Optimal Distributed Frequency Planning for OFDMA Femtocell Networks

Emanuel B. Rodrigues
Wireless Telecommunications Research Group (GTEL)
Federal University of Ceará (UFC)
Fortaleza, Brazil
Email: emanuel@gtel.ufc.br

Fernando Casadevall
Department of Signal Theory and Communications (TSC)
Universitat Politècnica de Catalunya (UPC)
Barcelona, Spain
Email: ferranc@tsc.upc.edu

***Abstract*—Femtocell networks have gained momentum due to their important benefits, such as improved indoor coverage, higher areal spectral efficiency, enhanced signal quality, among others. Those benefits are only achievable if adequate deployment decisions and efficient resource allocation techniques are able to assure the seamless co-existence among Femtocell Access Points (FAPs) within the femtocell tier and between FAPs and macrocell users. In this work, we study the frequency planning problem on the femtocell tier deployed on a closed access mode and using a dedicated spectrum. A novel radio resource allocation technique that performs a mid/long-term frequency planning for the FAPs in the femtocell tier is proposed. It is concluded in this paper that the Dynamic Frequency Planning (DFP) algorithm based on the Branch and Bound technique is able to find the optimal frequency planning according to any desired criterion and that the proposed algorithm is suitable for implementation in a distributed 4G femtocell network regarding signaling overhead and latency aspects.**

I. Introduction

Femtocell Access Points (FAPs) are small cellular base stations that are deployed by the end-users on their home or office premisses. Some characteristics of the FAPs are: reduced power and coverage area, low-cost, and direct connection to the backbone IP-based network. They were initially designed by the mobile operators to extend indoor coverage, aiming to solve the problem of coverage holes, improve system capacity and offload data traffic from the Macrocell Base Station (MBS), allowing the mobile operator to focus on outdoor and mobile users. Some other key benefits of femtocells are infrastructure cost reduction and signal quality enhancement.

Femtocells have the advantage for the operators that they already have the functionalities of a Base Station Controller (BSC) and they are in principle paid and deployed by the customers themselves. However, this big scalability must be guaranteed by proper self-organizing Radio Resource Management (RRM) techniques that are able to guarantee the seamless co-existence between the macro and femto tiers as well as performing interference management within the femtocell tier.

In this work we assume that femtocells are deployed in a Small Office and Home Office (SOHO) and home environments using closed access mode [1] and that macro and femtocells use orthogonal frequency bands.

Since a dedicated spectrum for the femtocells eliminates the cross-tier interference between the FAPs and the MBS, the main RRM problems left to be solved will reside in the femtocell tier. Moreover, it is expected that the majority of FAPs will be installed in a dense urban scenario. Such a scenario can be visualized as a crowded group of multi-flat apartments in a dense city center, which has a potential for high interference between neighboring femtocells. In this case, the femto-to-femto interference avoidance and frequency planning problems are the main challenges to be overtaken.

II. Related Work and Main Contributions

Some works have studied interference management in the femtocell tier, for example [2]–[4]. Two interference avoidance techniques were proposed in [4]: in the first one, the FAP prioritizes the usage of its sub-carriers based on a quality indicator, and the second technique aims to minimize the sum of the overall interference suffered by the users connected to the FAPs. Although the cross-tier interference management was the objective in [2], the authors formulated an algorithm that could also be applied in the femtocell tier. The objective was to minimize the overall network interference constrained by the interference restriction matrix and a required number of sub-channels per FAP. A simple randomized frequency planning was proposed in [3], where each FAP accesses a random subset of the available frequency resources in order to avoid persistent collisions with neighboring FAPs.

None of the techniques proposed in [2]–[4] are able to guarantee a complete interference avoidance. Furthermore, there is a lack of frequency planning policies able to provide a macroscopic planning of the resources to be used by the FAPs. The present work presents a novel interference management approach for the femtocell tier, facing the RRM problem from a hierarchical perspective: first determine the resource allocation to the FAPs following a given policy, and next decide the resource assignment to the end-users. Therefore, instead of looking directly into the short-term resource assignment to the end-users, we propose to firstly prepare a mid/long-term action plan for the femtocell resource allocation. This is performed by the Dynamic Frequency Planning (DFP) algorithm, which decides how many frequency resources must be allocated to each FAP. This self-organizing RRM framework completely avoids interference and allows the seamless co-existence of several FAPs in any interference topology.

