-k} \) and \( \forall x'_k \in \mathcal{X}_k \)

\[
\sum_{x_{-k} \in \mathcal{X}_{-k}} p(x_k, x_{-k}) \{ \Phi_k(x_k, x_{-k}) - \Phi_k(x'_k, x_{-k}) \} \geq 0. \quad (14)
\]

To get the CE point, we first discretize the strategy space (by discretizing the power). For the corresponding finite game we compute a CE via the regret matching algorithm ([11], [12]) provided below. We can show that as we reduce the discretization step of the strategy space, the CE obtained for the finite game converges to the CE of the original game ([24], [29]).

A stationary solution of a regret algorithm exhibits no regrets. In this algorithm the probability of a playing action is proportional to the “regret” for not having played this action. We use the following notation to describe this algorithm. The average utility of player \( k \) at time \( n \) is

\[
D^k_n(x_k, x_k^*) = \frac{1}{n} \sum_{\tau \leq n} \{ \Phi_k(x_k^*, x_k^\tau) - \Phi_k(x_k, x_k^\tau) \}. \quad (15)
\]

Average regret of player \( k \) for not playing action \( x_k^* \) is :

\[
R^k_n(x_k, x_k^*) = \max(D^k_n(x_k, x_k^*), 0). \quad (16)
\]

Algorithm-3 [11] describes the procedure for calculating CE for our game \( \mathcal{G} \).

Algorithm 3. Regret-Matching Learning Algorithm

- Initialize arbitrarily probability for taking action of player, \( p^1_k(x_k), \forall k \in \mathcal{I} \) for \( n = 1, 2, 3, \ldots \)
- 1. Find \( D^k_n(x_k, x_k^*) \) as in (15).
- 2. Find average regret \( R^k_n(x_k, x_k^*) \) as in (16).
- 3. Let \( x_k \in \mathcal{X}_k \) be the strategy chosen by user \( k \) at time \( n \), i.e. \( x_k^* = x_k \). Then probability distribution \( p^k_{n+1} \) for the action for next period is defined as,

\[
p^k_{n+1}(x_k') = \frac{1}{\mu} R^k_n(x_k, x_k'), \forall x_k \neq x_k^*,
\]

\[
p^k_{n+1}(x_k^*) = 1 - \sum_{x_k \neq x_k^*} p^k_{n+1}(x_k').
\]

where \( \mu \) is a sufficiently large constant.

Let the relative frequency of actions \( x \) played by all the players in first \( n \) periods be \( Z_n(x) = \frac{1}{n} \sum_{i=1}^{n} 1(x_i = x) \). The convergence of Algorithm 3 to a CE is guaranteed from [12].

V. EXAMPLES

In this section we first consider a system with 2 FCs each with 4 users and 10 subchannels deployed in a MC. We consider Time Division Duplex (TDD) scenario where all parameters, such as rate requirements, interference matrix and subchannel gain matrices do not change significantly within a given TDD duration. We use the following parameters.

Subchannel bandwidth \( B = 1800 \text{kHz}, \) SNR Gap \( \Gamma = 1, \) Noise Power \( N_0 = 1 \times 10^{-9} \text{W}, \) Noise variance \( \sigma^2 = N_0 B = 1.8 \times 10^{-4} \text{watt-sec}, \) \( \epsilon = 0.006. \)

Maximum Power constraint on each subchannel (mw):

\[
P^1_i = P^2_i = 8, \quad \forall i = 1, 2, \ldots, 10.
\]
Rate requirement of the users in kbps:
\[ \bar{R}^1 = [260, 300, 380, 430], \]
\[ \bar{R}^2 = [295, 395, 385, 445]. \]

Subchannel gain matrices; \( G^1, G^2, G^{1,2}, G^{2,1} \) were obtained by sampling from Gaussian distributions \( \mathcal{N}(1, 1), \mathcal{N}(0.5, 1), \mathcal{N}(0, 1), \mathcal{N}(0, 1) \) and taking square.

Interference matrices (mw):
\[
I^1 = \begin{pmatrix}
11 & 12 & 9 & 3 \\
1 & 11 & 8 & 6 \\
13 & 9 & 3 & 4 \\
11 & 5 & 4 & 13 \\
7 & 12 & 7 & 6 \\
6 & 7 & 3 & 2 \\
6 & 5 & 12 & 13 \\
4 & 14 & 2 & 14 \\
7 & 13 & 3 & 6 \\
7 & 8 & 2 & 1
\end{pmatrix},
I^2 = \begin{pmatrix}
10 & 1 & 3 & 8 \\
11 & 6 & 12 & 2 \\
6 & 1 & 6 & 12 \\
1 & 14 & 13 & 9 \\
3 & 0 & 2 & 5 \\
13 & 11 & 3 & 7 \\
2 & 12 & 2 & 6 \\
12 & 13 & 2 & 1 \\
8 & 1 & 13 & 3 \\
14 & 5 & 8 & 1
\end{pmatrix}.
\]

For this example, there were enough resources to satisfy each user’s requirements in each FC. The solution obtained for the downlink for Algorithm 1 and the optimization algorithm via best response is as follows. Actually due to very high complexity of the mixed integer programming algorithm, we use a suboptimal algorithm for the best response also and call it OPT. For that, FC \( k \) computes the channel allocation by binary integer programming by taking \( P^k_i \) as the power at each channel \( i \) and then for that allocation uses KKT conditions to minimize sum power to get the desired rate.