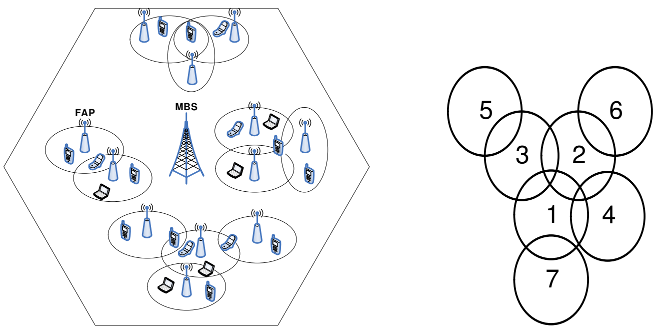
978-1-4577-1348-4/13/$31.00 ©2013 IEEE

Once the frequency resources are allocated to the FAPs, each FAP must decide the resource assignment to the users. The algorithms that assign the sub-carriers to the users can be simplified, because the interference was already avoided by the dynamic frequency planning and it is expected that the users will have good channel conditions for the majority of the sub-carriers due to the proximity to the FAP antenna. Therefore, we focus our attention on the resource allocation to the FAPs and leave the resource assignment to the users out of the scope of this paper.

The rest of the paper is organized as follows. The proposed frequency planning strategy is described in section III, where a general optimization problem is formulated, a distributed framework is described and an algorithimic description of the DFP technique is presented. Some discussion about the simulation results is presented, and finally section IV draws the main conclusions and perspectives for future work.

III. Frequency Planning for OFDMA Femtocell Networks

We assume that the FAPs that interfere mutually are organized in clusters, and there may exist several isolated clusters of FAPs, as illustrated in Fig. 1(a). A FAP can be aware that it belongs to a cluster using different strategies. Some options are: 1) The FAP has cognitive radio capabilities and sense its environment in order to find its neighbors; 2) Mobile terminals in the overlapping area of two or more FAPs inform the existence of mutual interference.



(a) System scenario

(b) Example of a cluster of FAPs - $N = 7$.

Fig. 1. Femtocell network.

In this work, we are interested at studying the frequency planning problem in each of these clusters of FAPs. Since there may exist a plurality of cluster topologies, the objective of this work is to propose a general frequency planning technique that is valid for any cluster topology. How the clusters are formed or identified is out of the sope of this paper.

In section III-A, we characterize the frequency planning paradigm considered in this work by formulating a general optimization problem. We propose to solve this problem by using a novel DFP algorithm, whose objective is to decide the number of frequency resources that must be allocated to each FAP. Details about the algorithmic implementation are presented in sections III-B and III-C.

A. General Optimization Problem

In this section, we formulate the frequency planning problem in an Orthogonal Frequency Division Multiple Access (OFDMA)-based femtocell tier as an integer optimization problem, as indicated below.

$$\max_{F_n} f(F_n), \tag{1}$$

$$\text{subject to} \sum_{n \in G_m} F_n \leq K, \quad \forall m = 1:M, \tag{2}$$

$$F_n > 0, \quad \forall n = 1:N, \tag{3}$$

where N is the number of FAPs in a cluster; K is the total number of frequency resources (sub-carriers) dedicated to the femtocell tier; F_n is the number of frequency resources allocated to the nth FAP; M is the total number of possible groups of mutual interfering FAPs; G_m is the set of indexes of the FAPs belonging to a given group of mutual interfering FAPs; and $f(F_n)$ is the objective function that depends on F_n and must be maximized. Details on the mechanisms how to compute M and G_m are given in sub-section III-B.

In a cluster of FAPs there may exist different groups G_m where their components interfere mutually inside the group. Constraint (2) requires that the FAPs inside each of these groups must not share the same sub-carriers and the sum of these frequency resources must not surpass the total bandwidth of the femtocell tier. A part or the total amount of the sub-carriers used by one group can be reused by other groups as far as they are not mutually interfering. Constraint (3) says that each FAP must be allocated at least one frequency resource.

Based on the general formulation presented above, different frequency planning policies with their respective objective functions can be proposed. In this work, rather than considering any specific policy, we propose an optimal DFP framework using the Branch and Bound (BnB) technique that is general and can be easily particularized for every desired policy, as can be seen in the following sections.

B. General Distributed Framework

We propose a frequency planning technique that can be implemented in a distributed network architecture. In a distributed approach, there are several processing entities that execute the DFP algorithm. Each of these entities is located in a different FAP. In this way, all clusters in the femtocell tier can calculate their resource allocation in a parallel and autonomous way. Neighbor FAPs that form a cluster communicate among them using the air interface to send broadcast messages and/or measurement reports [5]. If the coverage areas of neighbor FAPs are highly overlapped, they can communicate directly between them. If they are not visible to each other, a mobile terminal in the overlapping area can act as a relay to allow the communication between the FAPs. However, it may exist non-interfering FAP that belongs to the same cluster. In this case, a flooding procedure can be used to allow the communication between non-neighbor FAPs across the cluster. Some examples of naive and optimized flooding protocols for wireless mesh networks can be found in [6].

Fig. 2 depicts a basic processing and communication protocol for the execution of the DFP algorithm considering a distributed network architecture. As said before, the processing entities are located at each FAP. For didactic reasons, this figure shows only three FAPs that are assumed to belong to the same cluster. The DFP algorithm is meant to determine the frequency planning for a single cluster of FAPs. Therefore, the DFP solutions for all clusters can be found in a parallel way.

In order to exemplify this procedure, let us consider a cluster of 7 FAPs in the topology presented in Fig. 1(b).

Firstly, the FAPs exchange local interference topology information between them using a flooding protocol. In this way, after some messages, every FAP in the cluster can have interference information about all other FAPs in that cluster. Before deciding the final DFP allocation, the DFP algorithm needs to process some information (pre-processing steps).

With the interference topology information in hands, the processing entity creates an interference topology matrix $\mathbf{A}$, representing the interference pattern of the cluster. This is an N x N matrix, where N is the number of FAPs in the cluster, and each element a_{ij} of the matrix $\mathbf{A}$ indicates whether the ith FAP interferes with (is a neighbor of) the jth FAP or not.

Following that, the processing entity finds the groups G_m of mutual interfering FAPs, which have an important role on the constraints of the frequency planning optimization problem. In the example above, one can notice that there are a total of M=5 mutual interfering groups G_m, which are given by $G_1 = \{1, 2, 3\}$, $G_2 = \{1, 2, 4\}$, $G_3 = \{1, 7\}$, $G_4 = \{2, 6\}$ and $G_5 = \{3, 5\}$.

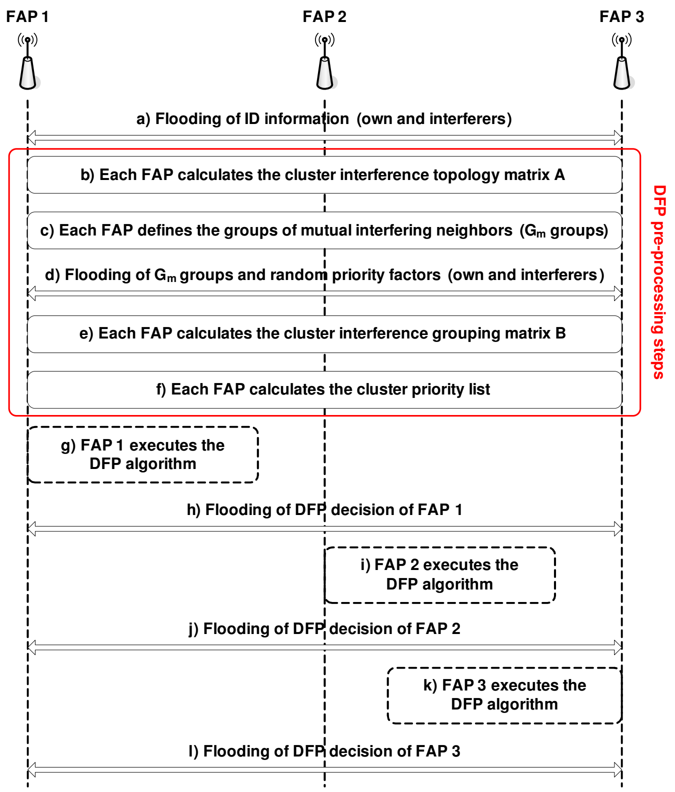


Fig. 2. Basic processing and communication protocol of the frequency planning technique in a distributed architecture.

The next step is to define the cluster interference grouping matrix $\mathbf{B}$ with dimension N x N, where each element b_{ij} indicates the number of G_m groups of size j that the ith FAP participates. If $b_{ij} = 0$, the ith FAP does not participate on any G_m group of size j.