Subchannel Allocations at NE: \( CA_{RA} \) and \( CA_{OPT} \) obtained by RA and the optimal best response algorithms are are given by \( (CA^k(i) \triangleq j \text{ if } A^k_{ij} = 1) \):
\[
CA_{RA}^1 = [2 1 1 2 3 3 2 1 4 4],
CA_{RA}^2 = [2 1 3 3 2 2 2 4 1 4],
CA_{OPT}^1 = [2 1 4 2 3 3 2 4 1 4],
CA_{OPT}^2 = [2 1 3 3 2 2 2 4 1 2].
\]

Minimal powers needed for transmission for the RA and the optimal algorithm at the NE are (in mw) given as
\[
P^1_{RA} = [8.0, 8.0, 6.0, 8.0, 5.4, 8.0, 5.8, 0.0, 0.8, 2.8, 0.0],
P^2_{RA} = [0.5, 6.1, 3.0, 4.0, 0.9, 0.0, 0.8, 2.8, 0.0],
P^1_{OPT} = [8.0, 8.0, 0.8, 8.0, 5.4, 8.0, 5.5, 0.5, 5.8, 6.5],
P^2_{OPT} = [0.5, 8.0, 3.0, 3.0, 0.9, 0.0, 0.8, 1.6, 0].
\]

Power in each subchannel for FC 2 for Algorithm 1 is plotted in Fig 1 and for OPT in Fig 2. The channel allocations and powers on most of the channels in both the algorithms are close at the NE. Convergence of the utility (sum of powers) in both FCs is very close for the two algorithms at NE. From these Figs we see that the sum powers in both FCs are very close at the NE. Next we provide an example to obtain a fair NE for (11)-(13). We use a system with 2 FCs each with 2 users and 4 subchannels for calculation of NE. We generated the subchannel gain matrices \( G^1, G^2, G^{1,2}, G^{2,1} \) as before. The interference matrices (mw) are
\[
I^1 = \begin{pmatrix}
4 & 3 \\
11 & 4 \\
8 & 9 \\
9 & 13
\end{pmatrix}, I^2 = \begin{pmatrix}
10 & 1 \\
11 & 6 \\
6 & 1 \\
1 & 14
\end{pmatrix}.
\]
Maximum powers (mw) allocated to the subchannels are,
\[ P^1 = [2.1, 1.2, 8.0, 1.8], \]
\[ P^2 = [1.1, 1.5, 7.5, 1.9]. \]

Rate requirement of the users, in kbps, is,
\[ \tilde{R}^1 = [200, 220], \]
\[ \tilde{R}^2 = [160, 180]. \]

We have verified that the rate requirement of all the users cannot be satisfied for this case. Thus, we obtain a fair NE via Algorithm 3. There are multiple NE. Depending on the initial conditions, the algorithm may converge to a different NE. We show the convergence of our algorithm in Fig 5. We have observed that the regret algorithm converges to the same NE in this example even though we started with different initial conditions. For this NE, the subchannel allocation for the two FC’s converges to 
\[ CA^1 = [2, 1, 1, 1], CA^2 = [2, 2, 1, 2] \]
and the power allocations are 
\[ P^1 \text{ (in mw)} = [2.1, 1.2, 8.0, 1.8], \]
\[ P^2 \text{ (in mw)} = [1.1, 1.5, 7.5, 1.9] \]
with the utilities 0.4294, 0.6348 and the allocated rates (in kbps) [85.8, 97.9], [123.5, 114.2].

VI. CONCLUSIONS

We have considered a channel and power allocation problem in a multiple FC environment when there are multiple users in each FC requiring different minimum rates. The channels need to be shared by the different FCs such that they do not cause much interference to each other and to the MC users. We have formulated the problem in a non-cooperative game theory setup and provided low complexity distributed algorithms to obtain Nash equilibria and correlated equilibria.

REFERENCES

[31] 3GPP TR 36.921, 36.922: “FDD, TDD Home eNodeB (HeNB) Radio Frequency (RF) requirements analysis”.

Fig. 5. Correlated Equilibrium using Algorithm-3