Using matrices $\mathbf{A}$ and $\mathbf{B}$, the processing entity is able to calculate the cluster priority list in which the FAPs must be processed to define the frequency planning. The calculation of this priority list is a critical issue for the correctness of the proposed DFP algorithm. A reliable way to define the priority order suitable for our purposes is to look at the columns of matrix $\mathbf{B}$ from the right to the left. The FAPs that belong to larger groups are the ones that suffer more interference and so must be the first ones to define their frequency planning. If two or more FAPs have the same b_{ij} in the rightmost column, the casting vote will be given by the columns on the left. If two or more FAPs have a parity in all instances, the final decision is taken based on a FAP-specific random priority factor.

Once this pre-processing steps are done, the processing entity is able to calculate the DFP allocation according to the chosen frequency planning policy.

C. Optimal Branch and Bound-Based DFP Algorithm

The BnB technique [7], [8] is a mathematical/algorithimic tool widely used to solve large scale combinatorial optimization problems, including NP-hard problems.

The BnB algorithm searches the complete space of solutions of a given problem for the optimal solution. The use of branching and bounding procedures enables the algorithm to search parts of the solution space only implicitly, and so accelerate the finding of the optimal solution.

In order to explain the branching procedure, we present Fig. 3, which illustrates the mapping of our DFP problem into the structure of a dynamic search tree. For didactic reasons, we consider a simple example characterized by a cluster of two interfering FAPs ($N = 2$) that share four sub-carriers ($K = 4$).

As illustrated in Fig. 3, this search tree has three levels when completely built ($N + 1$ levels, where N is the number of FAPs). Level 0 only represents the root node, i.e. the complete solution space. The other levels correspond to each one of the N FAPs in the cluster in the order indicated by the cluster priority list. Each node of the tree is represented by a vector Q with N elements. In the case of an internal node located in the nth level (associated to the nth FAP), the vector Q corresponding to this node is defined as $Q = [q_1, \ldots, q_n, q_{n+1}, \ldots, q_N] = [x_1, \ldots, x_n, y_{n+1}, \ldots, y_N]$. Let us assume a vector $X_n \subseteq Q \mid X_n = [x_1, \ldots, x_n]$, which represents the possible frequency allocations of the predecessors of the nth FAP in the priority list, including itself (elements in black color in Fig. 3). Let us also define a vector $Y_n \subseteq Q \mid Y_n = [y_{n+1}, \ldots, y_N]$, which represents an upper bound for the possible frequency allocations associated to the succeeders of the nth FAP in the priority list (elements in red color in Fig. 3). As can be seen, the vector Q is the union of vectors X_n and Y_n. In the example shown in the figure, the nodes of the first level associated with FAP 1 are internal nodes represented by a vector $Q = [x_1, y_2]$.

The root father node is branched into several children nodes located at the first level by applying the constraints associated with FAP 1. This is equivalent to divide the complete solution space in several subspaces, where each subspace is represented by a different children node. FAP 1 can hypothetically choose any number of resources for its own allocation, from 1 to 3, since it has the highest priority and is the first to define its allocation. This range of values is determined by the optimization constraints of our DFP problem. In our illustrative case, FAP 1 is interfered by FAP 2, which must use at least one frequency resource. Therefore the maximum number of resources to be allocated to FAP 1 shall be 3 in this example. Each possible frequency allocation for FAP 1 corresponds to a different node in the first level of the tree, yielding 3 different nodes. Considering the first level of the tree, the vector X_1 of possible frequency allocations has only one element and there are 3 different vectors X_1, one for each of the nodes in the first level. The elements of the vector Y_1 associated to each node of the first level are upper bounds of the possible frequency allocations for FAP 2 taking into account the optimization constraints and the frequency allocation already fixed for FAP 1. That is why the vector Y of each node depends on the vector X defined for that specific node.

The same reasoning applies to the other levels. In the second level, vector X_2 of possible frequency allocations is composed of two elements: the first element is defined by the corresponding father node located in the first level, and the second element represents one possible frequency allocation for FAP 2 (branching procedure). Since in our example we consider only two FAPs, the second level is the last level of the tree and the nodes belonging to this level are terminal nodes (leaves). There are no vectors Y_N associated with terminal nodes because it is not necessary to compute upper bounds for the frequency allocation in this case. There are two reasons for that: 1) there are not succeeders of FAP N because it is the last in the priority list to be processed; and 2) all terminal nodes represent feasible solutions X_N since the frequency allocations represented by these nodes take into account the allocation and interference constraints of all FAPs in the cluster.

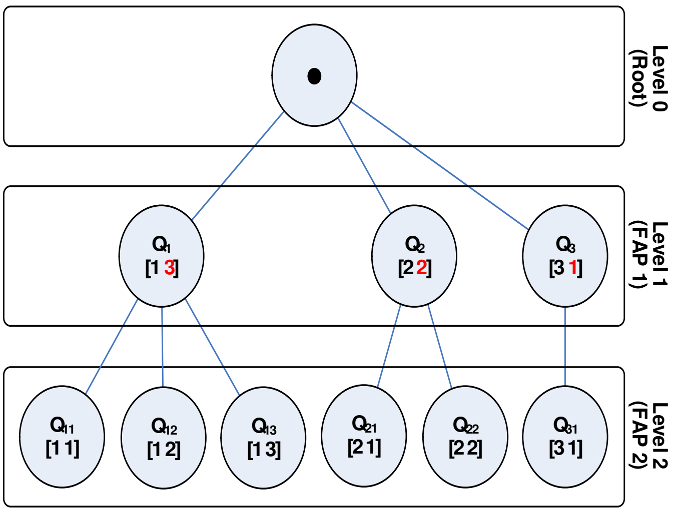


Fig. 3. Example of a dynamic search tree of the BnB-based DFP algorithm considering two interfering FAPs and four sub-carriers.

Fig. 3 shows a complete dynamic search tree for illustration purposes. In the execution of the BnB algorithm, it is not necessary to create all the branches of the tree. In order to reduce the search of the optimal solution, a bounding procedure is necessary to determine which branches of the search tree can possibly contain the optimal solution, discarding those branches that cannot contain it. The bounding procedure is explained in the following.

Considering a maximization problem, let us define a bounding function $g(\cdot)$, which is an upper bound of the objective function $f(\cdot)$ such that $g(\cdot) \geq f(\cdot)$ over the region of feasible solutions. When a terminal node of the tree is processed for the first time, the feasible solution X_N represented by that node is saved as the current best solution, which is called incumbent. During the execution of the BnB algorithm, the nodes of the search tree are processed. For a generic node Q, a bounding procedure is applied by calculating $g(Q)$. This upper bounded value is compared with the incumbent. If the bounded value is lower than the incumbent, it means that the subspace associated to node Q cannot contain the optimal solution and so the subproblem is discarded (or fathomed). Otherwise, the node Q is saved in the pool of live subproblems for further branching. The search terminates when there is no unexplored parts of the solution space, and then the optimal solution is the one recorded as the incumbent. Algorithm 1 sketches the pseudo-code of a general BnB-based DFP algorithm, which summarizes all the concepts explained above.

The BnB-based DFP algorithm has to be adapted to the desired frequency planning policy. What differentiates the particular DFP algorithms are the objective and bounding functions. The objective functions $f(\cdot)$ are equal to the selected bounding functions $g(\cdot)$. The difference is that the bounding function is used for any node Q of the search tree, while the objective function is used only for the feasible solutions X_N corresponding to the terminal nodes (leaves of the tree).

In order to evaluate the DFP algorithm, several simulations of clusters of FAPs with different topologies were performed. We considered several cluster sizes ($N = 4, \ldots, 7$) and different total numbers of frequency resources in the femtocell tier ($K = 15$ or 25). The chosen values for K were based on the number of Physical Resource Blocks (PRBs) standardized for the Long Term Evolution (LTE) system considering a bandwidth of 3 and 5 MHz, respectively. Fixing N and K, different frequency planning policies with their respective objective and bounding functions were evaluated over 500 different cluster topologies. Examples of objective functions are the sum of frequency allocations in a cluster, which is a metric of spectral efficiency, and the minimum number of frequency allocations in a cluster, which is a metric of fairness.

Algorithm 1: General BnB-based DFP algorithm.

Initialization

$Incumbent \leftarrow -\infty$ // Current best value of the objective function
$Solution \leftarrow 0$ // Current best solution
$UB(Q_0) \leftarrow g(Q_0)$ // Upper bound of the initial node Q_0
$Live \leftarrow \{Q_0, UB(Q_0)\}$ // Save the node Q_0 and its upper bound value on the set of alive nodes

Processing of the nodes in the search tree

repeat
  Select the node Q from the set $Live$ to be processed according to the *Best First* selection strategy [7]
  $Live = Live \setminus \{Q, UB(Q)\}$ // Remove the father node Q from the set of alive nodes
  Branch on the father node Q generating the children nodes $Q_1, \ldots, Q_k$
  for $i = 1$ **to** k **do**
    $UB(Q_i) \leftarrow g(Q_i)$ // Upper bound of the node Q_i
    if $UB(Q_i) = f(X_N)$ **and** $f(X_N) > Incumbent$ **then** // $f(X_N)$ is the value of the objective function for a feasible solution X_N
      $Incumbent \leftarrow f(X_N)$ // Save the current best value of the objective function
      $Solution \leftarrow X_N$ // Save the current best solution
      continue // Jump to the next iteration of the *for* loop
    end if
    if $UB(Q_i) \leq Incumbent$ **then**
      Fathom Q_i // Discard the node Q_i
    else
      $Live \leftarrow Live \bigcup \{Q_i, UB(Q_i)\}$ // Save the node Q_i and its upper bound value on the set of alive nodes
    end if
  end for
until $Live = \varnothing$

It was concluded from the simulations that the BnB technique was able to find the optimal planning of frequency allocations that maximizes the objective functions in all considered scenarios. Without a proper optimization tool, it is extremely difficult to find the optimal solution of the integer optimization problem described by (1)-(3), even for simple cluster topologies. Considering the sum of frequency allocations as the objective function, the proposed BnB-based DFP algorithm was able to find the optimal solution in an average time of 124 miliseconds for $N = 4$ and $K = 15$, and 61 seconds for $N = 7$ and $K = 25$.

It is worth pointing out that an important advantage of our DFP technique is the fact that it does not need to be executed so frequently. The resource allocation defined by the DFP algorithm for a given group of FAPs is valid until the interference pattern of that group changes. The interference topology changes only if the owner of a FAP changes its position or a FAP is turned on or off. Since these changes are not frequent events, the resource allocation determined by the proposed technique will remain valid in the mid/long term, and the signalling overhead generated will be negligible. Based on these arguments, we claim that the periodicity of the algorithm can be in the order of minutes, which is sufficient to guarantee an accurate interference avoidance.

IV. Conclusions and Perspectives

A novel technique that performs a macroscopic, mid/long-term frequency planning on generic clusters of FAPs in the femtocell tier is proposed. The optimization problem is solved by the DFP algorithm, which defines the allocation of frequency resources among the FAPs. It is based on the Branch and Bound technique, which is able to find the optimal solution in all scenarios and cluster topologies evaluated.

Since the proposed algorithm does not need to be executed so frequently, the signaling overhead in a distributed network is negligible and the latency involved in the allocation decision is adequate for implementation on a 4G femtocell network.

As a continuation of the work, we plan to propose heuristic versions of the DFP algorithm in order to find sub-optimal solutions sufficiently close to the optimal in less computational time. Furthermore, we intend to study and evaluate several frequency planning policies and their impact on the spectral efficiency and fairness among FAPs in the femtocell tier.

Acknowledgment

Emanuel B. Rodrigues would like to thank CAPES (Brazil) for a Ph.D. Scholarship support, and also CNPq (Brazil) and FUNCAP (Ceará, Brazil) for a Postdoctoral Scholarship support.

References

[1] G. de la Roche, A. Valcarce, D. López-Pérez, and J. Zhang, "Access control mechanisms for femtocells," *IEEE Communications Magazine*, vol. 48, no. 1, pp. 33–39, Jan. 2010.

[2] D. López-Pérez, G. de la Roche, A. Valcarce, A. J uttner, and J. Zhang, "Interference avoidance and dynamic frequency planning for WiMAX femtocells networks," in *Proc. 11th IEEE Singapore International Conference on Communication Systems - ICCS*, Nov. 2008, pp. 1579–1584.

[3] V. Chandrasekhar and J. G. Andrews, "Spectrum allocation in tiered cellular networks," *IEEE Transactions on Wireless Communications*, vol. 57, no. 10, pp. 3059–3068, Oct. 2009.

[4] D. López-Pérez, A. Ladanyi, and J. Zhang, "OFDMA femtocells: A self-organizing approach for frequency assignment," in *Proc. IEEE 20th Int. Symp. on Personal, Indoor and Mobile Radio Communications - PIMRC*, Sep. 2009, pp. 2202–2207.

[5] J. Zhang and G. de la Roche, *Femtocells: Technologies and Deployment*. John Wiley & Sons, 2010.

[6] T. Zahn, G. O'Shea, and A. Rowstron, "An empirical study of flooding in mesh networks," Microsoft Research, Cambridge, UK, Tech. Rep. MSR-TR-2009-37, 2009.

[7] J. Clausen, "Branch and bound algorithms - principles and examples," Departament of Computer Science, University of Copenhagen, Tech. Rep., 1999.

[8] S. Boyd and J. Mattingley, "Branch and bound methods," Stanford University, Tech. Rep., 2007.